



Tivoli Workload Scheduler for z/OS

Data set triggering - System Management Facility setup

Tivoli. software



© 2009 IBM Corporation
Updated August 13, 2009

Tivoli Workload Scheduler for z/OS® data set triggering System Management Facility setup.

This training module shows you how to set up and configure System Management Facility (SMF) for data set triggering. Data set triggering is a feature of Tivoli Workload Scheduler for z/OS. Several of the topics covered in this training also apply to the other SMF and JES exit routines used by Tivoli Workload Scheduler for z/OS. Before you begin this module, review the *Data Set Triggering Overview* training module.

SMF setup overview

Tivoli Workload Scheduler for z/OS

- Uses the SMF exit **IEFU83** to track data set close events in the z/OS system
 - ▶ Provides this exit as a member of the sample library SEQQSAMP
 - ▶ Uses SYS1.PARMLIB to implement and enable exit IEFU83
 - ▶ Assembles and link edits the exit IEFU83 into system accessible APF authorized loadlib
 - ▶ Uses IEFU83 to invoke the tracker subsystem that records the data set close type S event in ECSA
- Bases the data set close events on the creation of SMF records 14, 15, and 64
 - ▶ SMF recording for these three records are enabled in SYS1.PARMLIB.

SMF setup overview.

Tivoli Workload Scheduler for z/OS uses the SMF exit IEFU83 to track data set close events in the z/OS system. Tivoli Workload Scheduler for z/OS provides this exit as a member of the sample library SEQQSAMP. You must use SYS1.PARMLIB to implement and enable the IEFU83 exit. You must also assemble and link-edit the IEFU83 exit into a system-accessible APF authorized load library. IEFU83 intercepts SMF data set close records. When IEFU83 encounters an SMF record 14, 15, or 64, the IEFU83 exit invokes the tracker subsystem event-generation module EQQSSCMH passing the SMF record information to the module. EQQSSCMH then records the data set close event as a type S event in ECSA.

These data set close events are based on the creation of SMF records 14, 15, and 64. You must enable SMF recording for these three records in SYS1.PARMLIB.

The SMF setup is very similar for all of the Tivoli Workload Scheduler for z/OS SMF exit routines. All three of the SMF exit routines are discussed to some extent during this module.

SMF data set close activity records

- Use SMF type 14 records for non-VSAM data sets opened for INPUT or RDRBACK processing
 - ▶ Use Record 14 for data set triggering when the SRREAD parameter of the IEFU83 macro in the IEFU83 exit routine is set to YES
 - ▶ SRREAD=NO is the default
- Use SMF type 15 records for non-VSAM data sets opened for output
 - ▶ Use Record 15 for data set triggering when SRREAD=YES or SRREAD=NO. SRREAD has no affect for SMF record type15
- Use SMF type 64 records for VSAM data sets
 - ▶ Use Record 64 only for data set triggering with VSAM data sets

SMF data set close activity records.

Non-VSAM data sets that are opened for INPUT or RDRBACK processing require SMF Type 14 records. Use Record 14 for data set triggering when the SRREAD parameter of the IEFU83 exit is set to YES. SRREAD=NO is the default.

Non-VSAM data sets that are opened for output require SMF Type 15 records. Use Record 15 for data set triggering with SRREAD=YES or SRREAD=NO. SRREAD has no affect on SMF record type 15.

VSAM data sets require Type 64 records. Use Record 64 only for data set triggering when a VSAM data set data component is closed.

Tivoli Workload Scheduler for z/OS requires all three SMF records be collected if you have installed the SMF IEFU83 exit with SRREAD=YES specified on the EQQEXIT macro in IEFU831. When you specify SRREAD=YES, you want a special resource availability event automatically generated whenever a data set is closed regardless of why it was opened. When SRREAD=YES the data set can be opened for read processing, output processing, or both read and output processing. In this case, the data set close will always produce a special resource availability event.

To avoid data set triggering for record 14 and thus improve performance, specify SRREAD=NO on the EQQEXIT macro. SRREAD=NO is the default setting in the IEFU831 sample SMF exit routine.

You do not have to implement all three record types for data set triggering. For instance, if your triggering is not based on VSAM files, you do not need to implement record type 64.

Activating SMF data set activity records

- Define the SMF data set triggering records you want activated in SYS1.PARMLIB(SMFPRMxx) using the **SYS(TYPE** keywords
 - ▶ Create a second SMFPRMxx member and test it using the SET SMF=xx system console command to avoid an IPL
 - ▶ Use the **SET SMF=xx** console command again to revert to your original SMFPRMxx member if the test has problems
- SMFPRMxx **SYS(TYPE** example
SYS(TYPE(6,14,15,18,26,30,60,62,64,90)

Activating SMF data set activity records.

