

Message Flood Automation Function was made available through SPE OA17514 for z/OS® R6, R7, and R8

## Message Flood Automation

### IBM Education Assistant “Getting started with Message Flood Automation” SPE OA17514 for z/OS R6/R7/R8

W. Kevin Kelley  
[kkelley@us.ibm.com](mailto:kkelley@us.ibm.com)  
(845) 435-6697  
8+295-6697

February 08, 2007

This presentation will discuss Message Flood Automation -- a new component of z/OS Console support -- that was made available as a Small Programming Enhancement at the end of November 2006. The Message Flood Automation Small Programming Enhancement is available for z/OS release 6, release 7 and release 8.

## Agenda

### Message Flood Automation

- Problem statement and solution
- Installation, loading and activating
- Customization and tuning
  - ▶ Turning it on and off
  - ▶ Displaying your policy
- Publications

This presentation will discuss the problem that Message Flood Automation was designed to solve, and some of the considerations for installing and customizing Message Flood Automation.

## The problem

- Malfunctioning I/O devices and errant or malicious programs can generate very large volumes of messages in a very short amount of time.
- These very large volumes of messages can overwhelm system resources, leading to a loss of operator control, and sometimes a system outage.

The problem that Message Flood Automation is attempting to solve has been present since the earliest days of the operating system. It is most often caused by malfunctioning I/O devices that flood the system with a very large number of error messages, often within a matter of seconds. Message floods can also be caused unintentionally by improperly designed programs that get stuck in a message loop – and by malicious programs that deliberately generate large volumes of messages in an attempt to cause a system outage.

Message floods are a concern because they monopolize system resources and prevent the operator – and automation -- from seeing and reacting to other system messages in a timely manner. In some cases, message floods so monopolize system resources that resource shortages can develop and a system outage occurs.

## Attributes of the solution

- Message Flood Automation is able to react to potential message flooding situations in a matter of 10's or 100's of messages (as specified in installation policy)
  - ▶ Before buffers have begun to fill
  - ▶ Before console queues have begun to build
  - ▶ Before console message rates have begun to skyrocket

Unlike more traditional approaches, which deal with the consequences of a message flood, Message Flood Automation is designed to detect and react to a message flood *before* those consequences have had an opportunity to occur. Installation policy allows you to tailor Message Flood Automation to your individual environments. You can decide how quickly you want Message Flood Automation to react to a potential message flood, and the actions it is to take should a message flood occur. Depending on the policy that you establish, Message Flood Automation can prevent message buffers from filling, console queues from becoming overly long, and console displays from becoming unreadable.

## Attributes of the solution (cont.)

- Use of Message Flood Automation does not result in large residual buffers or queues of messages that must be “worked down” to return to normal operation
  - ▶ No need to take manual or automated action against each console to discard unwanted messages
  - ▶ Handles messages to all consoles, including EMCS
- Use of Message Flood Automation does not result in “throwing the baby out with the bathwater”
  - ▶ Targets the offending message
  - ▶ Targets the offending job or started task

Because Message Flood Automation deals with the flood messages as they are being generated, large numbers of flood messages do not have the opportunity to accumulate in message buffers or in various queues. This means that there is no need to take action, either automated or manual, to “flush” these unwanted messages from buffers or queues. In the past, this was one of the more aggravating aspects of dealing with message floods: the need to flush the unwanted messages from the queues of *each* console. Furthermore, Message Flood Automation deals with the messages going to *all* of the various consoles, including the extended consoles used by SDSF and by automation products such as Tivoli NetView and Tivoli System Automation.

In the past, the only mechanism available to remove unwanted messages from a console queue was the CONTROL Q command. While CONTROL Q was a very effective way of removing messages, it also was very indiscriminate and many users of the command found the unwanted messages were gone, but so were some other messages that were critical in some way to the continued operation of the system. Use of the CONTROL Q command was very much a double-edged sword. Message Flood Automation can potentially eliminate the need to issue the CONTROL Q command by preventing flood messages from ever reaching a console. Message Flood Automation policy allows you to have policy that targets individual messages or individual jobs or started tasks. By targeting specific messages and units of work, Message Flood Automation is designed to minimize the disruption to other work in the system.

