



IBM Software Group

z/OS® V1R9 Communications Server

CICS® sockets enhancements



@business on demand.

© 2008 IBM Corporation
Updated January 11, 2008


This presentation discusses the enhancements to IP CICS Sockets for the z/OS V1R9 Communications Server.

Agenda


- Listener will reconnect to its TCP/IP stack
- Listener tolerates a full socket descriptor table
- IP CICS Socket program load table (PLT) program supports immediate shut down method
- Identify the task's TCP/IP stack job name in the CEMT INQUIRE TASK output
- Automatic IP CICS Socket Support



Five CICS enhancements are addressed by this presentation.

IBM Software Group 

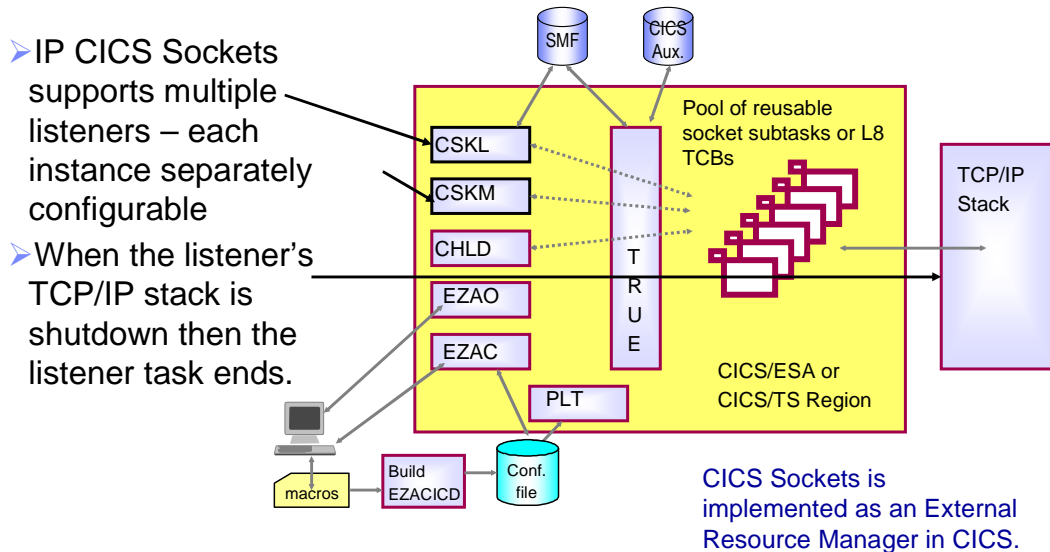
Listener reconnects to its TCP/IP stack



CICS sockets enhancements © 2008 IBM Corporation 3

This section describes the enhancement that allows the listener to reconnect to its TCP/IP stack.

Background information - IP CICS sockets structural overview



The IP CICS Socket interface consists of the a CICS task related user exit (TRUE), an application programming interface (API), a listener program (EZACIC02) and CICS transactions for configuring (EZAC) and managing (EZAO) the interface and listeners. The socket interface consists of a set of calls that your CICS application programs can use to set up connections, send and receive data, and perform general communications control functions. The programs can be written in COBOL, PL/I, assembler language, or the C language.

When the stack is shutdown and restarted then the Listener will be affected if it has affinity with that TCP/IP stack.

Problem - Listener task does not persist

- The IP CICS Sockets Listener task ends when the TCP/IP stack with which it has affinity is shutdown.
 - ▶ Requires a subsequent operation action to restart the listener task when the stack is reestablished.



The IP CICS Sockets listener task ends when its TCP/IP stack with which it has an affinity is shutdown. A subsequent operation is required to restart the listener task.

Solution - Automatically reconnect to the TCP/IP stack

- Change the listener to enable it to remain active when the TCP/IP stack with which it has affinity is recycled.
 - ▶ Enable the listener with the ability to reconnect to the stack.
 - ▶ RTYTIME – Retry time
 - ✓ The time in seconds to determine how long the Listener will wait (CICS/TS task delay) to reconnect to its TCP/IP stack after an outage
 - ✓ Will attempt to re-connect after delay expires
 - ✓ If the re-connection succeeds then the listener will continue to re-establish itself as a listener
 - ✓ Listener can be shutdown while delayed
 - ▶ The Listener's action is based on the combination of RTYTIME and the availability of the TCP/IP stack

Listener	RTYTIME	TCP/IP down	TCP/IP up
Initially started	0	Listener ends	Listener initializes or remains active
	>0	Listener waits	
Previously active	0	Listener ends	
	>0	Listener waits	

6

Change the listener to automatically re-establish itself (based on configuration) when its stack is restarted.

The retry time configuration option is used to indicate that you want the listener to re-connect to its TCP/IP stack. The value specified by RTYTIME affects the initial connection or re-connection of the listener to its TCP/IP stack. The range for RTYTIME is either 0 or 15-999 seconds. The value of 0 indicates that the listener will not be delayed but will retry to connect to its TCP/IP stack once and will end if that attempt fails. If RTYTIME is configured with the values 1 through 14 then the value of 15 seconds will be used instead to prevent excessive connection attempts. If RTYTIME is not specified then the default value of 15 seconds is used.

The chart shows how the listener will react based on how the RTYTIME configuration option is specified and the availability of the TCP/IP stack.

The RTYTIME configuration option can be specified on the listener definition, EZACICD TYPE=LISTENER. The IP CICS Sockets configuration macro, EZACICD, is designed to support the configuration of the IP CICS Socket interface and listeners exploiting that interface. Regenerate the IP CICS Socket configuration dataset, EZACONFG, with the new RTYTIME configuration option.

Configuration - Standard listener

- Standard listener definition: EZAC transaction

```
EZAC,DEFine,LISTENER (standard listener. screen 1 of 2)      APPLID = CICS1A
```

Overtyp e to Enter

APPLID	====> CICS1A	APPLID of CICS System
TRANID	====> CSKL	Transaction Name of Listener
PORT	====> 03010	Port Number of Listener
AF	====> INET	Listener Address Family
IMMEDIATE	====> YES	Immediate Startup Yes No
BACKLOG	====> 020	Backlog Value for Listener
NUMSOCK	====> 050	Number of Sockets in Listener
ACCTIME	====> 060	Timeout Value for ACCEPT
GIVTIME	====> 000	Timeout Value for GIVESOCKET
REATIME	====> 000	Timeout Value for READ
RTYTIME	====> 015	Stack Connection Retry Time
LAPPLD	====> INHERIT	Register Application Data

Verify parameters, press PF8 to go to screen 2

PF 3 END

8 NEXT

12 CNCL

7

CICS sockets enhancements

© 2008 IBM Corporation

The RTYTIME configuration option can also be specified using the online configuration transaction, EZAC. The EZAC transaction will update the configuration contained in the EZACONFG VSAM dataset but not the original EZACICD macro. Ensure you mirror any updates against your sites EZACICD configuration macros.

