



The Mainframe Software Partner
For The Next 50 Years

Abend-AID Fault Analytics Installation Guide

Release 16.03

Please direct questions about Abend-AID Fault Analytics
or comments on this document to:

Compuware Customer Support

<http://go.compuware.com>

This document and the product referenced in it are subject to the following legends:

Copyright 1999-2016 Compuware Corporation. All rights reserved. Unpublished rights reserved under the Copyright Laws of the United States.

U.S. GOVERNMENT RIGHTS-Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in Compuware Corporation license agreement and as provided in DFARS 227.7202-1(a) and 227.7202-3(a) (1995), DFARS 252.227-7013(c)(1)(ii) (OCT 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14 (ALT III), as applicable. Compuware Corporation.

This product contains confidential information and trade secrets of Compuware Corporation. Use, disclosure, or reproduction is prohibited without the prior express written permission of Compuware Corporation. Access is limited to authorized users. Use of this product is subject to the terms and conditions of the user's License Agreement with Compuware Corporation.

Abend-AID, Abend-AID for CICS, Abend-AID Fault Analytics, Compuware, and FrontLine are trademarks or registered trademarks of Compuware Corporation.

DB2, IBM, MVS, and z/OS are trademarks or registered trademarks of International Business Machines Corporation.

Adobe® Reader® is a trademark of Adobe Systems Incorporated in the United States and/or other countries.

All other company and product names are trademarks or registered trademarks of their respective owners.

Contents

Introduction	5
Contents of this Guide	5
Who Should Read this Guide	5
Online Documentation	5
Online Help	6
Customer Support	6
Compuware Go Customer Support Website	6
Contacting Customer Support	7
Phone	7
Web	7
Mail	7
Corporate Website	7
Chapter 1. Planning the Fault Analytics Installation	9
Overview	9
Where the Components Run	9
How the Components Communicate	10
Individuals to Participate in the Installation	10
Fault Analytics Collector Installation	11
Compuware Enterprise Services Installation	11
Memory Size Considerations for Linux and z/OS UNIX Installations	11
Chapter 2. The Fault Analytics Database	13
Chapter 3. Fault Analytics Installation on z/OS UNIX	15
Before Beginning	15
Installing Fault Analytics	15
Launching Fault Analytics	16
Chapter 4. The z/OS Fault Analytics Collector	17
Overview	17
z/OS Version Support	17
Datasets	17
FAC Instances	17
LNKLST and LPALIB	18
Storage Use	18
System Security	19
User-Written Fault Events	19
Default Subsystem	19
Subsystem-Directing DD Statement	19
Migrating to Fault Analytics 16.3	19
Verifying the Installation	20
Communicating with CES	21
IP Versions	21
Connection Parameters	21
Running the FAC	21
SYSIN Parameters	22
General SYSIN Dataset Format	23
Comments	23
Continuation Lines	23
Operator Commands	23

Log Open Command	23
Log Close Command	24
Shutdown Command	24
Emergency Termination	24
\$42FADEL	25
Copying the Log for Transmission	25
Chapter 5. User-Defined Events API – Mainframe	27
API Format	28
COBOL	28
Assembler	28
C/C++	28
PL/I	28
Debugging	28
Processing the API Call	29
Using the Fault Analytics Custom Mainframe API	29
Return Codes	30
Coding Examples	30
General Description	30
Input to CFMAPI	31
Sample Input to CFMAPI:	31
Input to COBFMAPI	31
Sample Input to COBFMAPI:	31
Appendix A. Fault Analytics Installation on Linux	33
Before Beginning	33
Installing Fault Analytics	33
Launching Fault Analytics	34
Appendix B. Fault Analytics Installation on Windows	35
Before Beginning	35
Installing Fault Analytics	35
Launching Fault Analytics	36
Appendix C. Fault Analytics Fields	37
Appendix D. Fault Analytics Custom API Fields	41
Appendix E. Using Fault Analytics with Abend-AID	45
Abend-AID	45
Abend-AID for CICS	45
Appendix F. Fault Analytics Export and Load Process	47
Glossary	51
Index	53

Introduction

Contents of this Guide

- **Chapter 1, “Planning the Fault Analytics Installation”** — Describes items to consider while preparing your site for Abend-AID Fault Analytics installation.
- **Chapter 2, “The Fault Analytics Database”**— Describes the installation process for the four supported databases: DB2 for z/OS, Microsoft SQL Server, DB2 for Linux UNIX, and Windows (LUW), and Oracle.
- **Chapter 3, “Fault Analytics Installation on z/OS UNIX”** — Provides detailed instructions for the Compuware-recommended Fault Analytics installation.
- **Chapter 4, “The z/OS Fault Analytics Collector”**— Provides detailed instructions for installing the Fault Analytics Collector.
- **Chapter 5, “User-Defined Events API – Mainframe”**— Describes how to use the mainframe API call for user-defined events.
- **Appendix A, “Fault Analytics Installation on Linux”**— Provides the instructions for the Fault Analytics Install Anywhere procedure for Linux.
- **Appendix B, “Fault Analytics Installation on Windows”** — Provides the instructions for the Fault Analytics installation procedure on Windows.
- **Appendix C, “Fault Analytics Fields”** — Describes each of the Fault Analytics fields.
- **Appendix D, “Fault Analytics Custom API Fields”** — Describes each of the custom API fields.
- **Appendix E, “Using Fault Analytics with Abend-AID”**— Details how to enable Abend-AID and/or Abend-AID for CICS so they can provide Fault Analytics transaction report and region fault data.
- **Appendix F, “Fault Analytics Export and Load Process”**— Provides detailed procedures for converting a Fault Analytics database from SQL Server to DB2.
- **“Glossary”** — Defines various terms used in relation to Fault Analytics.

Who Should Read this Guide

The *Abend-AID Fault Analytics Installation Guide* is intended for system administrators or programmers who are responsible for implementing Fault Analytics. Fault Analytics online help and documentation set contain additional reference and usage information.

Online Documentation

User documentation is provided in two online formats with Abend-AID: PDF and HTML. These formats are available on Compuware’s FrontLine customer support website at <http://go.compuware.com>.

Print complete documentation sets as needed from the PDF files on the product documentation site.

The following documents comprise the Abend-AID documentation set:

- *Abend-AID Installation Guide*: Provides instructions for installing Fault Analytics, Abend-AID for CICS, and the Abend-AID Viewer. It includes information about setting up these products to fit the development and system configurations at your site.
- *Abend-AID Messages and Codes*: Lists the informational and error messages and their explanations that you may encounter while using Fault Analytics, Abend-AID for CICS or the Abend-AID Viewer. It also describes any recommended user response. These messages are also available for downloading from the FrontLine website to a Chicago-Soft, Ltd.'s MVS/QuickRef™ database.
- *Abend-AID Reference Summary*: Provides a handy summary of the Abend-AID Viewer, Abend-AID for CICS, the Abend-AID Web Browser Interface, Fault Analytics report sections, Abend-AID SNAP-AID, and Fault Analytics DD statements.
- *Abend-AID Release Notes*: Provides a summary of the enhancements for the current release and the supported environments.
- *Abend-AID User/Reference Guide*: Provides guidelines and instructions for using Fault Analytics basic language support, Extended Language Support (XLS), SNAP-AID, Abend-AID for DB2, Abend-AID for IMS, Abend-AID for IDMS, Fault Analytics DD statements, and the Abend-AID Viewer.
- *Abend-AID for CICS User's Guide*: Describes the functions and features of Abend-AID for CICS. It also contains problem determination examples that illustrate the product's use.
- *Abend-AID Fault Analytics Installation Guide*: Provides information about how to install, configure, and customize the Abend-AID Fault Analytics option. Refer to the online help for information about using this option.

Online Help

The Abend-AID Fault Analytics online help includes help for customizing and viewing Management Reports, and reference information.

Click **Help** > **Help Contents** to open the online help for Fault Analytics:

- Click **Contents** to see an outline of available topics designed as a user guide.
- Click **Index** to see a list of key terms in alphabetical order.
- Click **Search** to hunt for topics.
- Click **Glossary** for terminology definitions.

After opening a dialog box and clicking within it, press **F1** to open online help for that dialog box.

Customer Support

Compuware provides a variety of support resources to make it easy for you to find the information you need.

Compuware Go Customer Support Website

You can access online information for Compuware products via our Compuware Go customer support website at <http://go.compuware.com>.

Compuware Go provides access to critical information about your Compuware products. You can review frequently asked questions, read or download documentation, access product fixes, or e-mail your questions or comments. The first time you access Compuware Go, you are required to register and obtain a password. Registration is free.

Compuware now offers User Communities, online forums to collaborate, network, and exchange best practices with other Compuware solution users worldwide. To join, go to <http://groups.compuware.com>.

Contacting Customer Support

Phone

- USA and Canada: 1-800-538-7822 or 1-313-227-5444.
- All other countries: Contact your local Compuware office. Contact information is available at <http://go.compuware.com>.

Web

You can report issues via the Quick Link **Create & View Support Cases** on the Compuware Go home page.

Note: Please report all high-priority issues by telephone.

Mail

Compuware Customer Support
Compuware Corporation
One Campus Martius
Detroit, MI 48226-5099

Corporate Website

To access Compuware's site on the Web, go to <http://www.compuware.com>.

The Compuware site provides a variety of product and support information.

If you need support services, please obtain the following information before calling Compuware's 24-hour product support hotline:

- The name and release number of the Fault Analytics product you are using. This information is available from the **Help** menu by clicking **About**.
- If you have an issue with the Fault Analytics Collector, provide Compuware Customer Support with the entire SYSPRINT report that contains the errors. This report can be found using SDSF under ISPF on the LPAR that is running the Event Processor in error.
- Installation information including:
 - Installed options
 - Whether the product uses local or network databases
 - Whether the product is installed in the default directories
 - Whether the product is a standalone or network installation
- Environment information, such as the operating system and release on which the product is installed, network specifications, and the names and releases of other applications that were running.
- The location of the problem in the Fault Analytics option software, and the actions taken before the problem occurred.
- The exact product error message, if any.
- The exact application, licensing, or operating system error messages, if any.

Chapter 1.

Planning the Fault Analytics Installation

Overview

The Abend-AID Fault Analytics option is made up of the following components, each of which must be installed:

- **z/OS Fault Analytics Collector (FAC):** collects fault events on a z/OS LPAR and sends these events to Fault Analytics. This is installed as part of Abend-AID/Common.
- **Fault Analytics Web Application:** receives fault events from one or more Fault Analytics Collectors and stores those events in a database. The FAS is Java-based and therefore is zAAP-eligible.

Follow the steps in the following chapters to complete each installation:

- **Chapter 2, “The Fault Analytics Database”**
- **Chapter 4, “The z/OS Fault Analytics Collector”**

Where the Components Run

The Fault Analytics Collector (FAC) runs on each LPAR in your enterprise from which you want fault events collected. Compuware's Abend-AID products are the main source of fault events. Other sources are jobs that end with JCL errors, jobs that end with non-zero return codes, and jobs that abend when no dump was produced (thus, no Abend-AID report). In addition, your own user-generated events are collected.