Define the SMF data set triggering records you want activated in SYS1.PARMLIB member SMFPRMxx by creating a second SMFPRMxx member. Test it using the SET SMF=xx system console command to avoid an IPL. If the test has problems, you can use the SET SMF=xx console command again to revert to your original SMFPRMxx member.

The example on the slide shows that all three data set triggering records, 14, 15, and 64 have been selected for activation.

Assume that the current working SMFPRMxx member on your z/OS test system is SMFPRM00. You can make a copy of it and name the copy SMFPRM01. Add the 14, 15, and 64 records to the SYS(TYPE statement of SMFPRM01 and save it. Run the SET SMF=01 command from the z/OS system console to activate the SMF member SMFPRM01. Then, run your test. If the test fails, you can revert to the SMFPRM00 member by using the SET SMF=00 console command.

SMF exit routines

Exit Name	Sample Exit	Sample JCL/usermod	Event Types Processed
IEFACTRT	EQQACTRT	EQQSMF	3J, 3S
IEFUJI	EQQUJI1	EQQSMF	2
IEFU83	EQQU831	EQQSMF	4, 5, S

The sample EQQU831 exit routine source code must be assembled and link-edited into an APF authorized load library

The **IEFU83** executable load module must be accessible within either the LNKLST concatenation or LPA (fixed, pageable, or modified)

SMF exit routines.

Tivoli Workload Scheduler for z/OS uses several SMF exits to implement job tracking and data set triggering. However, the **IEFU83** exit for data set triggering is the exit you will learn about in this training module. The **IEFUJI** exit plays a minor role in data set triggering when you filter data set triggers within Tivoli Workload Scheduler for z/OS based on user ID. The IEFUJI exit is discussed later in this module. The other SMF exit, called IEFACTRT, is not part of the data set triggering.

The IEFU83 exit routine generates type S event records based on SMF records 14, 15, and 64. The type S events are data set close events also known as special resource availability events. If you have properly implemented all aspects of data set triggering, you see the type S event records in the event data set of your tracker started task. However, if you have only implemented the SMF portion of data set triggering, the tracker subsystem does not create type S event entries in ECSA. Therefore, the tracker started task cannot create type S event records in the tracker event data set. You must also implement the data set triggering criteria definitions using either the Tivoli Workload Scheduler for z/OS provided EQQLSENT macro or Tivoli Workload Scheduler for z/OS provided sample XML before the tracker started task can create type S event records in the event data set.

The Tivoli Workload Scheduler for z/OS sample library contains several members that support the implementation of the IEFU83 exit routine. The sample IEFU831 exit routine source code is located in member IEFU831 of the SEQQSAMP data set. If you use SMP/E to install and maintain the IEFU83 exit as a usermod, you can use the sample member EQQSMF to install the exit routine.

You must assemble and link-edit the sample EQQU831 exit routine source code into an APF authorized load library. The resulting IEFU83 load module must be accessible within either the LNKLST concatenation or LPA(fixed, pageable, or modified).

Activating the SMF exit routines

- Update the initialization member **SMFPRMxx** in SYS1.PARMLIB

Example:

```
SYS(NOTYPE(16:19,65:69),EXITS(IEFU83,IEFUJI,IEFACTRT)
SUBSYS(STC,EXITS(IEFU83,IEFUJI,IEFACTRT)
SUBSYS(JES2,EXITS(IEFU83,IEFUJI,IEFACTRT)
SUBSYS(OMVS,EXITS(IEFU83,IEFUJI ))
```

- Update **PROGxx** in SYS1.PARMLIB

Example:

```
EXIT ADD EXITNAME(SYS.IEFU83) MODNAME (IEFU83)
....
EXIT ADD EXITNAME(SYSSTC.IEFU83) MODNAME (IEFU83)
....
EXIT ADD EXITNAME(SYSJESn.IEFU83) MODNAME (IEFU83)
....
EXIT ADD EXITNAME(SYSOMVS.IEFU83) MODNAME (IEFU83)
```

Activating the SMF exit routines.

Active SMF exits are defined using the EXITS parameter of the SYS and SUBSYS keywords in the SMFPRMxx SYS1.PARMLIB member. The NOTYPE parameter in the SMFPRMxx example on this slide means activate all SMF records excluding the types explicitly listed.