Configuration - Enhanced listener

- Enhanced listener definition: EZAC transaction

```

EZAC,DEFine,LISTENER (enhanced listener.  screen 1 of 2)      APPLID = CICS1A

Overtyp e to Enter

APPLID      ==> CICS1A          APPLID of CICS System
TRANID      ==> CSKL           Transaction Name of Listener
PORT        ==> 03010          Port Number of Listener
AF          ==> INET           Listener Address Family
IMMEDIATE   ==> YES            Immediate Startup  Yes|No
BACKLOG     ==> 020            Backlog Value for Listener
NUMSOCK     ==> 050            Number of Sockets in Listener
ACCTIME     ==> 060            Timeout Value for ACCEPT
GIVTIME     ==> 000            Timeout Value for GIVESOCKET
REATIME     ==> 000            Timeout Value for READ
RTYTIME     ==> 015            Stack Connection Retry Time
LAPPLD      ==> INHERIT        Register Application Data

Verify parameters, press PF8 to go to screen 2

PF 3 END          8 NEXT          12 CNCL

```

The RTYTIME configuration option can also be specified, for the enhanced listener, using the online configuration transaction, EZAC. The EZAC transaction will update the configuration contained in the EZACONFG VSAM dataset but not the original EZACICD macro. Ensure you mirror any updates against your sites EZACICD configuration macros.

The EZAC transaction will issue particular messages when validating the RTYTIME configuration option.

A non-numeric value is considered incorrect. A value other than 0 or 15-999 is considered an incorrect value. A value of 1-14 is considered to be below the minimum non-zero value of 15. Change any incorrect value and press the enter key.

RTYTIME > 0 - Messages

```

EZY1369E 07/24/06 14:22:19 LISTENER CSKL 0000021L IS DELAYED,
STACK TCPCS      IS UNAVAILABLE
EZY1363I 07/24/06 14:22:32 LISTENER CSKL 0004039L HAD 3 THREADS
ACTIVE WHEN STACK TCPCS      ENDED
EZY1367I 07/24/06 14:22:40 SOCK#      IP ADDRESS  PORT  CHILD
EZY1368I 07/24/06 14:22:46           1 9.42.104.42 03010 PAYR
EZY1368I 07/24/06 14:22:46           3 9.42.104.42 03010 ORDR
EZY1368I 07/24/06 14:22:46           9 9.42.104.42 03010 ATML
EZY1293E 07/24/06 14:23:01 INITAPI CALL FAILURE TRANSACTION =
CSKL TASKID= 0004 ERRNO= 1036
EZY1291I 07/24/06 14:30:29 LISTENER TRANSACTION= CSKL TASKID=
0000021L ACCEPTING REQUESTS VIA PORT 3010

```

9

CICS sockets enhancements

© 2008 IBM Corporation

Here is an example of the messages you may see when the listeners stack has been shutdown while it was processing connections and RTYTIME value is greater than zero.

Message EZY1369E will be issued by the listener when it is active and its TCP/IP stack has been shutdown. The message contains the listener's transaction ID, the listener's CICS task number, and the listener's TCP/IP stack job name. In this example the listener's transaction ID is **CSKL**, the listener's CICS task number is **0000021L** and the listener's TCP/IP job name is **TCPCS**.

Message EZY1363I is issued by the listener to provide the number of in-flight threads. An in-flight thread is that which has been established by the listener as being connected but not yet taken by a child server process. This message contains the listener's transaction ID, the listener's CICS task ID, the number of active threads, and the listener's TCP/IP job name. In this example there were 3 threads active when the TCP/IP stack was shutdown.

Message EZY1367I is a header message for EZY1368I.

Message EZY1368I provides details about in-flight threads. In this example the message is issued three times because there were 3 in-flight threads.

Message EZY1293E is an existing message which indicates that the listener experience a failure on an INITAPI call.

Message EZY1291I is an existing message. It indicates that the Listener transaction is now available to receive connection requests on the specified port. In this example it was issued after the listener successfully reconnected to its TCP/IP stack after the TCP/IP stack had been restarted.

RTYTIME = 0 - Messages

```

EZY1369E 07/24/06 14:22:19 LISTENER CSKL 0004039L IS DELAYED,
      STACK TCPCS      IS UNAVAILABLE
EZY1363I 07/24/06 14:22:32 LISTENER CSKL 0004039L HAD 3 THREADS
      ACTIVE WHEN STACK TCPCS      ENDED
EZY1367I 07/24/06 14:22:40 SOCK#      IP ADDRESS  PORT  CHILD
EZY1368I 07/24/06 14:22:46      1 9.42.104.42 03010 PAYR
EZY1368I 07/24/06 14:22:46      3 9.42.104.42 03010 ORDR
EZY1368I 07/24/06 14:22:46      9 9.42.104.42 03010 ATML
EZY1293E 07/24/06 14:23:01 INITAPI CALL FAILURE TRANSACTION = CSKL
      TASKID= 0004 ERRNO= 1036
EZY1289E 07/24/06 14:23:11 CICS/SOCKETS LISTENER TRANSACTION CSKL
      TERMINATING
EZY1278I 07/24/06 14:23:20 EZACIC03 DETACH SUCCESSFUL TCB
      ADDRESS=0079BBF8 TRAN=CSKL TASK=0004039L

```

10

CICS sockets enhancements

© 2008 IBM Corporation

Here is an example of the messages you may see when the listeners stack has been shutdown while it was processing connections and RTYTIME was configured with a value of 0.

Message EZY1369E will be issued by the listener when it is active and its TCP/IP stack has been shutdown. The message contains the listener's transaction ID, the listener's CICS task number, and the listener's TCP/IP stack job name. In this example the listener's transaction ID is **CSKL**, the listener's CICS task number is **0004039L** and the listener's TCP/IP job name is **TCPCS**.