Abend-AID or Abend-AID for CICS is not required on an LPAR that is running the FAC. The most complete information about an abend fault event is obtained from Abend-AID or Abend-AID for CICS. You are not restricted to only one FAC on any given LPAR. You may run as many as you want.

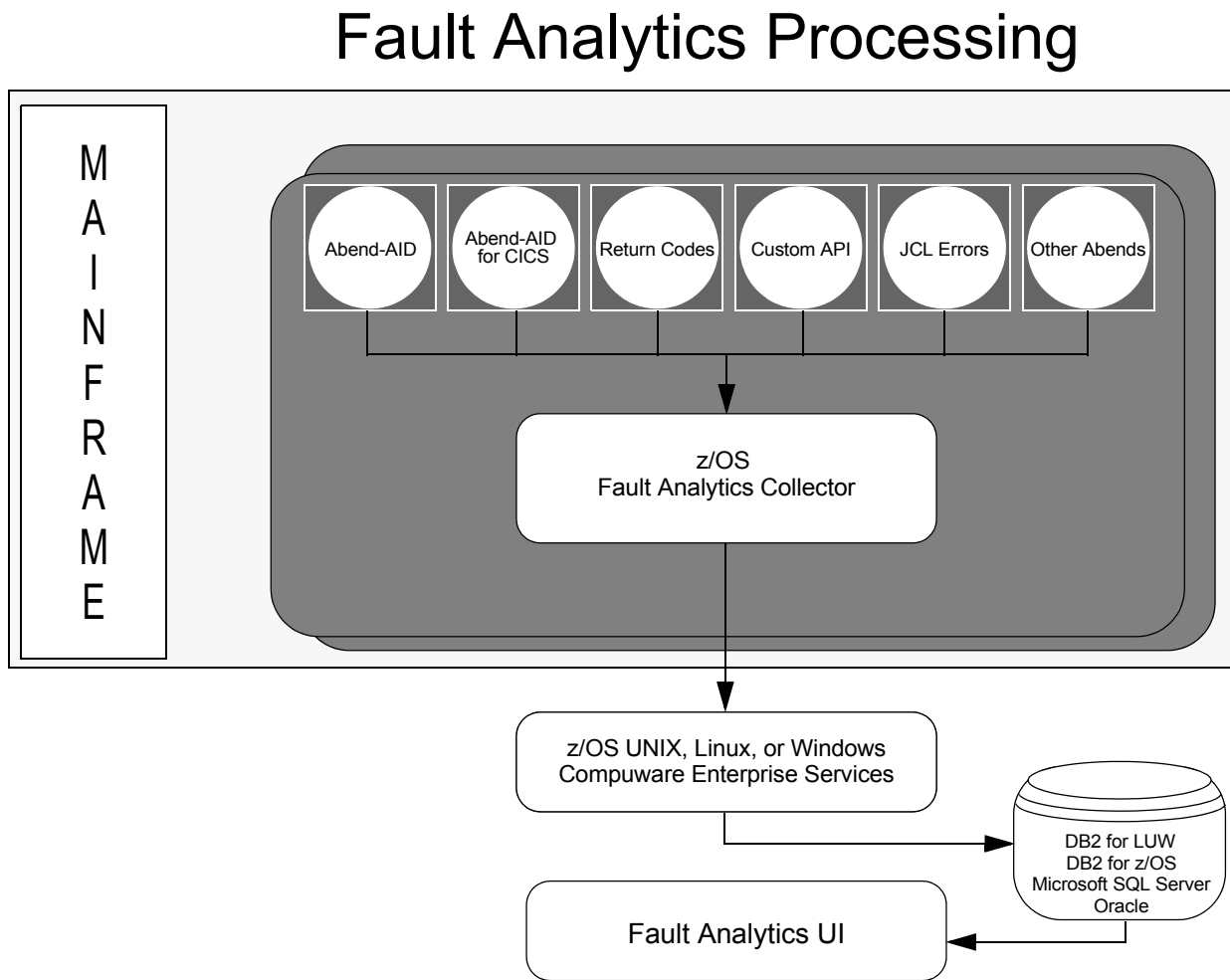
Fault Analytics runs on z/OS UNIX, Linux, or Windows. One instance of Fault Analytics can receive events from any number of FACs. The number of fault events and the topology of your network are two of the factors that determine how many FAC instances are required for your situation.

Fault Analytics is a web application that can be accessed from any of the supported web browsers.

Fault Analytics supports DB2 for z/OS, SQL Server, Oracle, and DB2 LUW.

The following diagram gives a pictorial overview of the components, their communication partners, and their processing.

Figure 1-1. Fault Analytics Event Processing



How the Components Communicate

All communication between the Fault Analytics Collector and the Compuware Enterprise Services (CES) is via TCP/IP. Remember, however, that:

- A Fault Analytics Collector must be running on each LPAR from which you want fault events collected.
- At least one instance of CES must be available.

Individuals to Participate in the Installation

Use the installation planning worksheet in **Appendix A, “Installation Planning Worksheet”** to help you identify the individuals who should participate in and to record information needed to complete the installation.

Fault Analytics Collector Installation

The Fault Analytics Collector is installed as part of Abend-AID/Common. Configuration of the Fault Analytics Collector is found in the *Abend-AID Installation and Customization Guide*.

The Fault Analytics Collector (FAC) requires the following:

- A familiarity with SMP/E and the mainframe environment.
 - Access and permissions to alter the datasets created under the high-level identifier used in the media browser.
 - RACF authority to:
 - Update a link list dataset.
 - Either update an existing APF-authorized dataset or to create a new one.
 - Update your RACF rules to assign a user ID for a started task if a started task is desired.
 - Knowledge to update the JCL to execute the FAC or to create a started task in your PROCLIB.
 - A systems programmer to:
 - Assign a four-character subsystem ID for each FAC.
 - Ensure that this ID is unique across the entire LPAR.
- Note:** The SYS1.PARMLIB must *not* be updated as part of the Fault Analytics installation.
- Your installation's Work Load Manager (WLM) support personnel, if you intend to use the zIIP-enabled SRB function of the FAC. The utilization of zIIP execution for the SRB is not required from a WLM point of view, but upon its execution, your zIIP utilization will increase. If you choose not to use this feature, then all of the FAC's execution will be on a GCP.

Compuware Enterprise Services Installation

Refer to the Compuware Enterprise Services Installation and Customization Guide for information on installing CES.

Memory Size Considerations for Linux and z/OS UNIX Installations

Ensure that the Fault Analytics install directory, the CES install directory, and the temporary directories are all backed by sufficient space in HFS/ZFS files mounted at those directories.

The System `/tmp` directory is configured in the z/OS parmlib BPXPRMxx member. There should be a section as follows:

```
MOUNT FILESYSTEM('/TMP')
TYPE(TFS)
MODE(RDWR)
PARAM('-s 2048')
MOUNTPOINT('/tmp')
```

If you are unable to alter the System `/tmp` size value for your site, then you will need to point the install script to a directory that is backed by a larger HFS/ZFS file system. For example, pointing the `install.sh` script to a user-defined temporary directory within the user's home directory (see Figure 1-2).

Figure 1-2. Sample install.sh altered for alternative temporary directory

```
export JAVA_HOME=/VERSYSB/usr/lpp/java/J7.1_64
export PATH=$JAVA_HOME/bin:$PATH
export LIBPATH=$JAVA_HOME/lib:$LIBPATH:.
export TMPDIR=/u/username/tmp
java -Xnoargsconversion -Dfile.encoding=ISO8859-1 -Dlax.nl.env.IATEMPDIR=/u/username/tmp/ -jar install.jar
-i console
```

Chapter 2.

The Fault Analytics Database

The database used by Fault Analytics is described in the Compuware Enterprise Services Installation and Configuration Guide. Refer to that manual for detailed information on setting up the database.

Chapter 3.

Fault Analytics Installation on z/OS UNIX

This chapter describes the Compuware-recommended Abend-AID Fault Analytics installation on z/OS UNIX.

A z/OS UNIX system administrator should install Fault Analytics.

Before Beginning

Before you begin the installation, do the following:

- Install Compuware Enterprise Services (CES). If multiple CES instances are installed, determine which instance to associate with this Fault Analytics installation. Refer to the *Compuware Enterprise Services Installation and Configuration Guide* for installation instructions.
- Install and configure a supported database.
- Determine the Fault Analytics port to be used.
- The installation may require up to 55,000 1k blocks in the temp folder and 55,000 1k blocks in the installation folder during the installation. When the installation is complete, the files are removed.

Installing Fault Analytics

1. Transfer the z/OS UNIX folder from the media or FTP image to the mainframe as binary.
2. Log onto the mainframe and navigate to an OMVS command prompt. Change the directory to the location where the `install.sh` file was transferred.
3. At the prompt, enter the command `chmod 777 install.sh` to give full access permissions.
4. Edit the `install.sh` file to include site-specific information.
5. Be sure the `install.sh` file has execute authority and execute it. The **Welcome** panel appears.

Note: Type **quit** at any time to cancel the installation, or type **back** to move back.

6. At the prompt, enter the path to the installation folder for a supported version of Java, then press **Enter**.
7. If the system temp folder does not have the 55,000 1k blocks of pre-allocated space available for the installation, type **Y** at the prompt, type another location, and press **Enter**. Otherwise, type **N** (the default) and press **Enter**. The **Introduction** panel appears.
8. Read the **Introduction** panel, then click **Enter**. The **Fault Analytics** panel appears.
9. At the prompt, do one of the following to indicate the path to the Compuware Enterprise Services instance to associate with this installation of Fault Analytics:
 - To accept the default path, which is `/opt/Compuware/CES`, press **Enter**.

- To change the path, type the path to the Compuware Enterprise Services instance to associate with this installation of Fault Analytics, then press **Enter**.

The **License Agreement** panel appears.

10. Read the License Agreement, then type **Y** to accept the terms of the agreement. The **Fault Analytics** panel appears.
11. At the prompt, type a valid Fault Analytics port number between 1024 and 65535 to be used to transmit messages, then press **Enter**. The **Ready to Install** panel appears.
12. Review the installation information and, if correct, press **Enter**. If incorrect, type **back** and make any necessary revisions. When the installation is finished, the **Installation Complete** panel appears and lists the installation folder, Fault Analytics port, and Fault Analytics URL.

CAUTION:

Before exiting the installer, open the Fault Analytics URL in a browser and bookmark the page so you can easily access Fault Analytics in the future.

13. Press **Enter** to exit the installer.
14. Restart the Compuware Enterprise Services service.

Launching Fault Analytics

1. Fault Analytics is launched via the URL that appeared on the **Install Complete** panel at the end of installation. If this URL was not bookmarked at that time, it can be identified by opening the config.properties file, which is located at *<your path>/CES/data/fa/config* by default.
2. Find the host.name= parameter, for example ABC123456N01.mycompanydomain.corp.
3. In a browser, enter the URL in the following format: `http://<host.name>:<CES port number>/aafa`.

Note: Log files are located in the Compuware Enterprise Services data directory specified during CES installation, which is *<CES install directory>/data/fa/logs* by default. Contact the CES installer to determine the location if necessary.

