

CICS Transaction Gateway



Messages

Version 8.0

CICS Transaction Gateway



Messages

Version 8.0

Note!

Before using this information and the product it supports, be sure to read the general information under "Notices" on page 159.

This edition applies to Version 8.0 of IBM CICS Transaction Gateway for Multiplatforms, program numbers 5655-W10 and 5724-I81 and CICS Transaction Gateway Desktop Edition, program number 5725-B65. It will also apply to all subsequent versions, releases, and modifications until otherwise indicated in new editions.

This edition replaces SC34-7061-01.

© Copyright IBM Corporation 1994, 2011.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

About this book.	v	Accessibility	129
Conventions and terminology used in this book	v	Glossary	131
Prerequisite and related information	v	Index	157
Messages	1	Notices	159
Product library and related literature	125	Trademarks	160
CICS Transaction Gateway books	125	Sending your comments to IBM	163
Sample configuration documents.	125		
IBM Redbooks publications	125		
Other useful information	126		
CICS Transaction Server publications	126		

About this book

This book lists the error and warning messages for the CICS® Transaction Gateway and CICS Transaction Gateway Desktop Edition.

It is intended for use as a quick reference, with messages organized in alphanumeric sequence. Each message entry gives the message identifier, message text, and further diagnostic and explanatory information.

Conventions and terminology used in this book

The terminology in this book should be familiar to anyone who has used CICS and the supported operating systems.

Prerequisite and related information

Some messages refer to several non-IBM® books; these mainly relate to the platform on which the product is running and it is assumed that these books are available to you.

Messages

CCL0001I CICS Client Daemon Messages

Explanation: Information message

**CCL0002I (C) Copyright IBM Corporation
Copyright dates . All rights reserved.**

Explanation: Information message

CCL0003I Press Enter to continue

Explanation: Information message

CCL0004I Press Enter to continue or Esc to cancel

Explanation: Information message

CCL0005I OK

Explanation: Information message

CCL0006I Cancel

Explanation: Information message

CCL0007I Help

Explanation: Information message

CCL0008I More:

Explanation: Information message

CCL0009I User ID:

Explanation: Information message

CCL0010I Password:

Explanation: Information message

CCL0011I CICS Emulator Messages

Explanation: Information message

CCL0021I Server Name:

Explanation: Information message

CCL0037I New Password:

Explanation: Information message

CCL0038I Confirm Password:

Explanation: Information message

CCL0049I Refer to the product documentation for copyright statements

Explanation: Information message

CCL0200I * IBM CICS Transaction Gateway Client log for *platform version* Build Level *level* *****

Explanation: Information message

CCL0201I * IBM CICS Transaction Gateway Desktop Edition Client log for *platform version* Build Level *level* *****

Explanation: Information message

CCL0501I Command options are:

Explanation: Information message

CCL0503I Issue 'INSERT-0 -?' to list command options

Explanation: Information message

CCL0510E Command option *option* is invalid

CCL0511E • CCL0519E

Explanation: An invalid command option was specified on the command line.

System action: The command line application terminates.

User response: Retype the command using the correct options.

CCL0511E Command option *option* must specify a value

Explanation: A command line application was run with no value supplied for the named required option.

System action: The command line application terminates.

User response: Retype the command using the correct options.

CCL0512E Command option *option* cannot be used more than once

Explanation: The named option was used more than once on the command line. This is invalid.

System action: The command line application terminates.

User response: Retype the command using the correct options.

CCL0513E Command option *option1* cannot be used with option *option2*

Explanation: The two named options, used on the command line, were incompatible.

System action: The command line application terminates.

User response: Retype the command using the correct options.

CCL0514E Command option *option1* or *option2* must be provided

Explanation: The command line did not include one of the named required options.

System action: The command line application terminates.

User response: Retype the command using the correct options.

CCL0515E Command option *option1* can only be used with *option2*

Explanation: *option1* was specified on the command line, but *option2* was not also specified. This is invalid.

System action: The command line application terminates.

User response: Retype the command using the correct options.

CCL0516E Command option *option* is invalid

Explanation: An invalid command option was specified on the command line.

System action: The command line application terminates.