Message EZY1363I is issued by the listener to provide the number of in-flight threads. An in-flight thread is that which has been established by the listener as being connected but not yet taken by a child server process. This message contains the listener's transaction ID, the listener's CICS task ID, the number of active threads, and the listener's TCP/IP job name. In this example there were 3 threads active when the TCP/IP stack was shutdown.

Message EZY1367I is a header message for EZY1368I.

Message EZY1368I provides details about the in-flight threads. In this example the message is issued three times because there were 3 in-flight threads.

Message EZY1293E is an existing message which indicates that the listener experience a failure on an INITAPI call.

Message EZY1289E is an existing message that indicates that the listener is terminating.

Message EZY1278I is an existing message that indicates that an attached subtask is terminating.

Things to think about

- Current IP CICS Socket configurations will inherit the default of RTYTIME=15 seconds.
 - ▶ Either use the EZAC transaction to change it or
 - ▶ Add RTYTIME to the configuration macros and re-generate the configuration.
- If you have written a program which makes use of any of the external IP CICS Socket control blocks then you must reassemble that program to pick up the latest changes.

For listener configurations established before V1R9 the retry time of 15 seconds will be used. Tailor this value to meet your downtime needs.

Recompile any user written socket programs or listeners which use the IP CICS Socket macros. Users who have written their own specialized listeners will be most affected.

Listener tolerates a full socket descriptor table

This section describes the enhancement to allow the listener to tolerate a full socket descriptor table condition.

Background information

- z/OS IP CICS Sockets
 - ▶ The IP CICS Socket Listener is a concurrent listener executing as a CICS/TS task which creates socket descriptors
 - ▶ z/OS UNIX system services support a configurable number of socket descriptors per process.
- MAXFILEPROC parameter
 - ▶ Specifies the maximum number of descriptors that a process can have open concurrently
 - ▶ This is a system wide limit
 - ▶ Can be overridden for an individual process by specifying the RACF® ADDUSER or ALTUSER for FILEPROC MAX

The listener is designed to function as a concurrent listener. It creates and manages its sockets giving them to child server tasks. It creates a listening socket and accepts sockets from client connections.

CICS/TS 2.2 is dubbed to OMVS as one process per address space. CICS/TS 2.3 is dubbed to OMVS as one process per CICS/TS task.

MAXFILEPROC is a z/OS UNIX System Services parameter that specifies the maximum number of descriptors for files, sockets, directories, and any other file system objects that a single process can have concurrently active or allocated. It is defined in SYS1.PARMLIB member, BPXPRMxx. It can be changed using the SETOMVS console command. Note that MAXFILEPROC is the same as the OPEN_MAX variable in the POSIX standard.

FILEPROC MAX specifies the maximum number of files the user is allowed to have concurrently active or open. The files-per-process you define to RACF is a numeric value from 3 and 542287. The value specified for FILEPROC MAX overrides any value provided by the MAXFILEPROC parameter of BPXPRMxx. The RACF ALTUSER command is used to change the information in a user's profile. The RACF ADDUSER command is used to define a new user to RACF and establish the user's relationship to an existing RACF-defined group.

Problem statement - Listener ends

- When the socket table becomes full the listener ends.
 - ▶ MAXFILEPROC is incorrectly configured or is configured lower than the listener's NUMSOCK configuration
 - ▶ Requires the operator to restart the listener transaction once the socket table full condition is resolved.

When the listener reaches its maximum number of allowed descriptors then it will end. The MAXFILEPROC default is 64000 (The POSIX standard is 16) and the IP CICS Socket NUMSOCK configuration option default is 50.

Solution - Listener remains active

- New message EZY1370I indicates a configuration problem
 - ▶ Issued at listener startup
- The IP CICS Sockets Listener will not end when the EMFILE (24) errno condition is raised when accepting client connections.
 - ▶ ACCEPT processing will be delayed
 - ▶ Message EZY1365E is issued
- Adjust the configuration
 - ▶ MAXFILEPROC
 - ▶ FILEPROCMAX
 - ▶ Listener's NUMSOCK

15

CICS sockets enhancements

© 2008 IBM Corporation

The solution to the problem is to configure the environment by setting the listener's NUMSOCK value to be less than or equal to the FILEPROCMAX value for the listener's user ID or the value specified for MAXFILEPROC if FILEPROCMAX is not being used.

A run-time check is done when the listener starts to determine if the z/OS UNIX System Services MAXFILEPROC value is less than or equal to the listener's NUMSOCK value. If so this message is issued:
EZY1370I mm/dd/yy hh:mm:ss LISTENER transactionid NUMSOCK numsock IS EQUAL TO OR GREATER THAN MAXFILEPROC maxfileproc

The listener's accept processing will pause when the number of sockets being handled exceeds the MAXFILEPROC value. No new connections will be accepted until the number of sockets falls below the MAXFILEPROC value.

Also, the value specified for the user ID's FILEPROCMAX should be configured appropriately. If the number of sockets the listener creates exceeds the listener's user ID's FILEPROCMAX value or the MAXFILEPROC value then an EMFILE error condition occurs and the listener will cease accepting new sockets until its active sockets are at the FILEPROCMAX value or less. If FILEPROCMAX is not being used then the number of active sockets will have to be equal to or less than the MAXFILEPROC value.

The listener has been changed to remain active when it reaches its maximum number of descriptors. The EMFILE errno is used to indicate that the process table is full. The following message will be issued when the listener has received the EMFILE error condition from ACCEPT processing.

EZY1365E mm/dd/yy hh:mm:ss LISTENER transactionid taskno IS NOT ACCEPTING REQUESTS ON PORT port

This message indicates that the listener identified by the specified transaction ID and task number is not able to process inbound connections because the listener's socket descriptor table is full. Once a successful ACCEPT is processed then this condition will be relieved.

**IP CICS Socket program load table (PLT) program
supports immediate shutdown method**

When shutting down the IP CICS Socket interface and listener using the EZACIC20 PLT program then it will shutdown only when all socket work completes.

Background information

- The IP CICS Socket interface supports two shutdown methods:
 - ▶ Deferred
 - ▶ Immediate
- Shutdown by these processes
 - ▶ Operator transaction, EZAO
 - ▶ Program link – EZACIC20
 - ▶ CICS/TS program load table (PLT) program – EZACIC20

Historically, the IP CICS Socket interface can be shutdown using many different methods. A deferred shutdown enables all IP CICS sockets tasks to end gracefully. An immediate shutdown directs all IP CICS sockets tasks to be immediately terminated.