Chapter 4.

The z/OS Fault Analytics Collector

Overview

The z/OS Fault Analytics Collector (FAC) is the mainframe component of Compuware's Fault Analytics system. It runs on all LPARs from which you want to collect fault events and can run with or without Compuware's Abend-AID products. The FAC is installed as part of Abend-AID Common Components. You generate the FAC jobs by using the Abend-AID Installation and Customization Dialog. Refer to the *Abend-AID Installation and Customization Guide* for more information.

You choose what information you want FAC to capture. These choices are made via the parameters in the SYSIN parameter dataset described "SYSIN Parameters" on page 22.

On each LPAR, FAC consists of the following:

- One long-running address space
- An optional z/OS subsystem routine that looks for JCL errors and operator commands
- An optional SMF accounting exit routine that looks for jobs that end with completion codes that you want reported to Fault Analytics.

The following sections give important information about the Fault Analytics Collector.

z/OS Version Support

The LPARs on which the FAC is executing must have z/OS Version 1.13 or more current.

Datasets

The FAC requires one mandatory and one optional dataset. The mandatory dataset is a VSAM RRDS, which holds the queue of records collected by FAC pending completion of processing. Storing the events reduces the likelihood that they will be lost due to communication or system errors. The optional dataset is a VSAM KSDS, which contains log/trace records.

Each of these datasets is defined automatically via the IDCAMS program using parameters you supply in the SYSIN parameter dataset. These datasets cannot be manually pre-allocated by you. The queue dataset is defined only once and is used thereafter. The log dataset can be defined fresh for each use or you can append new log information onto an existing log.

Compuware recommends that you turn logging on only when asked to do so by Compuware Customer Support because it can adversely affect FAC performance.

FAC Instances

You can execute multiple instances of the FAC simultaneously on one LPAR. For example, you may want to test a new or updated version while running your production version.

Each FAC instance must have a unique four-character subsystem ID. This ID distinguishes one FAC from another, as well as differentiating it from other subsystems running on the LPAR. The subsystem ID can be the same on every LPAR; however, if you run more than

one FAC on the same LPAR, each FAC must have a different subsystem ID. This subsystem ID is specified in the SYSIN parameter dataset. You may want to check with your system programmers to ensure that the value you pick is not already used by another subsystem. Note that this subsystem must *not* be defined in the SYS1.PARMLIB IEFSSNxx member. If it is, the FAC will fail to start.

Not only can multiple FAC instances run on a single LPAR, earlier versions of Fault Analytics can run on the same LPAR as the FAC. All events processed by the FAC can also be processed by an earlier supported version of Fault Analytics. This allows you to compare the event processing and ensure the new version is working as expected.

Each instance of the FAC has its own SYSIN parameter dataset and therefore each can collect the same or different sets of JCL errors, steps ending with non-zero completion codes, and steps that have abended but Abend-AID did not process. All FAC instances process all events generated by Abend-AID, License Management System, and user-written programs. The FAC is enabled as soon as you place the module AAFMSTUB into the LNKST dataset available to Abend-AID and you issue the F LLA,REFRESH operator command. See “Migrating to Fault Analytics 16.3” on page 19 for more information.

LNKST and LPALIB

The following six Fault Analytics modules, located in the Abend-AID Common Modules SKAZAUTH authorized library, must be available to all programs and must reside in a LNKST dataset:

- AAFMSTUB
- CWAFPUBL
- CWFEM
- CWFEMX
- MFMAGUID
- MFMAPUBX

Compuware recommends that you do not place any modules in your LPALIB.

With the exception of the modules listed above, execute all other FAC modules from a //STEPLIB or //JOB LIB DD statement in the JCL used to start FAC. This job can be a started task or a batch job. Executing FAC from a STEPLIB helps to ensure that the correct version of each module is used. In addition, doing so allows multiple versions to exist and to be executed without impacting each other.

The STEPLIB or JOBLIB from which FAC is executed must be APF-Authorized. Only modules MFMMAIN and MFMDELET are bound with AC=1. All other modules are bound with AC=0. Doing so conforms to IBM's system integrity guidelines.

In accordance with IBM's system integrity guidelines, all access to the FAC is via a stacking, non-space-switching PC routine, a z/OS subsystem, and an instance of the IEFACTRT SMF accounting exit routine.

Storage Use

Use of CSA (below the 16M line common storage) is restricted only to the subsystem interface control blocks, and use of ECSA (above the 16M line common storage) is kept to an absolute minimum. Shared data spaces are employed for logging and queuing in order to not use extended common storage more than is absolutely necessary. All modules are loaded either in private storage or in extended common storage. The FAC SYSPRINT report dataset (which can be browsed via SDSF while FAC is executing), lists all common storage use so that you can see and plan for FAC execution. In addition, this dataset shows all modules used along with their latest assembly date and time and the highest Compuware issue tracking number associated with each module. This report dataset will help in debugging later.

If you execute RMF and look at storage remaining after a job terminates, you may see 1328 bytes of CSA storage attributed to the FAC job. These bytes are used for the SSCT and SSVT subsystem control blocks and must never be freed. Therefore, the storage is shown on the RMF report. Do not be concerned about this storage.

System Security

RACROUTE (interfacing to RACF, ACF/2, or TOPSECRET) is used to ensure that the user ID under which the FAC address space is running has ACCESS(ALTER) to the queue and the log datasets when these need to be defined or ACCESS(CONTROL) when the datasets exist and need only to be updated.

No attempt is made by FAC to limit which user IDs can enter operator commands. This is an installation responsibility if it is desired. Refer to “Operator Commands” on page 23.

User-Written Fault Events

Any programs that you have written for earlier versions of Fault Analytics that call CWFM to publish your own events will continue to execute with the new Fault Analytics. No changes to these programs are required.

Default Subsystem

The default subsystem specification has been deprecated in Fault Analytics Release 12.3. Once the module named AAFMSTUB is installed in the LNKLST dataset, all instances of the FAC receive all events generated by Abend-AID, License Management System, and by user-written programs.

Subsystem-Directing DD Statement

An FAC subsystem-directing DD statement takes the form //AAFMxxxx DD DUMMY, where xxxx is replaced with the four-character subsystem ID that you want to handle fault events from this program. This DD statement is placed into the JCL of the job that may fail and that will publish fault events. This DD statement names the FAC subsystem that is to handle any fault events generated by the programs executed in this job step.

Note: Do not place this DD statement in the FAC JCL, but instead place it in the job that will abend and for which you want the abend event captured by the subsystem named in the DD statement.

The named FAC subsystem must be active. If it is not, the fault event is ignored and the data is lost. When the subsystem-directing DD statement is present, only the one subsystem named in the DD statement processes the event, regardless of the number of instances of the FAC active on the LPAR.

In the absence of any subsystem-directing DD statement, all fault events are processed by all instances of the FAC that are active on the LPAR. If the subsystem-directing DD is present, only the subsystem named in the DDNAME processes the event.

JCL errors, steps ending with non-zero completion codes, and steps that have abended but Abend-AID did not process have always been processed by all instances of the FAC. Now events generated by Abend-AID are processed in this manner as well.

Migrating to Fault Analytics 16.3

Each of your site’s LPARs can be changed to use FAC 16.3 one at a time. In addition, you can run both Fault Analytics 12.4 and FAC 16.3 releases at the same time on the same

LPAR, and you can have fault events processed by all instances of the FAC on the LPAR. In this way you can ensure that FAC 16.3 is executing correctly.

Compuware recommends that you install FAC on a test LPAR first. Copy the six modules previously discussed in “LNKLST and LPALIB” on page 18 into a link-listed dataset on this LPAR. If this dataset is shared with other LPARs, they will be affected by the presence of the new modules because changes have been made to these modules for FAC 16.3. The AAFMSTUB module in 16.3 sends each event from Abend-AID, License Management System, and from user-written programs to every active instance of the FAC on the LPAR.

Once you have the six modules in a LNKLST dataset, you can start the FAC 16.3 address space. This FAC plus any other instances of the FAC that are version 12.4 will process all events as well. Each instance of the FAC is invoked for each possible event, and depending on the parameters you have set for each, either processes or ignores the event.

Verifying the Installation

1. Submit the JCL located in the Abend-AID installation sample library (CTL) member \$46FAVER, to ensure that the modules that must be in a link list dataset are properly available.

This job prints a report giving the status of each module that must be in a link list dataset. The report created by \$46FAVER should resemble that shown below.

Figure 4-1. Sample \$46FAVER Report

```

MFM222E   AAFMSTUB   WAS FOUND IN LINK LIST DATA SET:
MFM224I           VOL=SMS900 DSN=ENTQA.AF.SMFMAUTH
MFM225I CORRECTLY BOUND AS REENTRANT
MFM227I           AND IS AT VERSION 120300

MFM223I   CWAFPUBL  WAS FOUND IN LINK LIST DATA SET:
MFM224I           VOL=SMS900 DSN=ENTQA.AF.SMFMAUTH
MFM225I CORRECTLY BOUND AS REENTRANT
MFM227I           AND IS AT VERSION 120300

MFM222E   CWF       WAS FOUND IN LINK LIST DATA SET:
MFM224I           VOL=SMS900 DSN=ENTQA.AF.SMFMAUTH
MFM225I CORRECTLY BOUND AS REENTRANT
MFM227I           AND IS AT VERSION 120300

MFM222E   CWFMX     WAS FOUND IN LINK LIST DATA SET:
MFM224I           VOL=SMS900 DSN=ENTQA.AF.SMFMAUTH
MFM225I CORRECTLY BOUND AS REENTRANT
MFM227I           AND IS AT VERSION 120300

MFM223I   MFMAGUID  WAS FOUND IN LINK LIST DATA SET:
MFM224I           VOL=SMS900 DSN=ENTQA.AF.SMFMAUTH
MFM225I CORRECTLY BOUND AS REENTRANT
MFM227I           AND IS AT VERSION 120300

MFM223I   MFMAPUBX  WAS FOUND IN LINK LIST DATA SET:
MFM224I           VOL=SMS900 DSN=ENTQA.AF.SMFMAUTH
MFM225I CORRECTLY BOUND AS REENTRANT
MFM227I           AND IS AT VERSION 120300

```

Any errors found by \$46FAVER will be flagged with an asterisk, and the text of the error will indicate what was found to be wrong. This program will end with a return code of 12 if any errors are found.

2. Correct any errors before attempting to execute the Fault Analytics Collector (FAC).

Your link list datasets may be APF-authorized, and their names and volume serials will differ. These do not constitute errors and will not be flagged with an asterisk.

3. Repeat this process on every LPAR on which you intend to collect fault events. Every time you start the FAC, the verification program is automatically run and the results are shown in the SYSPRINT dataset. Hence you can see over time whether the FAC in your link list dataset has changed.

Communicating with CES

The Fault Analytics Collector (FAC) establishes a TCP/IP connection with a single instance of Compuware Enterprise Services.

The FAC maintains a copy of all fault events on disk until the CES has replied that it has taken responsibility for the event. Once the FAC receives the reply, the event is deleted.