User response: Retype the command using the correct options.

CCL0517E Command option *option* must specify a value

Explanation: A command line application was run with no value supplied for the named required option.

System action: The command line application terminates.

User response: Retype the command using the correct options.

CCL0518E Command option *option* cannot be used more than once

Explanation: The named option was used more than once on the command line. This is invalid.

System action: The command line application terminates.

User response: Retype the command using the correct options.

CCL0519E Command option *option1* cannot

be used with option *option2*

Explanation: The two named options, used on the command line, were incompatible.

System action: The command line application terminates.

User response: Retype the command using the correct options.

CCL0520E **Command option *option1* or *option2* must be provided**

Explanation: The command line did not include one of the named required options.

System action: The command line application terminates.

User response: Retype the command using the correct options.

CCL0521E **Command option *option1* can only be used with *option2***

Explanation: *option1* was specified on the command line, but *option2* was not specified. This is invalid.

System action: The command line application terminates.

User response: Retype the command using the correct options.

CCL0522E **Command option 'B' requires that 'D' is also specified**

Explanation: The memory trace option 'B' was specified, without specifying that trace is to be switched on with 'D'.

System action: Tracing is not activated, and this error is returned to the user.

User response: Retype the command, specifying 'D' in addition to 'B'.

CCL1001I **CICSFTRC - CICS Client Binary Trace Formatter Program**

Explanation: Information message

CCL1002I **Command options are:**

Explanation: Information message

CCL1003I **-M=components - Format trace for specified components**

Explanation: Information message

CCL1004I **-W[=filename] - Format a wrapping trace**

Explanation: Information message

CCL1005I **-N - Perform indentation formatting**

Explanation: Information message

CCL1006I **-D - Insert detailed formatting into trace file**

Explanation: Information message

CCL1007I **-I=filename - Specify the input (binary) filename**

Explanation: Information message

CCL1008I **-O=filename - Specify the output (text) filename**

Explanation: Information message

CCL1009I **-F - Overwrite any existing files**

Explanation: Information message

CCL1010I **-S - Produce summary trace file**

Explanation: Information message

CCL1020I **CICSFTRC has ended**

Explanation: Information message

CCL1021E **Unable to write to text file *filename***

Explanation: The binary trace formatter is

unable to write any further information to the Client trace file during conversion of the binary trace.

System action: The binary trace formatter terminates.

User response: Determine the cause of the error. It may be that the disk which the binary trace formatter is writing to is full, or the disk might be write-protected.

CCL1022E Unable to read from binary file
filename (offset offset)

Explanation: The binary trace formatter tried to format the binary trace file, but could not read the whole file. *offset* is the number of bytes between the start of the trace file and the error.

System action: The binary trace formatter terminates.

User response: Check that the binary trace file has not been deleted or truncated during conversion, and that it is not read-protected.

CCL1023E Unable to open file *filename*

Explanation: The binary trace formatter is unable to open the specified file for reading or writing as required.

System action: The binary trace formatter terminates.

User response: If the file that is being opened is the binary trace file, check that the file exists and that it is not read-protected. If the file that is being opened is the output text file, make sure that the disk on to which the file is being written is not write-protected.

CCL1024E Unable to interpret all bytes of binary file *filename (offset offset)*

Explanation: The binary trace formatter cannot process the binary trace file *filename*, because the file contains some incorrect data structures. *offset* is the number of bytes between the start of the trace file and the error.

System action: The binary trace formatter terminates.

User response: Make sure that the file being read is a valid binary trace file, and that it was created with the version of the Client daemon currently installed on your system. If the trace file was transferred from another computer, check that it was not damaged in transit. If you still get the same error, delete the binary trace file and run the trace again.

CCL1025E Not enough memory to convert binary trace file

Explanation: The binary trace formatter has run out of free memory whilst processing the binary trace.

System action: The binary trace formatter terminates.

User response: Increase the amount of available memory in the system by shutting down applications that are not needed, and then re-run the binary trace formatter. You can reduce the amount of memory that the binary trace formatter needs by specifying on the command line the maximum amount of data to convert at any one time (refer to the documentation for further information on how to do this).