Use the PROGxx parmlib member to specify installation exits and control their use. Using PROGxx, you can associate multiple exit routines with installation exits at IPL, or while the system is running. You can use PROGxx and SMFPRMxx to specify exits. Control the actual name of the exit routine load module by using the PROGxx SYS1.PARMLIB member and the EXIT ADD statement as shown on the slide. In the examples shown on the slide, the module name IEFU83 in RPOGxx matches the exit names in the SMFPRMxx member. However, the MODNAME parameter in PROGxx can specify a different module name for a given exit routine module. In the PROGxx example on the slide only the IEFU83 entries are shown. However, in a real production environment, there would be entries for all of the SMF exit routines you intend to activate.

You might find it useful during installation to write the code for two SMFPRMxx members, one with the exits active and the other with the exits inactive. You can then use the SET SMF=xx z/OS console command to switch your current SMF parameters to the new member. Using the SET SMF=xx command, you avoid the need to re-IPL. You can use the SET SMF=xx console command to restore the original member if you encounter a problem. You can also use the SET PROG=XX MVS console command to dynamically specify the PROGxx member.

This slide shows all three SMF exit routines being activated in the SMFPRMxx member. The IEFU83 exit routine is critical for data set triggering.

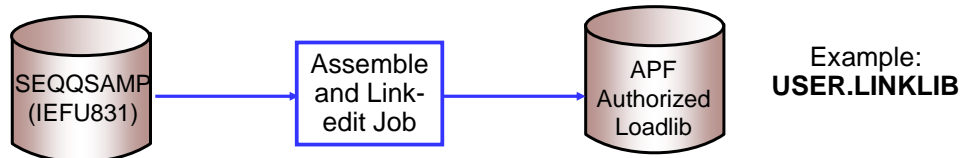
Exits for SUBSYS STC are required by Tivoli Workload Scheduler for z/OS.

JESn in the second example on the slide is either JES2 or JES3. This parameter does not refer to JES itself, but to batch jobs handled by JES.

If you are using FTP, you must add the SUBSYS(OMVS,EXITS(IEFUJI,IEFU83)) statement to the SMFPRMxx SYS1.PARMLIB member. Also, include an entry for the OMVS subsystem for the PROGxx member. Make sure that you replace MODNAME with the module name that was used when the exits routines were link-edited.

Assemble and link-edit SMF exit IEFU83

- Assemble and link-edit IEFU83 into an APF-authorized load library
 - Set **SRREAD=YES** first if you are using SMF record 14 for data set triggering events
 - Use **EQQSMF** for an SMP/E installation
- Create an assemble and link-edit job if you are not using SMP/E
 - For data set triggering, include **IEFU83**
 - SEQQSAMP(EQQU831)** is the source code input to the assemble and link-edit the job



Include the APF authorized load library in either the LNKLST concatenation or in LPA

Examples for **PROGxx**:

APF ADD DSNAME(USER.LINKLIB)	VOLUME(*MCAT*)
LNKLST ADD NAME(LNKLST00) DSN(USER.LINKLIB)	VOLUME(*MCAT*)

Assemble and link-edit SMF exit IEFU83.

Assemble and link-edit the SMF exit routine source code IEFU831 into an APF-authorized load library. Set SRREAD=YES first if you use SMF record 14 for data set triggering events. Use EQQSMF for an SMP/E installation or create an assemble and link-edit job if not using SMP/E. For data set triggering, include IEFU83 as an SMF exit. SEQQSAMP(EQQU831) is the source code input to the assemble and link-edit the job.

You can use two different methods to assemble and link-edit the IEFU83 exit routine into an APF-authorized load library. The first method is to use the EQQSMF sample member provided by Tivoli Workload Scheduler for z/OS and do an SMP/E installation of the exit. The SMP/E apply processing assembles and links the exit routine into a valid load library. You can then APF authorize the load library and include it in LPA or the LNKLST concatenation.

The second method is to create your own assemble and link-edit job. Use this method if you are not using SMP/E to install the IEFU83 exit. Use the SEQQSAMP member IEFU831 as the source code input to the job. Link-edit the IEFU83 exit into an APF-authorized load library that is in the LNKLST concatenation or in LPA. At the bottom of this slide is an example of an exit that has been assembled and link-edited into the USER.LINKLIB executable load library. The example also shows how you can use the PROGxx member APF ADD and LNKLST ADD statements to make the USER.LINKLIB data set APF-authorized and included in the LNKLST.