IP Versions

Communication with the CES can be via IPv4 or IPv6. The FAC always tries to obtain an IPv6 socket, but if this fails, reverts to IPv4. You do not need to enable IPv6 on all your LPARs at the same time because communication between IPv4 and IPv6 is correctly handled by TCP/IP.

You can specify a full IP address or IP name of the FAS using the IPv4 or IPv6 display formats.

Connection Parameters

One connection between each FAC and its associated CES is established by the FAC. The CES never establishes a connection with an FAC. Thus, each CES is considered a *server* and each FAC is considered a *client*. This distinction exists only for the setting up of communications. Once a connection exists, either end can initiate a request. The other end will respond with a reply. The connection exists until:

- Either the FAC or the CES terminates.
- An operator terminates the connection.
- A system or communication failure occurs.

The FAC attempts to reestablish the connection on a timed basis in every case except the second above. The operator must reestablish any connection that he has previously terminated.

Running the FAC

The Fault Analytics Collector (FAC) is a long-running address space active on each LPAR where you want to collect fault events. You can execute the FAC from JCL as a batch job or from a procedure as a started task. The Abend-AID installation sample library (CTL) members are \$40FACJB and \$41FACPR.

Choose whether to execute the FAC as a batch job or a started task.

- If you choose the batch job option, the z/OS STOP and MODIFY commands are directed to the *jobname* you specified in the JCL.
- If you choose the started task option, the z/OS STOP and MODIFY commands are directed to the *identifier* you specified on the START command. For example, if you named the procedure in your procedure library CPWRAAFM, you issue the following command to start the FAC:

```
S CPWRAAFM.AAFM.
```

In the above example, AAFM is the identifier of the stated task. It can be used to display information about the task or stop the task. Refer to the *MVS System Commands* manual for details.

Notes on the JCL:

1. Compuware recommends that a REGION parameter of 0M be specified in order that the private area for FAC not be unduly constrained. As indicated earlier in this document, the storage usage has been carefully analyzed to insure that no more than is necessary is consumed.
2. TIME=NOLIMIT is recommended because FAC is a long running job and should not be canceled due to exceeding the installation's CPU time maximum.
3. You can change the job class or performance group to achieve a specific storage and CPU time use instead of using the JCL default parameters.
4. PARM='LANGUAGE=EN' is required.
5. The SYSPRINT and SYSIN DD statements are both required but may point to DASD datasets instead of JES datasets if desired. The DCB characteristics for the SYSPRINT dataset must be RECFM=FBA,LRECL=133,BLKSIZE= * multiple of 133 *. The DCB characteristics for the SYSIN dataset must be RECFM=FB,LRECL=80,BLKSIZE= * multiple of 80 *.
6. The FREE=CLOSE and SPIN=UNALLOC parameters on the SYSPRINT DD statement are required if you have chosen the Automatic Restart Manager (ARM) feature of the FAC. These parameters cause JES/2 to retain each SYSPRINT dataset for each initiation of the FAC as a separate dataset. You can see each execution separately by browsing these datasets.
7. The STEPLIB (or JOBLIB) DD statement if present must specify an APF-authorized load library. If this library is not APF-Authorized, FAC abends with an S047 abend code.

If desired, you can place a procedure in your system PROCLIB and execute this PROC at IPL time using the automatic command facility. The initialization program for Abend-AID can run before or after starting FAC.

SYSIN Parameters

All values that tailor the execution of the FAC reside in the dataset pointed to by DDNAME SYSIN. This dataset can be in-line (via //SYSIN DD *) or in a separate dataset (via //SYSIN DD DSN=*hlq*.MKAZ241.CTL(MFMPARM),DISP=SHR). The dataset must contain 80 character records, unblocked or blocked to any multiple of 80 bytes.

When a change is made to the SYSIN dataset, the FAC must be stopped and restarted for the change to take effect. Whenever you apply PTF maintenance to the FAC, always check the FAC SAMPLE datasets. Any changes to the parameters or additions to them are reflected in this dataset as part of the SMP/E APPLY processing.

You can use the Abend-AID installation sample library (CTL) member MFMPARM as a template for your SYSIN parameters for the FAC. You can make any parameter you do not want into a comment (asterisk in CC 1), and you can change any values that are specific to your installation.

A description for each parameter is included in the MFMPARM member in the Abend-AID installation sample library (CTL) library. New parameters and/or changed ones will be documented in the MFMPARM member, which will be distributed with any PTF that changes the existing values. Refer to the description of each parameter to see which ones you must change in order to run the FAC.

General SYSIN Dataset Format

Positions 1 through 72 of each statement can contain parameter data. Positions 73 through 80 are ignored and can contain a sequence number, but are not required to do so.

Each parameter takes the form KEYWORD(value). Each keyword and the requirements for the values associated with it are documented in the MFMPARM member of the SMFMCNTL library.

Comments

A comment is specified by placing an asterisk (*) in position 1 of a statement. The entire statement is considered part of the comment. Additionally, a comment can be anything enclosed between a slash-asterisk (/*) and an asterisk-slash (*). Multiple lines can be contained within comments of this type.

Continuation Lines

If a parameter value must span multiple lines, specify the value through position 72 and continue the value on the next statement starting in position 1. As many lines as necessary can be used in this way, but each parameter keyword has its own restrictions.

Operator Commands

The FAC monitors all commands entered at any system console. When the FAC recognizes its command character, it processes the command. For example, if you define the *at* sign (@) as the command character, any text entered at a system console that is preceded by the @ character will be processed only by the corresponding FAC, not by any other subsystem component.

The command character is defined in the parameter dataset and, by default, the MODIFY *jobname* command is used. You can specify a command character for each FAC; however, it must be unique for each FAC. Multiple FAC programs in one LPAR *cannot* share the same command character. They must either have no command character specified, or each must have its own unique character. You are not required to define a command character.

All operator commands can also be entered using the z/OS MODIFY command. For instance, MODIFY *jobname*,SHUTDOWN would be equivalent to entering @SHUTDOWN. The verb MODIFY can be abbreviated F. The z/OS STOP command, abbreviated P, is also supported and is discussed in “Shutdown Command” on page 24.

Other methods of entering operator commands include sending them to a particular FAC or entering them from an ISPF interface on the z/OS LPAR.

Console messages indicating the success or failure of operator commands and console messages supplying information requested by the operator command are sent to the system console entering the command.

Log Open Command

The Log Open command causes logging to begin. It can be entered in one of two ways:

- Using the command character defined to the FAC
- Using the z/OS MODIFY (F) command.

Note that all of the parameters concerning logging (the dataset name, volser, tracks, and optional SMS parameters) must have been specified when the FAC started. They cannot be added dynamically.

An example of the MODIFY command is:

```
F jobname,LOG OPEN
```

Log Close Command

The Log Close command causes logging to end and all system resources consumed by the log task to be freed. It can be entered in one of two ways:

- Using the command character defined to the FAC
- Using the z/OS MODIFY (F) command.

When the log is closed, you can print it using the log print program, or you can REPRO it to a sequential dataset and transmit it to Compuware.

An example of the MODIFY command is:

```
F jobname,LOG CLOSE
```

Shutdown Command

The Shutdown commands shuts down one or more components of the system. They can be entered in any of three ways:

- Using the command character defined to FAC
- Using the z/OS MODIFY (F) command
- Using the z/OS STOP (P) command

Operands on the Shutdown command are optional and specify the manner of the shutdown. These operands are NORMAL and IMMED. NORMAL is the default.

NORMAL is the safest method of shutting down one or more components of the system and, barring any extremely unusual conditions, ensures that no fault events in flight at the time of the shutdown are lost. The FAC stops accepting fault events. If the connection with the FAC is active and if the communication is in either fault event state or fault event state pending, the FAC gracefully terminates the connection. If the communication is not in either of these states, the TCP/IP connection is closed immediately. The z/OS STOP (P) command is equivalent to entering **SHUTDOWN NORMAL**.

IMMED is the fastest way of terminating one or more components of the system and does not guarantee that in-flight fault events are processed. The FAC stops accepting fault events. If the connection with the CES is active and if the communication is in fault event state or fault event state pending, the FAC terminates the connection immediately. Fault events that have not been sent remain on the queue until the system is restarted, at which time they are sent. The FAC terminates immediately.

You can issue a second shutdown command. For example, a SHUTDOWN NORMAL command can be overridden by a SHUTDOWN IMMED command.

Emergency Termination

The FAC makes every effort possible to release system resources, especially common storage, when it terminates. This is true whether the termination is normal due to a SHUTDOWN command or abnormal due to a CANCEL command or a programabend. ESTAEX and FRR routines are always in effect to trap abnormal terminations and to free all system resources.

There may, however, be a situation in which the FAC cannot be cancelled due to a z/OS anomaly or failure that precludes the functioning of the CANCEL command. If you are unable to communicate with FAC and want to terminate it, follow the steps below in order. Stop issuing the commands once the FAC terminates.

1. Issue the **STOP *jobname*** command.
2. Issue the **MODIFY *jobname*,SHUTDOWN IMMED** command.
3. Issue the **CANCEL *jobname*** command.
4. Issue the **FORCE *jobname*** command.

Attempt to restart the FAC. If the restart is successful, you can continue to run this job. If the restart is not successful and the FAC indicates that it cannot continue, you can execute the emergency cleanup job, \$42FADEL, which is described below.

\$42FADEL

\$42FADEL is a stand-alone job, located in the Abend-AID installation sample library (CTL). It attempts to terminate a specified FAC job and attempts to release all system resources held by it.

The PARM= on the EXEC statement must specify the four-character subsystem ID of the Fault Analytics Collector instance you want to terminate.

User ABEND 100 and 101 may occur and messages describing the abends will be written to the job log. User ABEND 100 indicates that the PARM= is not exactly four bytes long, and User ABEND 101 indicates that the subsystem ID specified on the PARM= operand cannot be found in the system. Messages MFM400 through MFM403 may be written to the job log as well.

Copying the Log for Transmission

At the request of Compuware Customer Support, you may be asked to start the logging facility for a specified period of time and to transmit the log to Compuware for analysis.

The logging facility must be disabled before this log copy job will run. Issue the FAC **LOG CLOSE** command to stop logging and free the log for copying.

Note: Running the log for long periods of time degrades performance of the Fault Analytics Collector and can consume a large amount of disk space.

Sample JCL to copy the Fault Analytics Collector log to a sequential dataset is shown in Figure 4-2.

Figure 4-2. Sample Copy JCL for Fault Analytics Collector Log

```

//*
//*      Place a valid JOB statement here
//*
//DELETE EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
        DELETE *** sequential data set name *** NONVSAM
        SET MAXCC=0
/*
//REPRO EXEC PGM=IDCAMS
//LOGIN DD DSN=*** VSAM Log Data Set Name ***,DISP=OLD
//LOGOUT DD DSN=&&SEQ,DISP=(,PASS),UNIT=SYSDA,
//        SPACE=(CYL,(100,100),RLSE),VOL=SER=??????.
//        DCB=(RECFM=VB,LRECL=32756,BLKSIZE=32760,DSORG=PS)
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
        REPRO INFILE(LOGIN) OUTFILE(LOGOUT)
/*
//TERSE EXEC PGM=AMATERSE,PARM=PACK
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=&&SEQ,DISP=(OLD,DELETE,DELETE)
//SYSUT2 DD DSN=*** sequential data set name ***,DISP=(,CATLG),
//        VOL=SER=??????.UNIT=SYSDA,
//        SPACE=(CYL,(100,100),RLSE)

```

Chapter 5.