## Installing Message Flood Automation

- Use of Message Flood Automation is *optional*
- You must take overt action
  - ▶ To install Message Flood Automation
  - ▶ To activate Message Flood Automation
- Based on the Message Flood Automation function that has been distributed with the Geographically Dispersed Parallel Sysplex (GDPS®) high availability product since 2003.

Because of the way that Message Flood Automation has been implemented, its use is optional.

You must take overt actions to install and activate Message Flood Automation.

Message Flood Automation is based on very similar function that has been distributed with the Geographically Dispersed Parallel Sysplex high availability product since 2003. It has been installed and successfully used by some of the very largest GDPS customers.

## Installation

- Message Flood Automation occupies two system exit points:
  - ▶ IEAVMXIT general message exit
  - ▶ System command exit
- “Some assembly required”
  - ▶ Message Flood Automation does not ship a part named IEAVMXIT, but the exit must have that name
  - ▶ Message Flood Automation provides two sample programs which can become the IEAVMXIT front-end
  - ▶ You must select one of them, assemble it, and link it with the Message Flood Automation CNZZVMXT load module

The installation of Message Flood Automation is more difficult than the usual installation of a new system component because Message Flood Automation occupies two system exit points: the IEAVMXIT general message exit, and the system command exit.

Message Flood Automation does not ship a part named IEAVMXIT because this might cause the Message Flood Automation function to be inadvertently installed and activated when you do not want it to be. Instead, you must select one of the two sample programs provided in SYS1.SAMPLIB, assemble it, and link it to the Message Flood Automation CNZZVMXT load module. Sample job control language to do this is provided in the Message Flood Automation User's Guide and in the APAR.



## Installation (cont.)

- If you already have a pre-existing IEAVMXIT, you will need to either:
  - ▶ Rework your IEAVMXIT to invoke Message Flood Automation from it, or
  - ▶ Insert logic from your IEAVMXIT into one of the sample programs provided by Message Flood Automation
  - ▶ The User's Guide and the sample programs show you how to do this
- The level of the IEAVMXIT message exit and the level of the Command Exit must be the same.
  - ▶ Primarily a consideration for GDPS customers that are migrating from a previous level of Message Flood Automation

If you already have a pre-existing IEAVMXIT, you will need to do the following: either put the invocation of Message Flood Automation into your IEAVMXIT (using sample program CNZZVXT2 as an example of how to do this) or fit your existing IEAVMXIT logic into the sample program CNZZVXT2 and rename it IEAVMXIT.

The level of the message exit and the level of the command exit **MUST** be the same since they both map the same shared data area. There is code in both to ensure that they are at the same level. This is primarily a concern for GDPS customers who are migrating from a GDPS level of Message Flood Automation to the z/OS level of Message Flood Automation.

## Installation (cont.)

- You must make sure that your CONSOLxx PARMLIB member INIT statements do not specify UEXIT(N)
  - ▶ UEXIT(Y) is the default; it doesn't hurt to specify
  - ▶ coding UEXIT(Y) causes the Message Flood Automation message exit to be loaded automatically at IPL and when a CONTROL M,UEXIT command is processed
- You must add a .CMD USEREXIT(CNZZCMXT) entry to your MPFLSTxx PARMLIB member (or add CNZZCMXT to an entry that you already have)
  - ▶ the .CMD entry causes the Message Flood Automation command exit to be automatically loaded at IPL and when a SET MPF= command is processed

Installing the Message Flood Automation code into the system libraries is not sufficient. You must also make changes to two PARMLIB members.

You must make sure that the INIT statement in the CONSOLxx PARMLIB member that you use does not have UEXIT NO specified. UEXIT YES is the default and IBM recommends that you explicitly code UEXIT YES in order to cause the automatic loading of the Message Flood Automation message exit at IPL and when CONTROL M UEXIT commands are processed.

You must also add a dot CMD USEREXIT entry to the MPFLSTxx PARMLIB member that you use. The dot CMD entry will cause the Message Flood Automation command exit to be automatically loaded at IPL and whenever a SET MPF= command is processed.