CCL1027E Invalid binary trace file *filename (version number = number)*

Explanation: The binary trace formatter detected an invalid version number in the binary trace file.

System action: The binary trace formatter terminates without performing any further processing.

User response: Make sure that the file you are trying to convert is a valid binary trace file and that you are using the most recent version of the binary trace formatter that is available on your system.

CCL1028E Unable to access file *filename (offset = offset)*

Explanation: The binary trace formatter cannot access the section of the binary trace file which is referred to by *filename* . *offset* represents the offset in bytes, from the start of the trace file, where the error occurred.

System action: The binary trace formatter terminates.

User response: Check that the user has access to the binary trace file. Check that the binary trace file has not been truncated.

CCL1029I Trace formatting cannot continue

Explanation: Information message

CCL1030I Warning: component *component* not found in binary trace

Explanation: Information message

CCL1031I INSERT-0 % formatted

Explanation: Information message

CCL1032I Do you want to overwrite *filename* [Y|N]?

Explanation: Information message

CCL1033E Trace file *filename* contains no data

Explanation: The trace file that has been specified does not contain any trace data. This can occur if memory mapped trace wrapping is being used, and the trace was not on for long enough for all of the trace files to be used.

System action: The binary trace formatter will terminate. An empty text output file will be produced.

User response: Either format the entire wrapping trace using the 'w' option, or choose another wrapping file to format.

CCL1034E No trace files in the wrapping sequence contain any data

Explanation: Memory mapped wrapping trace

is being used. None of the trace files in the sequence contain any data.

System action: The binary trace formatter will terminate. An empty text output file will be produced.

User response: If the problem persists contact your service organization.

CCL1035E Trace file *filename* is out of sequence

Explanation: A memory mapped trace is being formatted. File *filename* is not in the correct point in the sequence.

System action: The binary trace formatter will terminate. No formatted output will be produced.

User response: If the files have been renamed, verify that they have been given the correct sequence number in the file type suffix. If files have been transferred from another machine, verify that they have not been corrupted.

CCL1041E Error in function *function* (Error Code = *error*)

Explanation: A system or internal product function failed.

System action: The message is written to the error log.

User response: If the problem persists, contact your service organization.

CCL1046E Error in function *function* (Error Code = *error*)

Explanation: A system or internal product function failed.

System action: The message is written to the error log. The function name and error code are logged.

User response: If the problem persists, contact your service organization.

CCL1047E **Error in the EPI (Function = *function*, Error Code = *Error*, TermNumber = *Termnumber*)**

Explanation: An internal EPI error has occurred.

System action: The message is written to the error log. The function name and error code are logged, together with the TermNumber.

User response: If the problem persists, contact your service organization.

CCL1048E **Error in function *function* (Error Code = *error*)**

Explanation: An system or internal Client function failed.

System action: The message is written to the error log. The function name and error code are logged.

User response: If the problem persists, contact your service organization.

CCL1100I **The following list shows what components can be traced**

Explanation: Information message

CCL1101I **An 'X' indicates which components are enabled**

Explanation: Information message

CCL1102I *INSERT-0 [INSERT-1] CICSCLI Command Process*

Explanation: Information message

CCL1104I *INSERT-0 [INSERT-1] Protocol Drivers (Level 1)*

Explanation: Information message

CCL1105I *INSERT-0 [INSERT-1] Protocol Drivers (Level 2)*

Explanation: Information message

CCL1106I *INSERT-0 [INSERT-1] API (Level 1)*

Explanation: Information message

CCL1107I *INSERT-0 [INSERT-1] API (Level 2)*

Explanation: Information message

CCL1108I *INSERT-0 [INSERT-1] Client Daemon*

Explanation: Information message

CCL1109I *INSERT-0 [INSERT-1] Terminal Emulators*

Explanation: Information message

CCL1111I *INSERT-0 [INSERT-1] C++ Class Libraries*

Explanation: Information message

CCL1112I *INSERT-0 [INSERT-1] Workload Manager*

Explanation: Information message

CCL1114I *INSERT-0 [INSERT-1] Client Daemon Statistics*

Explanation: Information message

CCL1115I *INSERT-0 [INSERT-1] All Components*

Explanation: Information message

CCL1116I *INSERT-0 [INSERT-1] Interprocess Communication (Level 1)*

Explanation: Information message

CCL1117I *INSERT-0 [INSERT-1] Interprocess Communication (Level 2)*

Explanation: Information message

CCL1120I Trace has not been started

Explanation: Information message

CCL1121I Component trace started to:
component list

Explanation: Information message

CCL1122E Re-type list of components as the
current list is invalid

Explanation: You have specified a list of components on the command line, but one or more of the components in the list is invalid.