User-Defined Events API – Mainframe

User-defined events can be generated on the mainframe through an API call. The API consists of a call to the Fault Analytics module CWFm. The only parameter required is the address of the record area containing a set of comma-separated fields (CSV file).

Each field requires both a field name and its data. For example, passing the job name XUSERID0A would be:

```
JOBNAME,XUSERID0A
```

The name is *not* case-sensitive. All data items are in character format. If the data contains a comma, the data field must be enclosed in double quotes.

Note: A binary zero must delimit the end of the record area.

The fields in Table 5-1 are required for each event.

Table 5-1. Required Event Fields

Field Name	Description
EventName	Name of the event to identify it
ErrorOccurredAtDate	Date of event: MM/DD/YYYY
ErrorOccurredAtTime	Time of event: HH:MM:SS
PublisherName	Name of the application that detected the event. For example, DCOMERR
PublisherVersion	Version of the application that detected the event
ErrorCode1	Unique error code for the problematic application
JobName	Job name of the problematic application

Refer to Appendix D, “Fault Analytics Custom API Fields” on page 41 for a complete list of fields that can be input to Fault Analytics.

A typical CSV record would resemble that shown in Figure 5-1.

Figure 5-1. Typical CSV Record Format

```
EVENTNAME,DCOM,ERROROCCURREDATDATE,04/24/2014,
ERROROCCURREDATTIME,16:25:22,
PUBLISHERNAME,DCOMERR,
PUBLISHERVERSION,V01R01,
ERRORCODE1,36,JOBNAME,DCOMJOB1,
PROGRAMNAME,DCOMPROG,DESCRIPTION,Uxxx-NOT OPEN
```

API Format

COBOL

```
01 RET-CODE      PIC S9(8)  USAGE IS BINARY.
01 CSV-REC-AREA PIC X(1024).
```

```
MOVE LOW-VALUES TO CSV-REC-AREA.
```

Insert comma-separated values in area.

```
CALL 'CWFM' USING CSV-REC-AREA RETURNING RET-CODE.
IF RET-CODE NOT EQUAL 0 ...
```

An alternative using the COBOL special register RETURN-CODE would resemble the following:

```
CALL 'CWFM' using CSV-REC-AREA.
IF RETURN-CODE NOT EQUAL 0 ...
```

Assembler

```
LA    R2,CSV_REC_AREA      POINT TO RECORD AREA
CALL  CWFM,((R2))          CALL FM MOD
```

or

```
LINK EP=CWFM,PARAM=((R2)) LINK TO MODULE
LTR  R15,R15                TEST RETURN CODE
```

C/C++

```
extern __asm int CWFM(char *csvrec); // prototype

int return_code; // return-code
char *CSV_REC_AREA; // pointer to CSV Rec

return_code = CWFM(CSV_REC_AREA); // Call API
```

PL/I

```
DCL RETCODE      BIN FIXED(31);
DCL CSVRECAREA  CHAR(1024) VARYINGZ;
DCL CWFM        EXTERNAL ENTRY ( CHAR(1024) VARYINGZ BYVALUE);
                RETURNS (BIN FIXED(31) BYVALUE);

RETCODE = CWFM ( CSVRECAREA );
```

Note: You must specify the compile option DEFAULT(LINKAGE(SYSTEM)).

Debugging

If the JCL statement

```
//AFDEBUG DD DUMMY
```

is present in the job-step, *debug* is turned on, and any rejection writes the appropriate error message to the job log. See “Return Codes” on page 30 for error messages.

Processing the API Call

After the call is made to the API, several record validation checks are performed:

- The record passed is checked for a terminating binary zero.
- The call is rejected if the terminating binary zero is not present or the total length of the entire event record exceeds 2048.
- A check is made to each Fault Analytics field name. Any names not defined cause the record to be rejected. (See Appendix D, “Fault Analytics Custom API Fields” for a list of valid Fault Analytics field names.)
- The length of each field is checked. If it exceeds 255 or the allowable field size, the record is rejected.
- The CSV record is checked to ensure all required fields are passed. If any one is not passed, the record is rejected.
- A check is made for each field to ensure that no duplicate fields are present. Duplicate fields also cause a rejection.
- Events for normal abend error codes already being detected by Abend-AID and delivered to Fault Analytics cannot be generated via the API. If the field ERRORCODE1 has contents such as SOC7, ASRA, or four-digit MVS abend codes, the request is rejected.

If the record passes the checks above, it is sent for subsequent processing to Fault Analytics.

Using the Fault Analytics Custom Mainframe API

The first step is to create the detection application. When detected, it must subsequently call the user-defined event API using the API formats shown previously.

Then linkedit CWFM into the application. To linkedit CWFM into the application, include the lines shown in Figure 5-2 in the link JCL.

Figure 5-2. Linkedit CWFM JCL

```
//INCLIB DD DISP=SHR,DSN=h1q.LOADLIB FM Loadlib
. . . . .
//SYSLIN DD . . . . .
// DD *
        INCLUDE INCLIB(CWFM)
. . . . .
```

Return Codes

After a call is made to the user-defined event API, one of the return codes in Table 5-2 is sent back to the calling program.

Table 5-2. Return Code

Return Code	Message If Debug On	Description
0		Record accepted.
4	CANNOT LOCATE PAB	Fault Analytics is not running.
8	INVALID FLD NAME: <i>nnnnnn</i>	A field unknown to Fault Analytics was present in the CSV record passed.
12	MISSING NULL/SIZE EXCEEDED	The terminating NULL was not found within the first 2048 bytes of the record passed.
16	BAD REC ADDRESS PASSED	Invalid record address.
20	ERROR SEARCHING FOR NULL	NULL not found, record is invalid.
24	ERRORCODE1 REJECTED: <i>cccccc</i>	The error code passed is either a mainframe system, user, or CICS abend code.
28	DUPLICATE FIELD NAME: <i>nnnnnn</i>	A duplicate field has been passed to CWFWM.
32	MISSING REQ FLD: <i>nnnnnnnn</i>	The required field was not passed to CWFWM.
36	DATA LEN ERR: <i>fffffff</i>	The data length of field <i>fffffff</i> is greater than 256 or exceeds the maximum allowed field length.
40	INVALID NAME FIELD	The name part of the field exceeds 24 bytes.
44	INVALID DATE: <i>ddddddddd</i>	The date field is invalid. It must be in the form: MM/DD/YYYY.
48	INVALID Time: <i>ttttttt</i>	The time field is invalid. It must be in the form: HH:MM:SS.
52	INVALID EVENT NAME: <i>nnnnnn</i>	The event name is reserved for Compuware products.
56	INVALID DATA FLD: <i>nnnnnn</i>	The data in field <i>nnnnnn</i> contains non-character data.
60	DATA ALL SPACES FLD: <i>nnnnnn</i>	The data in field <i>nnnnnn</i> contains all spaces.

Coding Examples

General Description

Table 5-3 lists sample source code for C and COBOL—as well as compile, link, and test JCL—that can be found in the Abend-AID installation sample library (CTL).

Table 5-3. Sample Source Code for C and COBOL

Member	Description
CFMAPI	C API sample source program
\$47FACCL	JCL to compile and link CFMAPI (Uses IBM compiler)
\$49FACPI	Sample JCL to run program CFMAPI
COBFMAPI	COBOL API sample source program
\$48FACBL	Compile and link JCL to build COBFMAPI executable
\$50FACBI	JCL to run COBFMAPI

Note: Before proceeding, complete the steps in “Using the Fault Analytics Custom Mainframe API” on page 29 for CFMAPI or COBFMAPI to work.

Input to CFMAPI

Input to CFMAPI consists of the following commands: TSTAPIX and PUBLISH. TSTAPIX causes a canned event (eventname TSTEVENT) to be published. Command TSTAPIX causes a CSV record to be built in memory, with each successive TSTAPIX command appending its parameters to the CSV record. When the record has been built, the PUBLISH command issues a call to CWFMM and the event is published.

Sample Input to CFMAPI:

```
//SYSIN DD *
TSTAPIX EVENTNAME,TESTEV,ERROROCCURREDATDATE,02/19/2014,
TSTAPIX ERROROCCURREDATTIME,09:01:33,PUBLISHERNAME,COBAPI,
TSTAPIX PUBLISHERVERSION,01.01.01,ERRORCODE1,COBERR1,
TSTAPIX JOBNAME,XUSERIDOQ,PROGRAMNAME,COBFMAPI
PUBLISH
```

Input to COBFMAPI

Input to COBFMAPI consists of the command TSTAPIX. Command TSTAPIX causes a CSV record to be built in memory, with each successive TSTAPIX command appending its parameters to the CSV record. The TSTAPIX command for COBFMAPI expects a delimiting tilde (~) at the end of the parameters. When CMDINPUT end-of-file occurs, a call to CWFMM is issued, the event is published, and the program ends.

Sample Input to COBFMAPI:

```
//CMDINPUT DD *
TSTAPIX EventName,TESTEV,ErrorOccurredAtDate,02/19/2014,~
TSTAPIX ErrorOccurredAtTime,09:01:33,PublisherName,COBAPI,~
TSTAPIX PublisherVersion,01.01.01,ErrorCode1,COBERR1,~
TSTAPIX JobName,XUSERIDOQ,ProgramName,COBFMAPI~
```


Appendix A.

Fault Analytics Installation on Linux

This chapter describes the required steps to set up and run the Fault Analytics installation on Linux.

Before Beginning

Before you begin the installation, do the following:

- Install Compuware Enterprise Services (CES). If multiple CES instances are installed, determine which instance to associate with this Fault Analytics installation. Refer to the *Compuware Enterprise Services Installation and Configuration Guide* for installation instructions.
- Install and configure a supported database.
- Determine the Fault Analytics port to be used.
- The installation may require up to 30 MB of space during the installation. When the installation is complete, the files are removed.

Installing Fault Analytics

1. Copy the entire <CDRom drive>:\cpwr\Linux folder in binary mode to a folder of your choice. The folder structure *must* be maintained as it is on the distribution media.
2. Ensure that the `install.bin` file has execute authority. Enter the command `chmod 777 install.sh` to give full access permissions. This file is located at `Disk1\InstData\NoVM`.
3. Log on to Linux and navigate to the folder that contains the `install.bin` file. Issue the following commands to initiate the installation procedure:
 - `export JAVA_HOME=<absolute path to your Java folder>`
 - `export PATH=$JAVA_HOME/bin:$PATH`
 - `./install.bin`
4. Read the **Introduction** panel, then click **Enter**. The **Fault Analytics** panel appears.
5. At the prompt, do one of the following to indicate the absolute path to the Compuware Enterprise Services instance to associate with this installation of Fault Analytics:
 - To accept the default path, which is `/opt/Compuware/CES`, press **Enter**.
 - To change the path, type the path to the Compuware Enterprise Services instance to associate with this installation of Fault Analytics, then press **Enter**.