## Loading and activating

- Loading and activating Message Flood Automation can be completely dynamic:
  - ▶ Use a LLA refresh command to load the Message Flood Automation code
  - ▶ Use a K M,UEXIT=Y command to enable the Message Flood Automation message exit
  - ▶ Use a SET MPF= command to reload MPFLSTxx and cause the .CMD entry to be processed, loading the Message Flood Automation command exit

The Message Flood Automation code can be loaded dynamically; there is no need to do an IPL. If you use the Link List Look aside facility, you will need to perform a Link List Lookaside refresh to bring the Message Flood Automation code into storage. (If you manipulate the libraries that are in your link list concatenation, consult the Message Flood Automation User's Guide for details on using the SETPROG command).

Once the Message Flood Automation code has been placed into storage, a CONTROL M UEXIT command can be used to actually load and enable the Message Flood Automation message exit. Likewise, a SET MPF= command can be used to load and activate the Message Flood Automation command exit. Since the command exit uses data structures that are set-up by the message exit, it is best to load and activate the command exit after loading and enabling the message exit. The reverse sequence will also work, but the command exit will be unable to perform any Message Flood Automation commands until the message exit has been loaded and creates the shared data structures.

## Customizing and tuning

- You must prepare one or more MSGFLDxx PARMLIB members to contain the Message Flood Automation policy; a usable sample MSGFLD00 is provided in SYS1.SAMPLIB
  - ▶ You should customize the message threshold values provided in MSGFLD00 before enabling Message Flood Automation on your system
  - ▶ Recommended message threshold values can be determined using the Message Rate Monitoring function
    - SETMF MONITORON
    - SETMF MONITOROFF
    - DISPLAY MSGFLD,MSGRATE

You should prepare one or more MSGFLDxx members of PARMLIB. These MSGFLDxx members of PARMLIB will contain your installation policy for Message Flood Automation.

IBM provides a usable, sample MSGFLDxx member of PARMLIB in the SYS1.SAMPLIB library. However, IBM recommends that you customize this sample MSGFLD00 member before you attempt to use it.

It is very important that the message rate thresholds be properly set in the Message Flood Automation policy. You can determine the values that are appropriate for your system by using the Message Flood Automation Message Rate Monitoring function which can be run without enabling any other Message Flood Automation function.

You should run the Message Rate Monitoring function on the system that you intend to run Message Flood Automation on – not on a test system. It is also very important that you obtain a representative sample. IBM recommends that you run the Message Rate Monitoring function for a 24 hour period that encompasses some of your busiest processing time.

You enable the Message Rate Monitoring function using the SETMF MONITORON command. When you have obtained a representative sample, Message Rate Monitoring should be turned off using the SETMF MONITOROFF command.

The DISPLAY MSGFLD message rate command will display a graph of the message rates that were observed and recommendations for setting the message threshold values.

You should run Message Rate Monitoring periodically (or whenever your processing load increases) to ensure that the message threshold values that you are using are still valid.

The next chart will discuss setting the various message threshold values.

## Tuning MSGFLD00

- Use the “suggested threshold” to set the REGULAR message MSGTHRESH value
  - ▶ Set INTVLTIME=1
  - ▶ JOBTHRESH must be < MSGTHRESH
    - Good “Rule of Thumb” is 20-30% of MSGTHRESH value
- Set ACTION message MSGTHRESH < REGULAR MSGTHRESH; ACTION messages much less frequent than other messages
  - ▶ Set INTVLTIME=1
  - ▶ JOBTHRESH must be < MSGTHRESH
    - Good “Rule of Thumb” is 20-30% of MSGTHRESH value
- Set SPECIFIC message MSGTHRESH < REGULAR MSGTHRESH; SPECIFIC messages typically less frequent than other messages
  - ▶ Set INTVLTIME=1
  - ▶ MSGLIMIT must be < MSGTHRESH
    - Good “Rule of Thumb” is 10-15% of MSGTHRESH value

You should set the REGULAR message MSGTHRESH value from the “suggested threshold” values provided at the bottom of the Message Rate Monitoring graph. IBM recommends using the 99% threshold value. This means that in the sample, 99% of the messages occurred at a message rate lower than the threshold value. Use of this threshold value will ensure that Message Flood Automation is not taking action unnecessarily.