Shutdown can be either deferred or immediate using the EZAO operator transaction.

Shutdown can also be deferred or immediate using the program link EZACIC20. The P20TYPE field in the COMMAREA provided in the EXEC CICS LINK to program EZACIC20 specifies whether immediate or a deferred termination is requested.

You can allow automatic shutdown of the CICS socket interface through updates to the program load table (PLT). This is achieved through placing the EZACIC20 module in the appropriate PLT. Ideally, the CICS system programmer should add the IP CICS Socket shutdown program, EZACIC20, to their program load table (PLT) to facilitate in shutting down the IP CICS Socket interface and listener. Only the deferred shutdown method is supported when using the PLT.

Problem - Only supports deferred shutdown

- The IP CICS Socket interface CICS/TS program load table (PLT) program, EZACIC20, only supports a deferred shutdown method.
 - ▶ Forces CICS/TS shutdown to wait for all in-flight socket programs to end before the interface will shutdown.
 - ✓ Affected by blocking socket calls

When the IP CICS Socket PLT program is used any transactions being blocked by blocking sockets command will wait for them to return. This may require the CICS system programmer to use CICS shutdown assist or manually terminate those blocking tasks. When recycling CICS/TS this manual action will elongate the user's down times due to having to wait for CICS/TS to shutdown before being restarted.

Solution - Support immediate shutdown method

- Allow user to specify how the interface will shutdown when the IP CICS socket PLT is used
- A new IP CICS sockets program load table (PLT) configuration option
 - ▶ PLTSDI
 - ✓ Can have a values of **NO** or **YES**
 - ✓ Can be specified on the interface definition, EZACICD TYPE=CICS
 - ✓ Can be specified on the EZAC transaction, EZAC,DEFine,CICS
- Stopping CICS TCP/IP with program link
 - ▶ EZACIC20 can query the PLTSDI configuration value

```

*
*          STORAGE DEFINITION FOR EZACIC20 PARAMETER LIST
*
P20PARMS DS      0D
P20TYPE  DS      CL1      Initialization Type
P20TYPEI EQU     C'I'      Initialization
P20TYPEP EQU     C'T'      Immediate Termination
P20TYPEQ EQU     C'D'      Deferred Termination (CICS Only)
P20TYPEQ EQU     C'Q'      Query PLTSDI

```

The PLT program, EZACIC20, now has the ability to shutdown using the immediate method whereby any transaction being blocked by a blocking socket call is immediately unblocked. The new PLT shutdown immediate configuration option, PLTSDI, is used to tell the PLT program how to shutdown the interface and listeners when CICS/TS is being shutdown. This option can have a value of NO or YES. The value of NO is used to signify that a deferred shutdown is required. This is the default. The value of YES is used to signify that an immediate shutdown is desired.

The configuration macro, EZACICD, is used to build the configuration data set. TYPE=CICS identifies a CICS object. The new PLTSDI option can be configured on the interface definition, EZACICD TYPE=CICS. If PLTSDI is not specified then a deferred shutdown is performed.

The EZAC transaction is a panel-driven interface that lets you add, delete, or modify the configuration file. The DEFINE function is used to create CICS objects and their listener objects. The PLTSDI can be configured on the definition of a CICS object.

In prior releases, you can specify a shutdown method of deferred or immediate when using the program link EZACIC20. In V1R9 you now have the option of querying the PLTSDI configuration to determine which shutdown method to be used. You still have the option of selecting a deferred or immediate shutdown method. If you have created your own maintenance transaction then you can specify that the EZACIC20 program query the PLTSDI configuration option by setting the P20TYPE field in the P20PARAM COMMAREA to the value of 'Q' or P20TYPEQ before linking to EZACIC20.

CICS definition - EZACICD macro

EZACICD TYPE=CICS,	CICS record definition	X
APPLID=CICSPROD,	APPLID of CICS region not using OTE	X
TCPADDR=TCPIP,	Job/Step name for TCP/IP	X
PLTSDI=YES,	PLT shutdown method is immediately	X
NTASKS=20,	Number of subtasks	X
DPRTY=0,	Subtask dispatch priority difference	X
CACHMIN=15,	Minimum refresh time for cache	X
CACHMAX=30,	Maximum refresh time for cache	X
CACHRES=10,	Maximum number of resident resolvers	X
ERRORTD=CSMT,	Transient data queue for error msgs	X
TCBLIM=0,	Open API TCB Limit	X
OTE=NO,	Use Open Transaction Environment	X
TRACE=NO,	Trace CICS Sockets	X
APPLDAT=YES,	Register Application Data	X
SMSGSUP=NO,	STARTED Messages Suppressed?	X
TERMLIM=100	Subtask Termination Limit	

The PLTSDI configuration option has been added to the interface definition, EZACICD TYPE=CICS. The IP CICS Sockets configuration macro, EZACICD, is designed to support the configuration of the IP CICS Socket interface and listeners exploiting that interface. Regenerate the IP CICS Socket configuration dataset, EZACONFG, with the new PLTSDI configuration option.

The EZACICD macro will issue this MNOTE when validating the PLTSDI configuration option for a value other than YES or NO:

```
MNOTE 12, 'INVALID VALUE SPECIFIED FOR PLTSDI, GENERATION  
TERMINATED'.
```

CICS definition - EZAC transaction

```

EZAC,DEFine,CICS                                APPLID = CICS1A

Overtype to Enter

APPLID      ==> CICS1A                          APPLID of CICS System
TCPADDR     ==> TCPIP                            Name of TCP Address Space
NTASKS      ==> 020                              Number of Reusable Tasks
DPRTY       ==> 000                              DPRTY Value for ATTACH
CACHMIN     ==> 015                              Minimum Refresh Time for Cache
CACHMAX     ==> 030                              Maximum Refresh Time for Cache
CACHRES     ==> 010                              Maximum Number of Resolvers
ERRORTD     ==> CSMT                             TD Queue for Error Messages
SMSGSUP     ==> NO                               Suppress Task Started Messages
TERMLIM     ==> 100                              Subtask Termination Limit
TRACE       ==> YES                              Trace CICS Sockets
OTE         ==> NO                               Open Transaction Environment
TCBLIM     ==> 00000                             Number of Open API TCBS
PLTSDI     ==> NO                             CICS PLT Shutdown Immediately
APPLDAT     ==> NO                               Register Application Data
  
```

PF 3 END

12 CNCL

21

CICS sockets enhancements

© 2008 IBM Corporation

The online configuration option also supports the PLTSDI configuration option. The EZAC transaction will update the configuration contained in the EZACONFG VSAM dataset but not the original EZACICD macro. Ensure you mirror any updates against your sites EZACICD configuration macros.