The **License Agreement** panel appears.
6. Read the License Agreement, then type **Y** to accept the terms of the agreement. The **Fault Analytics** panel appears.
7. At the prompt, type a valid Fault Analytics port number between 1024 and 65535 to be used to transmit messages, then press **Enter**. The **Ready to Install** panel appears.

8. Review the installation information and, if correct, press **Enter**. If incorrect, type **back** and make any necessary revisions. When the installation is finished, the **Installation Complete** panel appears and lists the installation folder, Fault Analytics port, and Fault Analytics URL.

CAUTION:

Before exiting the installer, open the Fault Analytics URL in a browser and bookmark the page so you can easily access Fault Analytics in the future.

9. Press **Enter** to exit the installer.
10. Restart the Compuware Enterprise Services service.

Launching Fault Analytics

1. Fault Analytics is launched via the URL that appeared on the **Install Complete** panel at the end of installation. If this URL was not bookmarked at that time, it can be identified by opening the config.properties file, which is located at *<your path>/CES/data/fa/config* by default.
2. Find the host.name= parameter, for example ABC123456N01.mycompanydomain.corp.
3. In a browser, enter the URL in the following format: `http://<host.name>:<CES port number>/aafa`.

Note: Log files are located in the Compuware Enterprise Services data directory specified during CES installation, which is *<CES install directory>/data/fa/logs* by default. Contact the CES installer to determine the location if necessary.

Appendix B.

Fault Analytics Installation on Windows

This chapter describes the required steps to set up and run the Fault Analytics installation on Windows.

Before Beginning

Before you begin the installation, do the following:

- Install Compuware Enterprise Services (CES). If multiple CES instances are installed, determine which instance to associate with this Fault Analytics installation. Refer to the *Compuware Enterprise Services Installation and Configuration Guide* for installation instructions.
- Install and configure a supported database.
- Determine the Fault Analytics port to be used.
- The installation may require up to 30 MB of space during the installation. When the installation is complete, the files are removed.

Installing Fault Analytics

Do the following:

1. Insert the product media or locate the FTP image. If you have Internet Explorer, the media browser opens. If it does not, run `setup.exe` from the root folder of the media.
Note: If installing from an FTP image, run `setup.exe` from the root of that image.
2. On the media browser, click **Install AAFA for Windows** and then click **Start Install**. The **Introduction** page of the **Fault Analytics** dialog box appears.
3. Read the introduction content and then click **Next** to proceed.
4. If more than one instance of Compuware Enterprise Services exists on the machine the **Select Instance** page appears. From the drop-down list, select which CES instance to associate with this installation, then click **Next**. The **License Agreement** page appears.
5. Accept the terms of the license agreement and click **Next**. The **Configuration** page appears.
6. In the **Fault Analytics port** field, enter a valid port number between 1024 and 65535 to be used to transmit messages and click **Next**. The **Pre-Installation Summary** page appears.
7. Review the installation information and, if correct, click **Install**. If incorrect, click **Previous** and make any necessary revisions. When the installation is finished, the **Install Complete** page appears and lists the installation folder, CES service associated with the installation, Fault Analytics port, and Fault Analytics URL.

CAUTION:

Before exiting the installer, open the Fault Analytics URL in a browser and bookmark the page so you can easily access Fault Analytics in the future.

8. Click **Done** to exit the installer.
9. Restart the Compuware Enterprise Services service.

Launching Fault Analytics

1. Fault Analytics is launched via the URL that appeared on the **Install Complete** dialog box at the end of installation. If this URL was not bookmarked at that time, it can be identified by opening the config.properties file, which is located at C:\ProgramData\Compuware\CES\data\fa\config by default.
2. Find the host.name= parameter, for example ABC123456N01.mycompanydomain.corp.
3. In a browser, enter the URL in the following format: http://<host.name>:<CES port number>/aafa.

Note: Log files are located in the Compuware Enterprise Services data directory specified during CES installation, which is ProgramData\Compuware\<CES data directory>\data\fa\logs by default. Contact the CES installer to determine the location if necessary.

Appendix C.

Fault Analytics Fields

Table C-1 lists Fault Analytics database field names for reports and the equivalent rules field names, as well as the operating systems for which those fields are valid. If a size is specified, that field is a text field.

Table C-1. Fault Analytics Fields

Database Field Name for Report	Rules Field Name	Size
General Event Information		
CALLING_PROGRAM	CallingProgram	32
CPU_TIME	Mainframe: Step or StepCPUTime or TransactionCPUTime	28
DESCRIPTION	JobName+ProgramName+ErrorCode1	255
ELAPSED_TIME	Mainframe: Step or TransactionElapsedTime	28
LINK	Link	255
MACHINE_NAME	SystemID	32
NAME	EventName	16
OBJ_MODULE_COMPILE_TS	ObjectModulecompileDate	mm/dd/yyyy
OS_NAME	OperatingSystemName	16
OS_RELEASE	OperatingSystemRelease	25
PROGRAM_LINK_TS	ProgramLinkDate	mm/dd/yyyy
PROGRAM_LOCATION	ProgramLibrary	255
PROGRAM_NAME	Program Name	32
TS_UTC	ErrorOccurredAtDate+ ErrorOccurredAtTime	mm/dd/yyyy time24
TYPE	EventType	32
USER_ID	JobUserID	32
Publisher Information		
NAME	PublisherName	64
LOCATION	PublisherID	20
RELEASE	PublisherVersion	8
Fault Information		
DDIO_SOURCE_FILE_NAME	SourceFileName	44
DESCRIPTION	Description	255
DNS_NAME	DNSName	64
PORT_NUMBER	PortNumber	5
PRIMARY_FAULT_CODE	ErrorCode1	12
PROGRAM_ADDRESS	ProgramOffset	16

Table C-1. Fault Analytics Fields

Database Field Name for Report	Rules Field Name	Size
REFERENCE_LOCATION	DataSetName	255
REFERENCE_NAME	ReportNumber	32
SECONDARY_FAULT_CODE	ErrorCode2	12
Performance Information		
EVENT_CODE	EventCode	16
MEASUREMENT_NUMBER	MeasurementNumber	4
SAMPLE_COUNT	SampleCount	6
SAMPLE_DATASET_NAME	SampleDatasetName	45
STEP_NUMBER	StepNumber	38
THRESHOLD	Threshold	32
IBM Mainframe Information		
AA_ASSIGNED_TO	Abend-AidAssignedToUserID	10
CPU_ID	CPUID	12
CSECT_COMPILER_ID	CSECTCompilerID	32
CSECT_NAME	CSECTName	16
CSECT_OFFSET	CSECTOffset	8
DUPLICATER_INDICATOR	Duplicate	1
ERROR_SUBCODE	ErrorSubCode	8
HEAP_FREE	LEUserHeapTotalFree	12
HEAP_INCREMENT_SIZE	LEUserHeapIncrementSize	12
HEAP_INITIAL_SIZE	LEUserHeapInitialSize	12
HEAP_SEGMENT_ERRORS	LEUserHeapTotalErrors	8
HEAP_SEGMENTS	LEUserHeapNumberOfSegments	8
HEAP_TOTAL	LEUserHeapTotalAllocated	12
JOB_ACCOUNTING	JobAccounting	144
JOB_APPLICATION_ID	JobApplicationID	8
JOB_CLASS	JobClass	1
JOB_CPU_TIME	JobCPUTime	28
JOB_ELAPSED_TIME	JobElapsedTime	28
JOB_JES_ID	JobJESID	8
JOB_NAME	JobName	8
JOB_PRIORITY	JobPriority	3
JOB_PROGRAMMER_NAME	JobProgrammerName	20
PROCEDURE_NAME	ProcedureName	64
STEPNAME	StepName	16
CICS Information		
NET_NAME	CICSNetName	8
OPERATOR_ID	CICSOperatorID	8
RELEASE	CICSRelease	8

Table C-1. Fault Analytics Fields

Database Field Name for Report	Rules Field Name	Size
TERMINAL_ID	CICSTerminalID	4
TRANSACTION_ID	CICSTransactionID	4
CICS Web Interface (CWI) Information		
ALIAS_TRANSACTION_ID	CWIWebAliasTranID	4
ALIAS_TRANSACTION_PROGRAM	CWIWebAliasTranProgram	8
ANALYZER_PROGRAM	CWIAnalyzerProgram	8
ATTACH_TRANSACTION_ID	CWIWebAttachTranID	4
ATTACH_TRANSACTION_PROGRAM	CWIWebAttachTranProgram	8
BRIDGE_EXIT_PROGRAM	CWIBridgeExitProgram	8
BRIDGE_TRANSACTION_ID	CWIBridgeTranID	4
CLIENT_IP_ADDRESS	CWIClientIPAddress	15
HOST_IP_ADDRESS	CWIHostIPAddress	15
PORT_NUMBER	CWIPortNumber	5
TCPIP_SERVICE_NAME	CWITCPIPServiceName	8
URL	CWISuppliedURL	255
DB2 Information		
AUTHORIZED_ID	DB2AuthorizedID	10
PLAN_PACKAGE	DB2PlanOrPackage	8
RELEASE	DB2Release	8
SQLCODE	DB2SQLCODE	8
SUBSYSTEM	DB2Subsystem	8
IDMS Information		
RELEASE	CA-IDMSRelease	8
STATUS_CODE	CA-IDMSStatusCode	8
SUBSCHEMA	CA-IDMSSubschema	8
VERSION	CA-IDMSCentralVersion	8
IMS Information		
PSB_NAME	IMSPSBName	8
RACF_ID	IMSRACFID	10
RELEASE	IMSRelease	8
SUBSYSTEM	IMSSubsystem	8
TERMINAL_IC	IMSTerminalID	8
TRANSACTION_ID	IMSTransactionID	8
User Defined Information		
FIELD1	UserField01	8
FIELD2	UserField02	8
FIELD3	UserField03	16
FIELD4	UserField04	16
FIELD5	UserField05	16

Table C-1. Fault Analytics Fields

Database Field Name for Report	Rules Field Name	Size
FIELD6	UserField06	16
FIELD7	UserField07	32
FIELD8	UserField08	32
FIELD9	UserField09	32
FIELD10	UserField10	32
FIELD11	UserField11	32
FIELD12	UserField12	32
FIELD13	UserField13	32
FIELD14	UserField14	32
FIELD15	UserField15	32
FIELD16	UserField16	32
FIELD17	UserField17	32
FIELD18	UserField18	64
FIELD19	UserField19	128
FIELD20	UserField20	255

Appendix D.

Fault Analytics Custom API Fields

Table D-1 lists Fault Analytics field names and other related information for use with the custom API.

Note: In the **Data Type** column, the number in parentheses is the size of the field.