The interval time value should be set to one for all of the message types. Setting the value to one will allow Message Flood Automation to be responsive without entailing undue overhead.

For the REGULAR and ACTION message types, the job threshold must always be less than the overall message threshold. These relationships are checked and will cause warning messages to be produced when the policy is loaded or when the various values are displayed by operator command. A good value for the job threshold is 20-30% of the respective overall message threshold.

For the SPECIFIC message type, the message limit must be less than the overall message threshold. This relationship is checked and will cause a warning message to be produced when the policy is loaded or when the various values are displayed by operator command. A good value for the message threshold is 10-15% of the respective overall message threshold.

The ACTION and SPECIFIC MSGTHRESH values should be set lower than the REGULAR MSGTHRESH value because the messages in these classifications typically occur much less frequently than more common messages.

All of this is explained in considerably greater detail in the Message Flood Automation User’s Guide.

## Turning it on

- Message Flood Automation command processing becomes active as soon as the Message Flood Automation message exit and command exits are loaded
  - ▶ automatically at IPL
  - ▶ when a SET MPF= command is processed
- Message Flood Automation message processing is loaded and enabled
  - ▶ automatically at IPL
  - ▶ when a CONTROL M,UEXIT= command is processed
- Message Flood Automation message processing is **INACTIVE** until you explicitly turn it ON

Message Flood Automation command processing becomes active as soon as the message exit and command exits are loaded and enabled. This will occur automatically during an IPL and whenever a SET MPF= command is processed that has a dot CMD entry for the Message Flood Automation command exit.

Message Flood Automation message processing will be loaded and enabled automatically at IPL and whenever a CONTROL M UEXIT YES command is processed. While Message Flood Automation message processing is automatically loaded and enabled, no message processing will occur until it is explicitly turned-on by operator command.

Note that Message Flood Automation can be affected by SET MPF= commands that have nothing to do with Message Flood Automation processing. For example, if a SET MPF= command loads an MPFLSTxx member that does not have the .CMD statement for loading the command exit and Message Flood Automation was previously active, Message Flood Automation command processing will be deactivated. (The symptom for this is that none of the Message Flood Automation commands will be recognized).

## Turning it on or off

- Load your Message Flood Automation policy:
  - ▶ SET MSGFLD=00
- Turn Message Flood Automation message processing ON
  - ▶ SETMF ON
- Turn Message Flood Automation message processing OFF
  - ▶ SETMF OFF
- Audit trail of activity provided in SYSLOG or OPERLOG (or both)

Once the Message Flood Automation code has been installed in the system libraries and the various PARMLIB members have been customized, Message Flood Automation is simple to operate.

A SET MSGFLD command can be used at any time to load or replace your Message Flood Automation policy.

Message Flood Automation message processing is easily turned-on using the SETMF ON command and easily turned-off using the SETMF OFF command. Note that you cannot automatically turn Message Flood Automation on by issuing the SETMF ON command from the COMMNDxx member of PARMLIB.

An audit trail of Message Flood Automation activity is provided in the SYSLOG or OPERLOG or both. You should review the messages that Message Flood Automation issues to make sure that it is behaving as you want it to. If you observe that Message Flood Automation is being invoked frequently, you may want to raise the message thresholds that you have specified in your policy.

## Displaying your policy

- Display your Message Flood Automation policy:
  - ▶ DISPLAY MSGFLD,PARAMETERS
  - ▶ DISPLAY MSGFLD,DEFAULTS
  - ▶ DISPLAY MSGFLD,JOBS
  - ▶ DISPLAY MSGFLD,MSGS
- Dynamically modify your Message Flood Automation policy:
  - ▶ SETMF command

Almost all of the currently active Message Flood Automation policy can be displayed using the DISPLAY MSGFLD command.

The PARAMETERS option will display in tabular form all of the threshold values and all of the time periods over which they are evaluated.

The DEFAULTS option will display in tabular form all of the default actions that will be taken for each of the three message classes: SPECIFIC messages, ACTION messages and REGULAR messages.

The JOBS option will display in tabular form the names of all of the jobs that will have unique actions taken for them.

The MSGS option will display in tabular form the identifiers of all of the messages that will have unique actions taken for them.