Remember to change the SRREAD=NO parameter in the SEQQSAMP member IEFU831 before you run the job if you create data set triggering event records using SMF record 14. The statement you want to change is about midway in the source code and defaults to EQQEXIT EXIT=IEFU83,SRREAD=NO.

After you link the Tivoli Workload Scheduler for z/OS exit routines into the target load library, IPL the system unless you intend to use the MVS console set command to activate your SMFPRMxx and PROGxx members.

Verifying SMF setup for exit routines (1 of 3)

- Verify the exit routine load library you are using
 - ▶ First in the LNKLST concatenation or LPA search order
 - ▶ Ahead of any older libraries that might be installed and that contain an older exit routine with the same name
- IPL the z/OS system if you have not done so already
 - ▶ The IPL ensures that the IEFU83 exit routine is available in LNKLST or LPA
- Start the Tivoli Workload Scheduler for z/OS Tracker started the task subsystem
- Job-tracking events are recording in the tracker event data set with the DD name of **EQQEVDS**
- To verify both SMF and JES exit routine implementation, read this Technote:
 - ▶ CHECKING LEVEL OF OPC Tivoli Workload Schedulerz JOBTRACKING EXITS LOADED INTO MEMORY Technote Reference #: 1141092
 - ▶ URL: <http://www-1.ibm.com/support/docview.wss?uid=swg21141092>

Verifying SMF setup for exit routines (1 of 3).

Verify that the exit routine load library you are using is first in the LNKLST concatenation or LPA search order. Make sure the library is ahead of any older libraries that might be installed and that contain an older exit routine with the same name. The system uses the first module it finds by that name.

IPL the z/OS system if you have not done so already. This action ensures that the IEFU83 exit routine is available in LNKLST or LPA. You can also use the MVS console commands SET SMF and SET PROG to avoid an IPL. Ensure that the correct IEFU83 exit routine is available. Start the Tivoli Workload Scheduler for z/OS tracker started task. The tracker started task should now be recording SMF exit driven job-tracking events as they occur. The tracker started task records the events in the event data set EQQEVDS.

The tracker subsystem is invoked by the IEFU83 exit routine when SMF records 14, 15, or 64 are encountered by the exit. The tracker subsystem records type S event entries in ECSA. The tracker started task eventually records them into the event dataset EQQEVDS as well. You can browse the EQQEVDS data set to see the type S event records. Remember that you must have data set close activity on the system for the events to occur. Also, you must define the data set triggering criteria in Tivoli Workload Scheduler for z/OS before the tracker started task can write the records into the event data set. This selection criteria is the names of the data sets that you want to have the tracker subsystem select when creating type S event records. The data set tracking criteria is defined using either the Tivoli Workload Scheduler for z/OS provided EQQLSENT macro or Tivoli Workload Scheduler for z/OS provided sample XML.

After implementing all aspects of data set triggering, you will see type S event records in the tracker started task event data set. Execute technote number 1141092 before you test the data set triggering function. The technote is available at the URL that is listed on the slide.

Verifying SMF setup for exit routines (2 of 3)

- If you do not have data set triggering fully implemented, you can still check EQQEVDs for the other SMF exit event records related to the routine
 - Event **type 2**: A job start event that indicates that SMF exit routine IEFUJI is correctly installed and functioning
 - Event **type 3J**: A job end event that indicates that the SMF exit routine IEFACTRT is correctly installed and functioning
- Example EQQEVDs entries of IEFUJI and IEFACTRT event records

```

...A1 -..LISTUCATJOB08026.....I.....>
...A2 -..LISTUCATJOB08026.....I.....>
...A3J -..LISTUCATJOB08026.....n.....>
  
```

The letter **A** in front of the event type indicates a JES2 system
 The letter **B** would indicate a JES3 system

Verifying SMF setup for exit routines (2 of 3).

If you do not have data set triggering fully implemented, you can still check EQQEVDs for the other SMF exit routine related event records. Event type 2 is a job-start event that indicates that the SMF exit routine IEFUJI is correctly installed and functioning. Event type 3J is a job-end event that indicates that the SMF exit routine IEFACTRT is correctly installed and functioning.

The slide shows a screen capture example of a segment in the EQQEVDs data set of a tracker started task. The example shows two SMF exit routine related event records that were generated when the job named LISTUCAT ran and completed on the system where the tracker is running.

The letter **A** in front of the event type indicates a JES2 system. The letter **B** would indicate a JES3 system.