The EZAC configuration transaction will issue the followings message when validating the PLTSDI configuration option for a value other than YES or NO.

Incorrect or no entry for PLTSDI field, re-enter parameters or press ENTER to continue.

Things to think about

- If the configuration is not changed then the PLT will continue to shutdown using the deferred shutdown method.
- Ensure user written listeners and socket programs are able to support the immediate shutdown method.
 - ▶ When OTE=NO then tasks with blocking socket commands will have their MVS subtask posted so the CICS task can subsequently end.
 - ▶ When OTE=YES then tasks with blocking sockets will be FORCE PURGED.
- If you have written a program which makes use of any of the external IP CICS Socket control blocks then you must reassemble that program to pick up the latest changes. The assembler macros are:
 - ▶ EZACICA – IP CICS Socket control blocks
 - ▶ EZACICSX – Listener security/user exit COMMAREA layout

22

CICS sockets enhancements

© 2008 IBM Corporation

If PLTSDI is not specified then the PLT program will continue to shutdown using the deferred shutdown method. When using the immediate shutdown method ensure user written programs can handle the responses to end or be cancelled.

When OTE=YES is specified, then the IP CICS socket interface exploits the Open Transaction Environment (OTE) to enable the IP CICS sockets task-related user exit to invoke and return from TCP/IP without switching TCBs.

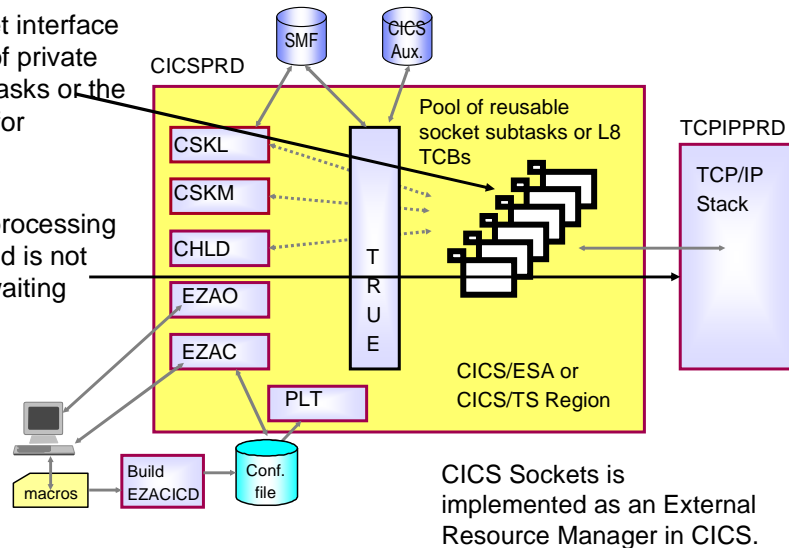
Recompile any user written socket programs or listeners which use the IP CICS Socket macros.

**Identify the task's TCP/IP stack job name in the
CEMT INQUIRE TASK output**

CICS Tasks using the IP CICS Socket API will show the job name of the TCP/IP stack processing the socket call.

Background information

- The IP CICS Socket interface uses either a pool of private MVS reusable subtasks or the CICS/TS L8 TCBs for processing socket commands.
- The TCP/IP stack processing the socket command is not identified with the waiting task.



CICS Sockets is implemented as an External Resource Manager in CICS.

The IP CICS Socket interface uses two distinct tasking methods to process socket commands. A MVS subtask and a L8 TCB provided by CICS/TS when using the open transaction environment.

CICS/TS Open Transaction Environment (OTE) introduces a new task of Task Control Blocks (TCBs) called open TCB, which can be used by applications. It is characterized by the fact that it is assigned to a CICS task for the life of the CICS task. Multiple OTE TCBs can run concurrently in CICS.

In the CICS/TS environment a socket command is considered a non-CICS resource. CICS/TS supports non-CICS resource through a task related user exit (TRUE). So when using the IP CICS Socket API then a TRUE will be driven to support the non-CICS resource. The socket command will be processed on a TCB other than the CICS main TCB (QR TCB) to prevent the entire CICS workload from waiting when processing blocking socket commands (such as RECV). This TCB used for processing the socket command will be either one from the pool of private MVS subtasks (as established by the NTASKS IP CICS Socket configuration option) or one of the L8 TCBs supplied when exploiting CICS/TS open transaction environment (OTE).

Problem - Cannot tell what task is waiting on

- IP CICS socket enabled programs are put into an external CICS wait while TCP/IP is processing the socket call.
 - ▶ OTE=NO configured
- When this occurs the CEMT INQ TASK transaction shows the task as being in USERWAIT condition
 - ▶ No information provided to identify the name of the TCP/IP stack

The CICS supplied transaction, CEMT, can be used by an operator to inquire about and change the values of parameters used by CICS, alter the status of the system resources, terminate tasks, and shut down the CICS system. The CEMT INQUIRE TASK transaction returns information about user tasks. Only information about user tasks can be displayed or changed. The Htype field displays the reason why the task is suspended. In prior releases, when Htype indicates a USERWAIT condition, then there is no further information to indicate which TCP/IP stack is associated with this task.

Solution - Associate the USERWAIT with a stack

- The USERWAIT condition will contain the job name of the TCP/IP stack with which the IP CICS Socket enable program has affinity
 - ▶ The suspend value (Hvalue) contains the TCP/IP job name

The user wait condition will now be associated with the job name of the TCP/IP stack which the CICS transaction has affinity. Tasks processing a socket command can then be associated with the TCP/IP stack job name processing that socket command.