System action: The request that required a list of components to be specified is ignored.

User response: Refer to the Administration book for a list of valid components and retype the request.

CCL1123I Reading binary trace file *filename*

Explanation: Information message

CCL1124I Reading binary trace files
filename1 and *filename2*

Explanation: Information message

CCL1125I Generating text trace file *filename*

Explanation: Information message

CCL1126I Failed to make new trace
component list persistent

Explanation: Information message

CCL1127I Updated trace component list to:
list

Explanation: Information message

CCL1128I Generating text summary trace
file *filename*

Explanation: Information message

CCL1129I Reading *num* files in the sequence
filename, '*wrapname <n>*' ...

Explanation: Information message

CCL1160E Error: DaemonAbend driven for
Signal==%d1

Explanation: The system has thrown a signal that has been caught by the Client daemon and has entered the DaemonAbend function for processing.

System action: The message is written to the error log. The signal code is logged.

User response: Investigate what caused the system to generate the signal. If the problem persists, contact your service organization.

CCL1161E Error: DaemonExit driven for
Signal==%d1

Explanation: The system has thrown a signal that has been caught by the Client daemon and has entered the DaemonExit function for processing.

System action: The message is written to the error log. The signal code is logged.

User response: Investigate what caused the system to generate the signal. If the problem persists, contact your service organization.

CCL1164E Error: *message* (Error Code = *error*)

Explanation: An internal product error occurred.

System action: The message is written to the error log.

User response: If the problem persists, contact your service organization.

CCL1195E Unable to open binary trace file
filename

Explanation: The Client cannot open the binary trace file for writing.

System action: The Client continues, but tracing is disabled.

User response: Check that the directory into which the binary trace file is being written is not write protected.

CCL1197E Unable to write to binary trace file (Error Code = Error Code)

Explanation: The binary trace file opened successfully, but an error occurred while a trace point was being written.

System action: The Client continues, but tracing is disabled.

User response: Check that the permissions of the trace file have not changed.

CCL1198E Out of memory while running with trace on

Explanation: There is not enough available memory for tracing to continue.

System action: The Client continues, but tracing is disabled.

User response: Make more memory available to the Client and then try again.

CCL1199E No memory buffer - unable to perform tracing operations

Explanation: An internal error has occurred. The Client is unable to buffer its writes to the trace file.

System action: The Client continues, but tracing is disabled.

User response: If the problem persists, contact your service organization.

CCL1205E Invalid system return code (Function = Function, Error Code = Error)

Explanation: An internal system function has returned an error while tracing was turned on.

System action: The Client continues, but tracing is disabled.

User response: If the problem persists, contact your service organization.

CCL1206E Internal error (Function = Function, Error Code = Error)

Explanation: An internal function has returned an error whilst tracing was turned on.

System action: The Client continues, but tracing is disabled.

User response: If the problem persists, contact your service organization.

CCL1207E Security failure - the owner of the binary trace file has changed

Explanation: This message will appear in the log file on UNIX and Linux if the owner of the trace file changes. For security reasons, you may not change the owner of the trace file whilst trace is running.

System action: The Client continues, but tracing is disabled.

User response: Restart trace and make sure that the owner of the trace file does not change.

CCL1208E Trace initialization failure

Explanation: An internal error prevented tracing from being turned on.

System action: The Client continues, but tracing is disabled.

User response: If the problem persists, contact your service organization.

CCL1209E Trace file deleted whilst tracing active

Explanation: The trace file was deleted whilst tracing was active.

System action: This message is written to the error log. Tracing continues.