Table D-1. Fault Analytics Fields

Field Name	Data Type	Required Field	Fault Mgr. Gen.	Field Name Descriptions
Event Information				
Application	Char (32)	No	Yes	Command line used to start the program
CallingProgram	Char (32)	No		Program that called the faulting program
CPUTime	Number(28)	No		Amount of CPU time (ms) taken by the process or step
Description	Char (255)	No		Description of the event
ElapsedTime	Number(28)	No		Amount of time (ms) the process or step has been running
ErrorOccurredAtDate	Date/Time	Yes		Date stamp when the event occurred Format: mm/dd/yyyy
ErrorOccurredAtTime	Date/Time	Yes		Time stamp (time24) when the event occurred
EventName	Char (16)	Yes		Name given to the event (Must not be: Batch, CICS Transaction, CICS Region, JCL, Log Event, SMF Batch, Strobe Measurement Started, Strobe Warning, Strobe Measurement Completed)
EventType	Char (32)	No		Type of event that occurred (for example, Batch, CICS Transaction)
JobUserID	Char (32)	No		User ID associated with the program that caused the fault
Link	Char (255)	No		Hyperlink (such as a URL) to more event information
ObjectModuleCompileDate	Date	No		Date the faulting program was last compiled
OperatingSystemName	Char (16)	No		Operating system where the program fault occurred
OperatingSystemRelease	Char (25)	No		Operating system release where the program fault occurred
ProgramLibrary	Char (255)	No		Location where the program resides (for example, path or load library)
ProgramLinkDate	Date/Time	No		Timestamp when the program was built Format: mm/dd/yyyy
ProgramName	Char (32)	No		Name of the software program that caused the fault event
SystemID	Char (32)	No		Name of the machine/system ID where the fault occurred

Table D-1. Fault Analytics Fields

Field Name	Data Type	Required Field	Fault Mgr. Gen.	Field Name Descriptions
Publisher Information				
PublisherID	Char(20)	Yes		Network name or IP address assigned to the machine that published the event
PublisherName	Char (32)	Yes		Application name that published the event (for example, Abend-AID)
PublisherVersion	Char (8)	Yes		Software release of the application that published the event
Fault Information				
DataSetName	Char (255)	No		Data file name containing the fault information (i.e., dump)
ErrorCode1	Char (12)	Yes		Error code of the fault
ErrorCode2	Char (12)	No		Secondary error code or sub-code of the fault
ProgramOffset	Char (16)	No		Instruction pointer address (in hexadecimal) where the program fault occurred
ReportNumber	Number(32)	No		Report number identifying the dump
IBM Mainframe Information				
Abend-AIDAssignedToUserID	Char (10)	No		User ID error assigned to by Abend-AID
CPUID	Char (12)	No		ID of the CPU where the fault event occurred
CSECTCompilerID	Char (32)	No		Compiler ID of the control section in error
CSECTName	Char (16)	No		Control section name of the faulting program
CSECTOffset	Char (8)	No		Control section offset into the faulting program
Duplicate	Char (1)	No		Indicator if Abend-AID for CICS region/transaction error is a duplicate (Y/N)
ErrorSubCode	Char (8)	No		Error sub-code of the fault event
JobAccounting	Char (144)	No		Faulting job's accounting code
JobApplicationID	Char (8)	No		Application ID assigned to the job
JobClass	Char (1)	No		Faulting job's class
JobCPUTime	Number (28)	No		CPU time (ms) consumed by the job
JobElapsedTime	Number (28)	No		Total time (ms) since the start of the job
JobESID	Char (8)	No		Faulting job's JES ID
JobName	Char (8)	Yes		Name of the job for the fault event
JobPriority	Char (3)	No		Faulting job's priority
JobProgrammerName	Char (20)	No		Name of the programmer assigned to the job
LEUserHeapIncrementSize	Char (12)	No		Increment size of the user LE heap
LEUserHeapInitialSize	Char (12)	No		Initial size of the user LE (language environment) heap
LEUserHeapNumberOfSegments	Char (8)	No		Number of user LE heap segments allocated
LEUserHeapTotalAllocated	Char (12)	No		Total amount of user LE heap allocated
LEUserHeapTotalErrors	Char (8)	No		Total number of errors in all LE heap segments
LEUserHeapTotalFree	Char (12)	No		Total amount of user LE heap free
ProcedureName	Char (64)	No		Name of the procedure where the fault occurred

Table D-1. Fault Analytics Fields

Field Name	Data Type	Required Field	Fault Mgr. Gen.	Field Name Descriptions
CICS Information				
CICSNetName	Char (8)	No		Transaction server name where the CICS fault occurred
CICSOperatorID	Char (8)	No		Operator ID assigned to the CICS fault event
CICSRelease	Char (8)	No		CICS release where the fault occurred
CICSTerminalID	Char (4)	No		Terminal ID where the fault occurred
CICSTransaction	Char (4)	No		Transaction ID of the fault
CICS Web Interface (CWI) Information				
CWIANalyzerProgram	Char (8)	No		Analyzer program
CWIBridgeExitProgram	Char (8)	No		Bridge exit program
CWIBridgeTranID	Char (4)	No		Bridge transaction
CWIClientIPAddress	Char (15)	No		IP address of the client
CWIHostIPAddress	Char (15)	No		IP address of the host
CWIPortNumber	Char (5)	No		Port number (1-32767)
CWISuppliedURL	Char (255)	No		Supplied URL
CWITCPIPServiceName	Char (8)	No		TCP/IP service name
CWIWebAliasTranID	Char (4)	No		Web alias transaction ID
CWIWebAliasTranProgram	Char (8)	No		Web alias transaction program
CWIWebAttachTranID	Char (4)	No		Web attach transaction ID
CWIWebAttachTranProgram	Char (8)	No		Web attach transaction program
DB2 Information				
DB2AuthorizedID	Char (10)	No		Authorized ID associated with the fault event
DB2PlanOrPackage	Char (8)	No		DB2 plan or package used
DB2Release	Char (8)	No		DB2 release where the fault occurred
DB2SQLCODE	Char (8)	No		SQL code of the DB2 fault event
DB2Subsystem	Char (8)	No		DB2 subsystem where the fault occurred
IDMS Information				
IDMSCentralVersion	Char (8)	No		Central version of IDMS
CA-IDMSRelease	Char (8)	No		IDMS release where the fault occurred
CA-IDMSStatusCode	Char (8)	No		IDMS status code of the fault event
CA-IDMSSubschema	Char (8)	No		Sub-schema associated with the fault event
IMS Information				
IMSPSBName	Char (8)	No		Program Specification Block (PSB) name of the fault event
IMSRelease	Char (8)	No		IMS release where the fault occurred
IMSSubsystem	Char (8)	No		IMS subsystem where the fault occurred
IMSTerminalID	Char (8)	No		Terminal ID where the fault occurred (not currently used)
User Defined				

Table D-1. Fault Analytics Fields

Field Name	Data Type	Required Field	Fault Mgr. Gen.	Field Name Descriptions
UserField01	Char (8)	No		User-defined Data Field #1
UserField02	Char (8)	No		User-defined Data Field #2
UserField03	Char (16)	No		User-defined Data Field #3
UserField04	Char (16)	No		User-defined Data Field #4
UserField05	Char (16)	No		User-defined Data Field #5
UserField06	Char (16)	No		User-defined Data Field #6
UserField07	Char (32)	No		User-defined Data Field #7
UserField08	Char (32)	No		User-defined Data Field #8
UserField09	Char (32)	No		User-defined Data Field #9
UserField10	Char (32)	No		User-defined Data Field #10
UserField11	Char (32)	No		User-defined Data Field #11
UserField12	Char (32)	No		User-defined Data Field #12
UserField13	Char (32)	No		User-defined Data Field #13
UserField14	Char (32)	No		User-defined Data Field #14
UserField15	Char (32)	No		User-defined Data Field #15
UserField16	Char (32)	No		User-defined Data Field #16
UserField17	Char (32)	No		User-defined Data Field #17
UserField18	Char (64)	No		User-defined Data Field #18
UserField19	Char (128)	No		User-defined Data Field #19
UserField20	Char (255)	No		User-defined Data Field #20
Strobe Information				
EventCode	Char (16)	No		Strobe event code
MeasurementNumber	Char (4)	No		Strobe measurement number that has exceeded threshold
SampleCount	Char (6)	No		Number of samples in the Strobe measurement
SampleDatasetName	Char (45)	No		Dataset name of the Strobe sample
StepNumber	Number	No		Job step number for the Strobe event

Appendix E.

Using Fault Analytics with Abend-AID

Abend-AID

If you want to use the Abend-AID Viewer's URLPUB feature, refer to the Abend-AID installation guide.

Abend-AID for CICS

Abend-AID for CICS must be enabled before it can provide Fault Analytics transaction and/or region fault data. To enable Abend-AID for CICS, set the parameter **MANAGEMENT_REPORTING** to **YES** in both the transaction dump capture address space (TDCAS) and the viewing server address space.

You can find the parameters required to support the Fault Analytics Web Interface Tool in the Abend-AID installation sample (CTL) library.

Note: After you add the parameters, you need to recycle both the TDCAS and viewing server.

If you need more information about TDCAS or the viewing server, refer to the Abend-AID for CICS installation guide. Compuware manuals can be found online, in PDF format, on Compuware's FrontLine Customer Support site: <http://go.compuware.com>.

Appendix F.

Fault Analytics Export and Load Process

This appendix provides procedures to convert Fault Analytics data from a SQL Server database to a DB2 database. The following procedure extracts the data from each SQL Server table into flat files, transfers those flat files to z/OS, and loads them into the Fault Analytics DB2 tables.

1. **ARCHIVE:** If your database is large, consider archiving some of the data so that the files created for the extract and load process are not unreasonably large.
2. **STOP COLLECTION:** Before extracting any data, stop all Fault Analytics components. Data should not be collected during the extract process.
3. **ENABLE EXPORT:** The export process, uses the `bcp` (bulk copy) command with `xp_cmdshell`. The `xp_cmdshell` may need to be enabled. This can be accomplished by using `sp_configure`. Execute the following to enable `xp_cmdshell`.
4. **CUSTOMIZE EXPORT:** See the `bcpcommand.txt` file. Edit the file to reflect your environment.
 - a. Replace `[database]` with the name of your database.
 - b. Replace `[schema]` with the name of your schema.
 - c. Replace `[path]` with the directory path where the exported data will reside. Insure that the userid performing the extract has write permission to the directory specified.
 - d. Replace `[server]` with the server name.
 - e. Replace `[loginid]` with userid for the server.
 - f. Replace `[password]` with the password for the server userid.
5. **EXPORT:** In SQL Server Management Studio, after connecting to the server that contains the Fault Analytics database,
 - a. Open a new query window.
 - b. Copy the contents of the `bcpcommand.txt` file into the query window and execute the query. This issues a `bcp` command for each of the tables in the database and exports the data in the table to a delimited flat file with the naming convention `tablename.txt`. The results from these commands will be needed in the next step.
6. **RECORD:** In the table shown below, record the number of records exported for use in the DB2 load utility job. The tables will be loaded in the order listed to ensure enforcement of referential integrity.