TCP/IP JOB NAME

- Task's TCP affinity defined using:

EZACICD TYPE=CICS,	CICS record definition	X
APPLID=CICSPROD,	APPLID of CICS region not using OTE	X
TCPADDR=TCPIP,	Job/Step name for TCP/IP	X
PLTSDI=YES,	PLT shutdown method is immediately	X
NTASKS=20,	Number of subtasks	X
DPRTY=0,	Subtask dispatch priority difference	X
CACHMIN=15,	Minimum refresh time for cache	X
CACHMAX=30,	Maximum refresh time for cache	X
CACHRES=10,	Maximum number of resident resolvers	X
ERRORTD=CSMT,	Transient data queue for error msgs	X
TCBLIM=0,	Open API TCB Limit	X
OTE=NO,	Use Open Transaction Environment	X
TRACE=NO,	Trace CICS Sockets	X
APPLDAT=YES,	Register Application Data	X
SMSGSUP=NO,	STARTED Messages Suppressed?	X
TERMLIM=100	Subtask Termination Limit	

The IP CICS Socket interface TCPADDR configuration option was not changed in z/OS V1R9 but this shows the TCP/IP job name with which IP CICS Socket enabled transaction will have affinity.

CEMT INQ TASK

- CEMT INQ TASK command showing suspend type:

```
INQ TASK
STATUS:  RESULTS - OVERTYPE TO MODIFY
Tas(0000036) Tra(CSKL)          Sus Tas Pri( 255 )
      Sta(S ) Use(CICSUSER) Uow(BFB5C69A59B93481) Hty(USERWAIT)
Tas(0000037) Tra(CEMT) Fac(211A) Run Ter Pri( 255 )
      Sta(TO) Use(CICSUSER) Uow(BFB5C6E942E9FB61)

                                SYSID=CICT APPLID=CICS1A
RESPONSE: NORMAL                                TIME: 12.05.55  DATE:
11.15.06
PF 1 HELP          3 END          5 VAR          7 SBH 8 SFH 9 MSG 10 SB 11 SF
```


When issuing the CEMT INQ TASK transaction a user will see the suspend type (Hty) as USERWAIT. This indicates that task is processing a socket command and is waiting for its processing to complete in the TCP/IP stack.

CEMT INQ TASK


- CEMT INQ TASK command showing suspend value

```
INQ TASK
RESULT - OVERTYPE TO MODIFY
Task(0000036)
Tranid(CSKL)
...
...
Userid(CICSUSER)
Uow(BFB5C69A59B93481)
Htype(USERWAIT)
Hvalue(TCPIP)
Htime(000225)
Indoubt(Backout)
Indoubtwait(Wait)
...
```

Using the question mark command, you can drill down to further CICS task attributes and see the suspend value (Hvalue) of the job name of the TCP/IP stack with which the task has affinity.

IBM Software Group 

Automatic IP CICS socket support

 30
CICS sockets enhancements © 2008 IBM Corporation

The IP CICS Socket Interface and listener can be configured to automatically exploit the ioctl used to register application data.

Problem - Cannot correlate connection with CICS task

- Needed: the ability to quickly identify TCP connections for IP CICS Socket applications
 - ▶ Listener, child server, and client transactions
- The new identifying data should be provided with existing connection information records by the provided management interfaces
 - ▶ Netstat
 - ▶ SMF
 - ▶ NMI

It is very difficult to correlate a connection in a Netstat, SMF or NMI report with the actual CICS/TS task. Users have requested that a mechanism be provided such that they can easily correlate a connection with an actual CICS/TS task.

Solution - Automatic IP CICS support

- Automatically register application data
 - ▶ New configuration option APPLDAT=YES|NO for the interface
 - ✓ Can be specified on the interface definition, EZACICD TYPE=CICS
 - ✓ Can be specified using the transaction EZAC,DEFine,CICS
 - ▶ New configuration option LAPPLD=INHERIT|YES|NO for the listener
 - ✓ Can be specified on the interface definition, EZACICD TYPE=LISTENER
 - ✓ Can be specified using the transaction, EZAC,DEFine,LISTENER
 - ▶ Registration of data can be requested by the Listener Security/User Exit
 - ▶ Dynamically query and change the settings of APPLDAT and LAPPLD
 - ✓ Query the status of the interface, EZAO,INQUIRE,CICS
 - ✓ Controls whether the interface registers data, EZAO,SET,CICS
 - ✓ Query the LAPPLD setting, EZAO,INQUIRE,LISTENER
 - ✓ Controls whether the listener registers data, EZAO,SET,LISTENER
- Exploits the “Enable Application identifier in NMI, SMF and Netstat” support in z/OS V1R9

32

CICS sockets enhancements

© 2008 IBM Corporation

Enabling the IP CICS Socket interface and listener with the ability to automatically register application data with its TCP connections. The automatic registration of the application data occurs at the IP CICS Socket Interface listeners, clients and child server programs. The application data used will be enough to direct the user back to the CICS/TS region where the connection originates. The, tools such as the CICS/TS master terminal transaction, CEMT, can be used to see the CICS attributes.

The IP CICS socket interface supports a new configuration option, APPLDATA, designed to cause application data to be automatically registered before LISTEN and GIVESOCKET or after CONNECT and TAKESOCKET. Valid values for this option are Yes or No. The default is No.

You may want to control listeners apart from the interface. The LAPPLD listener configuration option will allow you to override the APPLDAT configuration option. The LAPPLD option determines whether the IP CICS Sockets listener will automatically register IP CICS Sockets unique application data against its TCP connections. Valid values are Yes, No or the default of INHERIT. The value of INHERIT causes the listener to use the value specified for APPLDAT. LAPPLD affects the SIOCSAPPLDATA IOCTL being issued by the IBM Listener and therefore only affects the IOCTL surrounding the GIVESOCKET socket call.

The IP CICS Sockets security/user exit COMMAREA provides a one byte field that can be used by the security exit to indicate that the listener will register application data for the accepted socket to be given. Changes to this field will be honored by the listener. The character value of 0 indicates that no application data is being registered. The character value of 1 indicates that application data is being registered. If a security exit is used then it has the ability to override whether the listener will register application data for the accepted socket to be given.

To summarize the interface setting, APPLDAT, determines whether application data is processed for the overall interface. The listener setting, LAPPLD, will either supersede or inherit the value established by APPLDAT. And the security exit will reflect the listener's LAPPLD setting and may alter that setting upon return to the listener.

The IP CICS Socket operator transaction, EZAO, is enhanced to allow the APPLDAT and LAPPLD settings to be dynamically changed without stopping and restarting the listener. This will be handy for users who only want to register application data during problem analysis. When EZAO is done for a listener, you can select from a list of active listeners or specify a specific listener. Optionally you can do a query or set for a specific listener by specifying the transaction ID.