User response: Deleting the trace file while

tracing is active is not recommended. To limit the size of the trace file, set the maximum Client wrap size. See the Administration guide for more details.

CCL1210W Trace component *component* is invalid and will be ignored

Explanation: The specified trace component is not valid.

System action: Tracing continues, but the specified component is ignored.

User response: Remove the reference to the invalid component.

CCL1211I Start wrapping trace request (total wrapping size = *INSERT-0*, size of each wrap file = *INSERT-1*, number of files = *INSERT-2*)

Explanation: Information message

CCL1212E Error acquiring trace lock (Function = *Function*, Error Code = *Error*), halting trace

Explanation: A previously caught system signal has caused the trace lock semaphore to become unobtainable.

System action: Tracing is terminated to prevent the possibility of further error messages.

User response: Investigate what caused the system to generate the initial signal.

CCL1213E Unable to activate memory mapped trace, because wrapping trace is not enabled

Explanation: The CICSCLI -B option has been used to specify memory mapping trace, but MAXWRAPSIZE is 0, or is not set in the configuration file ctg.ini. Memory mapped trace can be used only if trace wrapping is enabled.

System action: Tracing will be written using standard file I/O instead of memory mapped trace.

User response: Specify a maximum Client wrap

size, restart the Client and invoke memory mapped trace again.

CCL1214W A thread ended while writing a trace point (pid = *pid*, tid = *tid*, trace pt = *trace pt*, buffer size = *size*)

Explanation: This message is output when running with trace on. It indicates a thread ended while holding the trace lock, possibly leaving a partially written trace point in the internal trace buffer. Before the thread ended, it recorded the process id *pid*, thread id *tid* and trace point number *trace pt* in the trace buffer. This message is output during normal process termination and may be output if a thread terminates unexpectedly.

System action: This message is logged, and Client trace continues. The specified trace point may not appear in the trace.

User response: Ignore this warning if the message is logged during termination of the specified process. Otherwise if the warning persists, contact your service organization.

CCL1215E A thread ended unexpectedly while writing a trace point (buffer size = *size*)

Explanation: A thread ended unexpectedly, leaving a partially written trace point in the internal trace buffer. The buffer contains too little information to determine which trace point was being written when the thread ended.

System action: This message is logged, and Client trace continues. The individual trace point being written is lost.

User response: If the problem persists, contact your service organization.

CCL1216E Unable to activate memory mapped trace, due to insufficient space on file system

Explanation: The MAXWRAPSIZE parameter in the configuration file ctg.ini controls the maximum wrapping size of memory mapped

trace. This is set to a value which is larger than the available space in the filesystem that trace is written to.

System action: This message is logged. Client tracing is not activated.

User response: Either increase the file system so that it can contain the full trace file, or reduce the value of MAXWRAPSIZE.

CCL1501E Unexpected return from WaitForMultipleObjects (Error = error number)

Explanation: The Client service process received an unexpected return code from the Windows API function WaitForMultipleObjects.

System action: This message is written to the error log file. The service will terminate.

User response: Inspect the Windows application event log, other messages may indicate the cause of failure. If the problem persists, contact your service organization.

CCL1521E Exception *exception* raised in class *class* *::method* during 'call' request

Explanation: The exception named, one of the Ccl::ExCode enumeration values, was raised during the execution of the *class ::method* method during the server request indicated by the 'call' value. An exception object encapsulating details of the exception is passed to CclECL::handleException or CclEPI::handleException and/or thrown via the C++ exception mechanism.

System action: The message is written to the trace file if tracing is enabled.

User response: None.

CCL1540E Unable to change to user *user*

Explanation: ctgd was unable to change the effective user ID of the running process to the specified user. This is due to either the current user having insufficient privilege, or the specified user ID being invalid.

System action: The Gateway daemon does not start.

User response: Check that the current user has sufficient privilege to change user ID and that the user ID specified in ctgd.conf is valid.

CCL1541E Unable to change to group *group*

Explanation: ctgd was unable to change the effective group of the running process to the specified group. This is due to either the current user having insufficient privilege, or the specified group being invalid.

System action: The Gateway daemon does not start.