Verifying SMF Setup for Exit Routines (3 of 3)

- If you have data set triggering fully implemented, you see type **S** event records in the EQQEVDs data set of the tracker
 - You provided the tracker with the names of the data sets you want to track
 - You set up SMF correctly
- Example EQQEVDs record entries for Data Set Triggering event type **S**

```

.A2 ..... LISTUCATJOB08033 ..... eYj ..... eYy ..... eYj .....
.A3J ..... LISTUCATJOB08033 ..... eY0 ..... eYy ..... eYj ..... eY0 .....
.A3P.0-... LISTUCATJOB08033 ..... eY2 ..... eYy ..... eYj ..... eY1H...
.NFSE  .2 ..... u&-... i1.SYY) ..... ePú ..... ePúJAYHILL JA
.NNSE  .2 ..... u&-... 1.EXSA02JAYHILL.AJCL
.NNSE  .2 ..... u&-... 1 ..... R ..... JAYH
.NFSE  .0 ..... u&-... i1.SYY) ..... e@i ..... e@iJAYHILL JA
.NNSE  .0 ..... u&-... 1.EXSA02JAYHILL.AJCL
.NNSE  .0 ..... u&-... 1 ..... R ..... JAYH
.NFSE  .0 ..... u&-... i1.SYY) ..... e£) ..... e£)JAYHILL JA
.NNSE  .0 ..... u&-... 1.EXSA02JAYHILL.AJCL
.NNSE  .0 ..... u&-... 1 ..... R ..... JAYH

```

Type S records have an **SY** in columns 21-23

Verifying SMF setup for exit routines (3 of 3).

If you have data set triggering fully implemented, you see type S events in the EQQEVDs data set of the tracker. You provided the tracker with the names of the data sets you want to track, and you set up SMF correctly.

The example screen capture on this slide shows the EQQEVDs event record entries for several data set triggering events that have occurred for the data set called JAYHILL.AJCL. The following steps generated these records.

1. SMF records 14 and 15 were activated in SYS1.PARMLIB member SMFPRMxx.
2. The Tivoli Workload Scheduler for z/OS supplied exit routine IEFU83 was implemented using SYS1.PARMLIB members SMFPRMxx and PROGxx.
3. The Tivoli Workload Scheduler for z/OS supplied exit routine IEFU83 was assembled and link-edited into an APF authorized load library.
4. The IEFU83 module was placed in LPA using SYS1.PARMLIB member name IEAFIXxx.
5. The system was IPLed.
6. The data set name JAYHILL.AJCL was provided to the tracker using the EQQLSENT macro. This step is covered in a separate training module.
7. The data set JAYHILL.AJCL was opened and closed.

Technote 1141092 example (1 of 2)

Run the following two MVS console commands in sequence from SDSF

1. D PROG,EXIT,ALL to get the formal name of the IEFU83 exit routine

```
SYS.IEFUTL      E  SYS.IEFU84      E  SYS.IEFU83      E
SYS.IEFUJI      E  SYS.IEFACTRT   E  IRREX01          E
```

2. D PROG,EXIT,EN=SYS.IEFU83,DIAG to get the entry point address of IEFU83 using the formal name

```
CSV464I 17.14.23 PROG,EXIT DISPLAY 096
EXIT SYS.IEFU83
MODULE      STATE EPADDR      LOADPT      LENGTH      JOBNAME
IEFU83      A      850B9268    00000000   00000000   *
```

Run the two commands that are on the next slide in sequence
Remember the EPADDR (entry point address)

Technote 1141092 example (1 of 2).

Technote 1141092 provides a detailed explanation of how to verify that the correct Tivoli Workload Scheduler for z/OS exits are properly implemented on your system.

The current slide and the next slide provide a summary of how to verify the implementation of the IEFU83 exit routine. The two slides follow the command sequence described in the technote.

Run the D PROG,EXIT,ALL MVS console command from SDSF to get the formal name of the IEFU83 exit. You can see in the example screen capture on the slide that the formal name is SYS.IEFU83.

Use the formal name to run the D PROG,EXIT,EN=SYS.IEFU83 MVS console command from SDSF to get the entry point address of the IEFU83 exit routine. You can see from the example screen capture that the entry point address is 850B9268. Copy or write down this address for future use. In the next slide you see how to use the address to browse the actual memory location of the IEFU83 exit routine module.