When enabled, the interface or listener (or both) can register application data before or after specific socket commands. Application data is registered after CONNECT or connect(), before GIVESOCKET (IBM Listener only), before LISTEN or listen(), and after TAKESOCKET or takesocket().

z/OS V1R9 Communication Server introduced a new function to enable application identifiers in NMI, SMF and the Netstat report. IP CICS Sockets is exploiting that function. Refer to the Communications Server New Function Summary for detail information on the new function.

CICS definition - EZACICD macro

EZACICD TYPE=CICS,	CICS record definition	X
APPLID=CICSPROD,	APPLID of CICS region not using OTE	X
TCPADDR=TCPIP,	Job/Step name for TCP/IP	X
PLTSDI=YES,	PLT shutdown method is immediately	X
NTASKS=20,	Number of subtasks	X
DPRTY=0,	Subtask dispatch priority difference	X
CACHMIN=15,	Minimum refresh time for cache	X
CACHMAX=30,	Maximum refresh time for cache	X
CACHRES=10,	Maximum number of resident resolvers	X
ERRORTD=CSMT,	Transient data queue for error msgs	X
TCBLIM=0,	Open API TCB Limit	X
OTE=NO,	Use Open Transaction Environment	X
TRACE=NO,	Trace CICS Sockets	X
APPLDAT=YES,	Register Application Data	X
SMSGSUP=NO,	STARTED Messages Suppressed?	X
TERMLIM=100	Subtask Termination Limit	

The APPLDAT configuration option can be specified on the interface definition, EZACICD TYPE=CICS. The IP CICS Sockets configuration macro, EZACICD, is designed to support the configuration of the IP CICS Socket interface and listeners exploiting that interface. Regenerate the IP CICS Socket configuration dataset, EZACONFG, with the new APPLDAT configuration option.

The following Assembler MNOTE will be produced by the EZACICD macro when processing the APPLDAT configuration option for a value other than YES or NO.

```
MNOTE 12, 'INVALID VALUE SPECIFIED FOR APPLDAT, GENERATION  
TERMINATED' .
```

The following Assembler MNOTE will be produced by the EZACICD macro when the APPLDAT is not specified.

```
MNOTE 0, 'APPLDAT NOT SPECIFIED BUT DEFAULTS TO NO'
```

CICS definition - EZAC transaction

```

EZAC,DEFine,CICS                                APPLID = CICS1A

Overtype to Enter

APPLID      ==> CICS1A                APPLID of CICS System
TCPADDR     ==> TCPIP                 Name of TCP Address Space
NTASKS      ==> 020                   Number of Reusable Tasks
DPRTY       ==> 000                   DPRTY Value for ATTACH
CACHMIN     ==> 015                   Minimum Refresh Time for Cache
CACHMAX     ==> 030                   Maximum Refresh Time for Cache
CACHRES     ==> 010                   Maximum Number of Resolvers
ERRORTD     ==> CSMT                 TD Queue for Error Messages
SMSGSUP     ==> NO                   Suppress Task Started Messages
TERMLIM     ==> 100                   Subtask Termination Limit
TRACE       ==> YES                   Trace CICS Sockets
OTE         ==> NO                   Open Transaction Environment
TCBLIM     ==> 00000                 Number of Open API TCBS
PLTSDI     ==> NO                   CICS PLT Shutdown Immediately
APPLDAT    ==> NO                   Register Application Data

PF 3 END                                         12 CNCL

```

The APPLDAT configuration option can also be specified using the online configuration transaction, EZAC. The EZAC transaction will update the configuration contained in the EZACONFG VSAM dataset but not the original EZACICD macro. Ensure you mirror any updates against your sites EZACICD configuration macros.

The following message will be produced by the EZAC transaction when processing the APPLDAT configuration option for a value other than YES or NO.

Incorrect or no entry for APPLDAT field

Re-enter parameters or press ENTER to continue.

Listener definition - EZACICD macro

EZACICD TYPE=LISTENER,	Listener record definition	X
FORMAT=STANDARD,	Standard Listener	X
APPLID=CICSPROD,	Applid of CICS region	X
TRANID=CSKL,	Transaction name for Listener	X
PORT=3010,	Port number for Listener	X
IMMED=YES,	Listener starts up at initialization?	X
BACKLOG=20,	Backlog value for Listener	X
NUMSOCK=50,	# of sockets supported by Listener	X
MINMSGL=4,	Minimum input message length	X
ACCTIME=30,	Timeout value for Accept	X
GIVTIME=30,	Timeout value for Givesocket	X
REACTIME=30,	Timeout value for Read	X
RTYTIME=15,	Wait 15 seconds for TCP to come back	X
TRANTRN=YES,	Is TRANUSR=YES conditional?	X
TRANUSR=YES,	Translate user data?	X
SECEXIT=EZACICSE,	Name of security exit program	X
LAPPLD=YES,	Register application data	X
WLMGN1=WLMGRP01,	WLM group name 1	X
WLMGN2=WLMGRP02,	WLM group name 2	X
WLMGN3=WLMGRP03	WLM group name 3	

The LAPPLD configuration option can be specified on the interface definition, EZACICD TYPE=LISTENER. The IP CICS Sockets configuration macro, EZACICD, is designed to support the configuration of the IP CICS Socket interface and listeners exploiting that interface. Regenerate the IP CICS Socket configuration dataset, EZACONFG, with the new LAPPLD configuration option.