User response: Check that the current user has sufficient privilege to change group and that the group specified in ctgd.conf is valid.

CCL1542E Unable to start *program* (errno = error)

Explanation: ctgd was unable to start the specified program. The specific error code can be referenced in the documentation for your operating system.

System action: The program in question, typically the Gateway daemon, does not start.

User response: Ensure that the path is absolute and the user specified in ctgd.conf has permission to execute it.

CCL2001E Server *server* is undefined

Explanation: The Client daemon received a request for a server, but the server is not defined in the configuration file.

System action: The request fails but the Client daemon continues.

User response: Ensure the server name specified by the ECI or EPI application, or by the 3270 Emulator is correct. If required, add the server name to the configuration file, then stop and restart the product to activate the changes.

CCL2003I **Server *server* requires a security identification**

Explanation: Information message

CCL2004E **The Client daemon cannot continue**

Explanation: An error was detected.

System action: The Client daemon terminates.

User response: Examine any other messages and the Client error log to determine the cause of the error.

CCL2005I **The Client daemon will continue**

Explanation: Information message

CCL2006I **Refer to the Client error log for more details**

Explanation: Information message

CCL2007E **Free memory is exhausted**

Explanation: The Client daemon ran out of memory while processing a request.

System action: The request fails and the Client daemon attempts to continue processing.

User response: Try altering the values in the configuration file to provide a larger free memory pool or to reduce the overall memory requirements.

CCL2008E **An internal state error occurred**

Explanation: The Client daemon detected an unexpected internal error.

System action: Error information is written to the Client error log.

User response: If the problem persists, contact your service organization.

CCL2009E **An unexpected error has occurred**

Explanation: The Client daemon detected an unexpected error.

System action: Error information is written to the Client error log.

User response: If the problem persists, contact your service organization.

CCL2010E **Internal transport error (Function = *Function*, Error Code = *Error*)**

Explanation: An internal Client function failed.

System action: The message is written to the error log and the Client terminates. The function name and error code are logged.

User response: If the problem persists, contact your service organization.

CCL2011E **Out of memory creating session control structure**

Explanation: The Client was unable to allocate enough memory to create an internal control structure for a new server conversation.

System action: The message is written to the error log and processing continues. The conversation is not created.

User response: Try altering the values in the configuration file *ctg.ini* to provide the Client with a larger free memory pool or to reduce the overall memory requirements.

CCL2012E **Unable to find session entry (*session id*, *slot number*)**

Explanation: The Client was unable to locate an internal control structure for an existing server conversation.

System action: The message is written to the error log and processing continues. The conversation is ended.

User response: If the problem persists, contact your service organization.

CCL2013E **Out of memory creating terminal control structure**

Explanation: The Client was unable to allocate enough memory to create an internal control structure associated with an emulator or EPI terminal.

System action: The message is written to the error log and processing continues. The control structure is not created.

User response: Try altering the values in the configuration file `ctg.ini` to provide the Client with a larger free memory pool or to reduce the overall memory requirements.

CCL2014E Unable to find terminal entry
(*session id, terminal index, terminal id*
)

Explanation: The Client was unable to locate an internal control structure associated with a terminal or EPI program. This may occur if an ATI request is received for a terminal that no longer exists because either the Client or the terminal was shut while the ATI was in transit.

System action: The message is written to the error log and Client processing continues but the emulator may be unable to continue.

User response: If the problem persists, contact your service organization.

CCL2015E Unable to find server entry (*link id*
)

Explanation: The Client received an event associated with an unconnected server.

System action: The message is written to the error log and processing continues. The event is ignored.

User response: If the problem persists, contact your service organization.

CCL2016E Invalid communications event identifier (*event id*)

Explanation: The Client received an unknown event from one of the communications protocol drivers.

System action: The message is written to the

error log and processing continues. The event is ignored.

User response: If the problem persists, contact your service organization.

CCL2017W Default userid has been set to spaces

Explanation: The default userid has been set to a value which contains only spaces.

System action: The message is logged, and processing continues.

User response: When using connection security, blank userids and passwords are valid only if 'Usedfltuser' is set to YES in the CICS connection definition.