Technote 1141092 example (slide 2 of 2)

3. TSO ISRDDN to start the ISRDDN utility from TSO/ISPF

```

Current Data Set Allocations                               Row 1 of 38
Command ==> BROWSE 850B9268.                               Scroll ==> PAGE
Volume  Disposition Act DDname  Data Set Name  Actions: B E V M F C I Q
G1802C  SHR,KEEP    >  _  ISPALIB      ISP.SISPALIB

```

4. BROWSE 850B9268. Run the BROWSE subcommand of the ISRDDN utility to display the memory content at the IEFU83 exit routine entry point. Use a **period** after the memory address.

```

BROWSE  STORAGE Start:850B9268                               850B9268.
Command ==> _____ Scroll ==> PAGE
***** Top of Data *****
+0 (050B9268)  47F0F01C C9C5C6E4 F8F34040 F0F561F2 * 00.IEFU83 05/2 *
+10 (050B9278) F061F0F9 40404040 40404040 90ECD00C * 0/09 00}. *
+20 (050B9288) 05C04100 01380700 47F0C00E 00000000 * .{ ... 00{... *

```

Technote 1141092 example (2 of 2).

Run the TSO command, TSO ISRDDN, from the TSO/ISPF command line. The ISRDDN utility screen is displayed. On the command line of that screen, enter the BROWSE subcommand of the ISRDDN utility. Specify the entry point address of the IEFU83 exit routine that you had previously obtained and noted. Place a period after the entry point address when invoking the BROWSE command.

The memory location of the IEFU83 exit routine is displayed. Check the date that starts on the first line at that location. Ensure that the date matches the date when you assembled and link-edited the Tivoli Workload Scheduler for z/OS 8.5 IEFU83 module. If the dates do not match, a different version of the module is being used. Investigate your SYS1.PARMLIB setup.

Additional useful MVS console commands

- D SMF,O displays the current SMF options
- D PROG,LPA,MODNAME=IEFU83 display the memory locations of a program module in LPA
- D PROG,LNKLST displays the current LNKLST concatenation
- D PROG,APF displays the APF authorized libraries

Additional useful MVS console commands.

This slide shows four additional MVS console commands that you can run from SDSF. Use these commands to determine whether you have properly set up SYS1.PARMLIB for the IEFU83 exit routine. You can also use these commands for the other Tivoli Workload Scheduler for z/OS SMF exit routines.

The first command D SMF,O displays the current SMF options. You can see the SMF records and exits that you have activated.

If you implemented your SMF exit routines using LPA, the second command D PROG,LPA,MODNAME=IEFU83 shows the entry point address of the IEFU83 exit routine. You see the address if you used LPA to implement the exit routine.

The third command D PROG,LNKLST shows the current order of the LNKLST concatenation. Seeing the current order is useful if you are using LNKLST to implement your exit routine library.

The fourth MVS console command D PROG,APF displays the APF authorized load libraries on your system and the DASD volume that they are located on.

Filtering events based on user ID

- You can filter data set triggering events within Tivoli Workload Scheduler for z/OS based on the user ID of the user that creates or modifies the data set
- Set the SETUID=parameter of the EQQEXIT macro to YES
The parameter is in the Tivoli Workload Scheduler for z/OS supplied IEFUJI exit routine sample
 - ▶ Located in SEQQPARM(IEFUJI1)
 - ▶ SETUID=YES is the default

```
EQQEXIT EXIT=IEFUJI,SETUID=YES
```
- Assemble and link-edit the routine

Filtering events based on user ID.

If you have a requirement to filter data set triggering events based on the user ID of the user that creates or modifies the dataset, set the SETUID parameter of the IEFUJI exit macro to YES before you assemble and link-edit the IEFUJI exit routine. The default is YES in the sample. Therefore, do not change anything if you use the sample in SEQQSAMP member IEFUJI1.

Summary

You should now be able to:

- Implement the SMF records and exit routines required for Tivoli Workload Scheduler for z/OS data set triggering.
- Verify that your implementation of SMF for data set triggering is correct.

Summary.

In summary you should now be able to implement the SMF records and exit routines required for Tivoli Workload Scheduler for z/OS data set triggering. You should also be able to verify that your implementation of SMF for data set triggering is correct.

Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_smf-setup.ppt

This module is also available in PDF format at: [../smf-setup.pdf](..../smf-setup.pdf)

You can help improve the quality of IBM Education Assistant content by providing feedback.

Trademarks, copyrights, and disclaimers

IBM, the IBM logo, ibm.com, and the following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

Tivoli

z/OS

If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of other IBM trademarks is available on the Web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>

Other company, product, or service names may be trademarks or service marks of others.

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.

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 2009. 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.