Almost all of the Message Flood Automation policy can be modified, when necessary, by the operator using the SETMF command. You should consult the Message Flood Automation User's Guide for details of the SETMF command syntax. Because changes made by the SETMF command will persist only until the next IPL, IBM recommends that changes that you want to be permanent be made in your MSGFLDxx PARMLIB member. You may find the SETMF command useful during Message Flood Automation testing and during emergency situations.

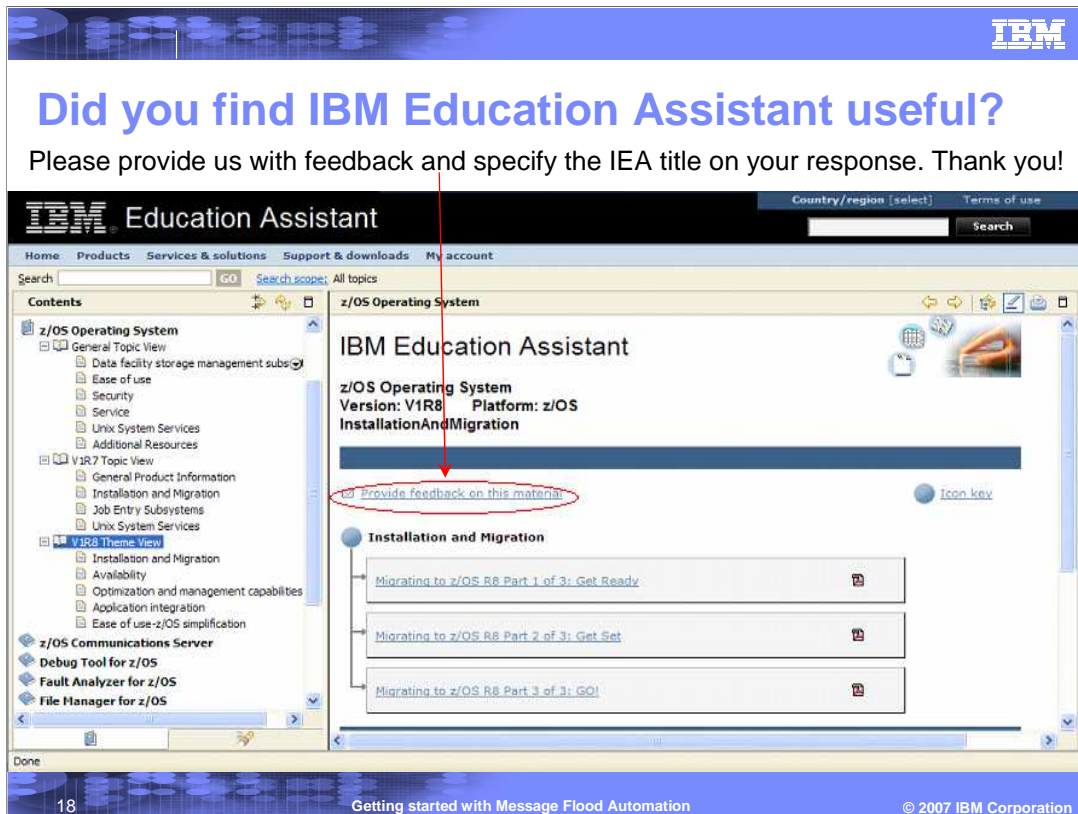


## Message Flood Automation: Publications

- Message Flood Automation User's Guide – PDF document available for download from the z/OS Downloads site:

<http://publibz.boulder.ibm.com/zoslib/pdf/mfaguide.pdf>

The Message Flood Automation User's Guide is the primary documentation for the Message Flood Automation facility. It explains in considerably more detail all of the topics covered in this presentation. It also contains an explanation of all of the messages issued by Message Flood Automation.



In your feedback to IBM please answer the following three questions:

1. How helpful was this IBM Education Assistant presentation? Please give it a rating from 1 to 5 where 1 = very helpful and 5 = not at all helpful
2. Did this presentation save you a service call to IBM? Yes or No
3. If there are any other topics would you like see covered in IBM Education Assistant materials for z/OS, what are they? \_\_\_\_\_

## Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

GDPS          IBM          z/OS

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2007. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.