Table F-1. Record the number of exported records

Migration Order	Table Name	Exported (Y/N)	Number of Records Exported
	CWAF_CONFIGURATION	N	0
	CWAF_AUTOINCREMENT	N	0
1	CWAF_PUBLISHER	Y	
2	CWAF_EVENT	Y	
3	CWAF_FAULT	Y	
4	CWAF_FAULT_CICS	Y	
5	CWAF_FAULT_CWI	Y	
6	CWAF_FAULT_DB2	Y	
7	CWAF_FAULT_IDMS	Y	
8	CWAF_FAULT_IMS	Y	
9	CWAF_FAULT_MQSERIES	Y	
10	CWAF_FAULT_OS390	Y	
11	CWAF_CUSTOM†	Y	
12	CWAF_SCHED_REPORTS	Y	
13	CWAF_PERFORMANCE	Y	
14	CWAF_LMS	Y	
15	CWAF_USER_DEFINED	Y	
	CWAF_TIME_ZONE_INFO	N	0
16	CWAF_OPEN_FILES	Y	
17	CWAF_OPEN_FILES_MAP	Y	
18	CWAF_OPEN_FILES_HOLD	Y	
19	CWAF_DB2_HOLD	Y	
20	CWAF_DB2_TABLES	Y	
21	CWAF_DB2_TABLES_MAP	Y	
22	CWAF_DB2_TABLES_HOLD	Y	
23	CWAF_ROOT_CAUSE	Y	
24	CWAF_ROOT_CAUSE_HOLD	Y	
25	CWAF_AA_LINKS	Y	
	CWAF_ORDER_BY_CONSTANTS	N	0
	CWAF_USERS	N	0
	CWAF_ROLES	N	0
26	CWAF_EMAIL_CONFIG	Y	
27	CWAF_DISTLISTS	Y	
28	CWAF_DISTLIST_ADDR	Y	
29	CWAF_NOTES	Y	
	CWAF_USERROLES	N	0
30	CWAF_REPORT_DEFAULTS	Y	
	CWAF_FAULT_WIN	N	0
	CWAF_FAULT_UNIX	N	0

†. CWAF_CUSTOM can contain customized column names.

7. **CREATE DATASETS** (*optional*): If you prefer to pre-allocate the datasets that will hold the extracted data for the DB2 load job,
 - a. Log on to TSO, modify, and run the CRTDSN job. The SPACE allocations will need to be reviewed and modified to reflect the size of the flat files that will be transferred.
 - b. Change the value USERID to the high level qualifier of the datasets that will receive the extracted data. (This step may not be needed if the transfer method allocates the datasets.)

8. **TRANSFER TO DATASETS:** Transfer the flat files to the datasets created in the previous step.
9. **INSTALL:** Before running the DB2 Load utility, the Fault Analytics database and tables should be created. The load job assumes that the data will be loaded into empty tables. Some tables are populated as part of the install and are not loaded.
10. **CUSTOMIZE LOAD:** Edit the LOADDATA JCL.
 - a. Modify the Jobcard.
 - b. Change the value SQLID to the SQLID that created the Fault Analytics tables.
 - c. Change the value USERID to the high level qualifier of the datasets that contain the exported data specified in the CREATE DATASETS step.
 - d. Change the value ?? for each table to an integer value that is an approximation of the number of records being loaded. This information is provided in the EXPORT step and should have been recorded in the table of the RECORD step. This number will help the load perform efficiently.
 - e. Change the value of the LIB= parameter to the DB2 dataset that contains the DB2 utility module DSNUTILB.
 - f. Change the value of the SYSTEM= parameter to the DB2 subsystem ID on which the tables have been defined.
 - g. Review the LOAD statement for the table CWAF_CUSTOM. Modify the list of columns and types to reflect customization, if necessary.
11. **LOAD:** Submit the LOADDATA JCL to load the DB2 tables with the exported SQL Server data. The userid submitting the JCL should have the proper DB2 authority. See the *DB2 Utilities Guide* for more information.
12. **COPY:** Make an image copy of the tablespace. The load job performs a separate load for each Fault Analytics table. It will place the tablespace in check pending. This will require that an image copy take place prior to turning on Fault Analytics to record events. See the *DB2 Utilities Guide* for more information.

Glossary

Abend. An abnormal end or termination of a computer program because of a problem with how it is running, usually because of application errors.

Abend-AID. A product developed by Compuware that automatically collects program and environmental failure information, analyzes the information, and presents diagnostics and supporting data. One of the available agents or publishing agents, Abend-AID integrates with Abend-AID Fault Analytics to send its collected information to the z/OS Event Processor.

Abend-AID for CICS. A product developed by Compuware that automatically collects program and environmental failure information, analyzes the information, and presents diagnostics and supporting data. One of the available agents or publishing agents, Abend-AID for CICS integrates with Abend-AID Fault Analytics to send its collected information to the z/OS Event Processor.

API: Application Programming Interface. Fault Analytics allows customized applications to detect their own events and publish them to Fault Analytics through an API available on the mainframe platform.

BIRT. Business Intelligence and Reporting Tools. A reporting server and design tool from Eclipse that is supported by Fault Analytics to view, create, or modify Fault Analytics reports.

Compuware Enterprise Services. A core component of Fault Analytics that is responsible for receiving events, storing the event information into a database, and processing these events according to user-defined rules.

Data Repository. A repository that stores event information that Fault Analytics collects.

DBMS. Database Management System used to store Fault Analytics event data. Fault Analytics supports DB2 and Microsoft SQL Server databases.

E-mail. A method of electronically sending messages. Fault Analytics provides a notification event handler that integrates with an SMTP (Simple Mail Transfer Protocol) service to send e-mail or pass messages to individuals or applications. This event handler allows Fault Analytics to integrate with applications like help desk tools that typically accept e-mail messages that can be used to automatically generate trouble tickets. In addition, the Management Reporting tool provides the ability to e-mail a link to specified reports on an on-demand or regularly scheduled basis.

FAAdministrator. A user role for the person who is responsible for configuring and administering Fault Analytics. Those who administer Fault Analytics have access to tasks such as configuring the event handlers and creating rules, activating and de-activating event routers, and so on. Only users with this role are authorized to access the Administration function.

FAReport Scheduler. A user role for the person who is responsible for scheduling reports to be e-mailed on a periodic basis. The screen for configuring scheduled reports is accessible on a permission-only basis to users with this role. A link to the report will be e-mailed.

Fault Analytics Collector. A core component of Fault Analytics that gathers failure events from a number of sources that include the operating system, Abend-AID, and user events.

Fault Analytics Server. Obsolete. Refer to Compuware Enterprise Services.

Fault Analytics Viewer. *Obsolete* (is now simply *Fault Analytics* (or *Fault Analytics UI*) a web-based tool used to view reports and centrally administer the various machines in the Fault Analytics network. Using this tool, you can select and view a report, configure event handlers, define rules, and manage event routers.

FAUser. A general user role available to those users who only need to view management reports. In the Management Reports function, users can view Fault Analytics reports based on the historical event data contained in the database. Everyone, by default, has access to this information as a user. The reports themselves, however, can be assigned user-defined roles, roles that can be created and defined to view specific reports. If roles are defined to view specific reports, the user must have the proper role assigned for it to be shown in that user's report tree. Users without the proper report role never see the report within their report tree, so it is inaccessible.

FrontLine Web Site. A customer support Web site for Compuware products (<http://frontline.compuware.com>). FrontLine provides you with fast access to critical information about Compuware products. You can read or download documentation and product fixes or directly e-mail Compuware with questions or comments. If you have questions about using Fault Analytics, you can check the frequently asked questions on FrontLine before calling Compuware Customer Support. To access FrontLine, first register and obtain a password. After you login to FrontLine, select Fault Analytics from the list of products.

JCL. Job Control Language. A scripting language that controls the execution of programs on mainframes.

JCL Error Detection: An optional component of the Event Processor that detects mainframe JCL errors and publishes them to Fault Analytics. If the job entry sub-system is JES2, the specific error message is also captured. Using this detection agent, a user can specify additional error messages to be captured and sent to Fault Analytics.

Mainframe API: A component installed on a mainframe system that allows a user program to invoke an API to transmit custom fault events to Fault Analytics.

Management Reports. A component of the Fault Analytics Web Interface Tool that uses BIRT (Business Intelligence and Reporting Tools) from Eclipse for displaying reports. User-defined Management Reports provide access to desired event information from ODBC-compliant databases. Fault Analytics collects fault information from Compuware's Abend-AID and Abend-AID for CICS products and SMF utility. Using the viewer, a user can view these reports. There are several types of Management Reports such as, summary charts, summary reports, detail reports, grouped detail reports, and selection reports.

ODBC. Open Database Connectivity. A standard database access method that makes it possible for applications to interact with a database, regardless of the database management system (DBMS) used. Both the application and DBMS used within Fault Analytics must be ODBC compliant.

Rules Processing: Allows events to be defined to perform specific tasks.

SMF Accounting Routine Exit. An MVS system exit that collects information about jobs and programs that have abended or which fit the criteria for capturing jobs ending with non-zero return codes. Information not captured by the other Abend-AID products is collected by this exit and is sent to Fault Analytics for displaying. This exit is an optional feature of the Event Processor.

SMTP. Simple Mail Transfer Protocol. A protocol for sending e-mail messages between servers. E-mail allows Fault Analytics to integrate with any application that accepts e-mail, such as help desk tools, for trouble ticket generation.

XML. Extensible Markup Language. A specification designed for web documents that allows the creation of customized tags. This enables data to be defined, transmitted, validated, and interpreted between applications.

Index

Special Characters

\$40FACJB
 FAC batch job, 21
 \$41FACPR
 FAC procedure, 21
 \$46FAVER
 sample report, 20
 verifying the installation, 20

A

Abend-AID
 using the URLPUB feature, 45
 Abend-AID for CICS
 enabling for use with Fault Analytics, 45
 AFDEBUG JCL statement, 28
 API
 call processing, 29
 coding examples, 30
 custom fields, 41
 mainframe, 27, 29
 return codes, 30
 API format
 Assembler, 28
 C and C++, 28
 COBOL, 28
 debugging, 28
 PL/I, 28

C

C and C++
 API format, 28
 CFMAPI member, 31
 COBFMAPI member, 31
 COBOL API format, 28
 components
 communication between, 10
 of Fault Analytics, 9
 pictorial overview, 9
 custom API
 field names, 41
 mainframe, 27, 29
 CWFM module, 27

D

database
 field names, 37
 debugging
 API format, 28

E

event fields
 required, 27

F

FAC
 See Fault Analytics Collector
 Fault Analytics Collector, 17
 connection parameters, 21
 datasets, 17
 defined, 9
 emergency termination, 24
 instances, 17
 migrating to this release, 19
 operator commands, 23
 running, 21
 runs on each LPAR, 9
 shutdown command, 24
 SYSIN parameters, 22
 system security, 19
 verifying the installation, 20
 field names
 custom API, 41
 database, 37
 FrontLine customer support, 6

I

Installation Planning Worksheet, 10

O

online documentation, 5
 online help, 6

P

PL/I API format, 28

R

rules
 field names, 37

W

Windows
 installing Fault Analytics, 35