The following Assembler MNOTE will be produced by the EZACICD macro when processing the LAPPLD configuration option for a value other than INHERIT, YES or NO.

```
MNOTE 12,'INVALID VALUE SPECIFIED FOR LAPPLD, GENERATION  
TERMINATED'
```

The following Assembler MNOTE will be produced by the EZACICD macro when the LAPPLD is not specified.

```
MNOTE 0,'LAPPLD NOT SPECIFIED BUT INHERITS INTERFACE APPLDAT'
```

Standard listener definition - EZAC transaction

EZAC,DEFine,LISTENER (standard listener. screen 1 of 2) APPLID = CICS1A

Overtyp e to Enter

APPLID	====> CICS1A	APPLID of CICS System
TRANID	====> CSKL	Transaction Name of Listener
PORT	====> 03010	Port Number of Listener
AF	====> INET	Listener Address Family
IMMEDIATE	====> YES	Immediate Startup Yes No
BACKLOG	====> 020	Backlog Value for Listener
NUMSOCK	====> 050	Number of Sockets in Listener
ACCTIME	====> 060	Timeout Value for ACCEPT
GIVTIME	====> 000	Timeout Value for GIVESOCKET
REATIME	====> 000	Timeout Value for READ
RTYTIME	====> 015	Stack Connection Retry Time
LAPPLD	====> INHERIT	Register Application Data

Verify parameters, press PF8 to go to screen 2

PF 3 END

8 NEXT

12 CNCL

36

CICS sockets enhancements

© 2008 IBM Corporation

The LAPPLD configuration option can also be specified using the online configuration transaction, EZAC. The EZAC transaction will update the configuration contained in the EZACONFG VSAM dataset but not the original EZACICD macro. Ensure you mirror any updates against your sites EZACICD configuration macros. Shown here is the panel for the standard listener. The LAPPLD configuration option can also be specified for the enhanced listener.

The following message will be produced by the EZAC transaction when processing the LAPPLD configuration option for a value other than INHERIT, YES or NO.

Incorrect or no entry for LAPPLD field

Re-enter parameters or press ENTER to continue.

Query or set interface using EZAO transaction

Query system setting

```

EZAO,INQUIRE,CICS                                APPLID = CICS1A

TRACE      ==> YES                               Trace CICS Sockets
MAXOPENTCBS ==> 00160                             CICS Open API, L8, TCB Limit
ACTOPENTCBS ==> 00001                             Active CICS Open API, L8, TCBS
TCBLIM     ==> 00000                             Open API TCB Limit
ACTTCBS    ==> 00000                             Number of Active Open API TCBS
QUEUEDEPTH ==> 00000                             Number of Suspended Tasks
SUSPENDHWM ==> 00000                             Suspended Tasks HWM
APPLDAT    ==> YES                               Register Application Data

PF 3 END                                         12 CNCL

```

Set system setting

```

EZAO,SET,CICS                                APPLID = CICS1A

Overtyp e to Enter

TRACE      ==> YES                               Trace CICS Sockets
TCBLIM     ==> 00000                             Open API TCB Limit
APPLDAT    ==> YES                               Register Application Data

PF 3 END                                         12 CNCL

```

The operator can query the status of the interface automatically registering application data by using the EZAO,INQUIRE,CICS transaction. The operator can also control whether the interface automatically registers application data by using the EZAO,SET,CICS transaction.

The following messages can be produced by the EZAO transaction when processing the APPLDAT configuration option.

Error on APPLDAT Entry, Please Re-Enter (Re-enter parameters or press ENTER to continue.)

CICS Sockets application data registration is already YES (Press ENTER to continue or PF3 to exit)

CICS Sockets application data registration is already NO (Press ENTER to continue or PF3 to exit)

Automatic application data registration is now enabled

Automatic application data registration is now disabled

Query or set a specific listener using EZAO transaction

- Query a specific listener setting

```

EZAO,INQUIRE,LISTENER(CSKL)                                APPLID = CICS1A
LAPPLD      ==> YES      Register Application Data
PF 3 END                                          12 CNCL
  
```

- Set a specific listener setting

```

EZAO,SET,LISTENER(CSKL)                                    APPLID = CICS1A
LAPPLD      ==> YES      Register Application Data
PF 3 END                                          12 CNCL
  
```

The operator can query the status of the listener automatically registering application data by using the EZAO,INQUIRE,LISTENER transaction. The operator can control whether the listener automatically registers application data.

The following messages will be produced by the EZAO transaction when processing the LAPPLD configuration option.

Error on LAPPLD Entry, Please Re-Enter (Re-enter parameters or press ENTER to continue)

Listener application data registration is already YES (Press ENTER to continue or PF3 to exit)

Listener application data registration is already NO (Press ENTER to continue or PF3 to exit)

Automatic application data registration is now enabled

Automatic application data registration is now disabled

Query or set a specific listener using EZAO transaction

- Select an active listener to query

```

EZAO,INQUIRE,LISTENER                                APPLID = CICS1A

Choose a listener transaction:

Sel  Tran Task#  Type    Date    Time    Message
----  ---  ---  ---  ---  ---
CSKL 0000464  STANDARD 11/13/06 14:30:39 ACTIVE
CSKM 0000465  ENHANCED 11/13/06 14:30:41 ACTIVE

Select a listener to continue

PF 3 END  7 UP  8 DOWN  9 TOP  10 BOTTOM  12 CNCL  ENTER SELECT
  
```

- Select an active listener to set

```

EZAO,SET,LISTENER                                APPLID = CICS1A

Choose a listener transaction:

Sel  Tran Task#  Type    Date    Time    Message
----  ---  ---  ---  ---  ---
CSKL 0000464  STANDARD 11/13/06 14:30:39 ACTIVE
CSKM 0000465  ENHANCED 11/13/06 14:30:41 ACTIVE

Select a listener to continue

PF 3 END  7 UP  8 DOWN  9 TOP  10 BOTTOM  12 CNCL  ENTER SELECT
  
```



The EZAO operator transaction now supports a list of active listeners to choose from when querying the status of a listener automatically registering application data. The operator transaction also supports a list of active listener to choose from when controlling whether a listener automatically registers application data.

The following messages will be produced by the EZAO transaction when selecting active listeners.

Select a listener to continue

No listeners active

Listener is no longer active, select another listener

At the top of data

At the bottom of data

Things to think about

- When any of the following are true then an extra socket call will be automatically issued on behalf of the task:
 - ▶ APPLDAT=YES
 - ▶ LAPPLD=YES
 - ▶ LAPPLD=INHERIT and APPLDAT=YES
- Reassemble any user written programs using any IP CICS Socket macros
- SIOCSAPPLDATA IOCTL socket command fails has no impact on other processing

Realize that an extra pass to the TCP/IP stack will be generated on behalf of applications using the interface and listener.

If you use the IP CICS Socket control blocks then re-compile your programs to get the latest versions of the shipped macros.

The EZY1391E message will be issued when the SIOCSSAPPLDATA IOCTL socket command returns in error. This failure will not impact user's socket calls made before or afterwards.

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_CICSEnh.ppt

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



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

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:

CICS CICS/ESA IBM RACF 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 NON-INFRINGEMENT. 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 2008. 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.