CCL2018E Internal function error (Function =
Function, Error Code = Error)

Explanation: An internal function failed.

System action: The message is written to the error log and processing continues. The function name and error code are logged.

User response: If the problem persists, contact your service organization.

CCL2024E Unknown application
(*SessId=session*)

Explanation: The Client tried to send a response to an unknown application. Often this occurs if the application has ended before its response was received. The *session* value identifies the application.

System action: The message is written to the trace file if tracing is enabled.

User response: None.

CCL2044E Out of memory creating Client daemon control structure

Explanation: The Client was unable to allocate enough memory to create an internal control structure.

System action: The message is written to the error log and processing continues if possible.

User response: Try altering the values in the configuration file `ctg.ini` to provide the Client with a larger free memory pool or to reduce the overall memory requirements.

CCL2051E CICS system transaction *tran* failed on server *server*

Explanation: The specified Client system transaction *tran* failed to run on the specified server. If connecting to a CICS TS for iSeries server, a failure of transaction CCIN indicates that the server is not available.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL2052E Cannot open error log file *file name*

Explanation: The Client daemon could not open or write to the specified error log file.

System action: Log messages are lost.

User response: Check that the name specified in the `LogFile` parameter in the configuration file is correct and accessible.

CCL2053I Cannot open trace file *file name* - tracing has been disabled

Explanation: Information message

CCL2054I A CICS server error has occurred

Explanation: Information message

CCL2070E 'MaxBufferSize' limit of *size* exceeded by incoming data length *length*

Explanation: The Client received data from a server that exceeded the `MaxBufferSize` limit. The *size* value indicates the current `MaxBufferSize`

setting and the *length* value indicates the length of the incoming data.

System action: The message is written to the error log and processing continues.

User response: Try altering the value of `MaxBufferSize` in the configuration file `ctg.ini` to at least the size of the incoming data.

CCL2079E Failed to put transport message, retry number *number*

Explanation: The Client has attempted to put a transport message to a window owned by the Client application. This has failed, the most likely reason for this is that the message queue associated with the windows Client application is full.

System action: The message is written to the trace file if tracing is enabled. The Client allows windows to process messages in the queue and then tries posting the transport message again, a number of times.

User response: Try using `SetMessageQueue` to increase the size of the message queue.

CCL2087I Server *server*, User ID invalid, please re-enter

Explanation: Information message

CCL2088I Server *server*, Password invalid, please re-enter

Explanation: Information message

CCL2089I Server *server*, Password expired, please change

Explanation: Information message

CCL2090I Server *server*, New Password unacceptable, please re-enter

Explanation: Information message

CCL2091I New and Confirm Passwords do not match, please re-enter

Explanation: Information message

CCL2094E **Unable to access print output file for terminal** *netname*

Explanation: The Client daemon failed to open the print output file for the specified terminal.

System action: The terminal screen data is not printed to file.

User response: Ensure that the file system permissions for the specified file are sufficient for the Client daemon to write to the file and that no other processes have the file open. If the problem persists, contact your service organization.

CCL2153I **Userid or password is too long**

Explanation: Information message

CCL2162W **The Client daemon process has become unresponsive**

Explanation: An internal element of work took more than 30s to complete.

System action: This message is output.

User response: Check if any requests have failed to respond. If this problem continues contact your service organization.

CCL2163I **The Client daemon is configured to gather problem diagnosis information if it becomes unresponsive**

Explanation: Information message

CCL2164I **The Client daemon is configured to gather problem diagnosis information when CCL** *CCL message number* **is written to the log**

Explanation: Information message

CCL2165I **The CTGDIAG environment variable is set to** *CTGDIAG*

Explanation: Information message

CCL2218I **The initialization of the Client daemon is complete**

Explanation: Information message

CCL2219I **A normal shutdown of the Client daemon has started**

Explanation: Information message

CCL2220I **An immediate shutdown of the Client daemon has started**

Explanation: Information message

CCL2221I **The Client daemon has shut down**

Explanation: Information message

CCL2222I **About to connect to server** *server*

Explanation: Information message

CCL2223I **Connected to server** *server* **using protocol to** *netname*

Explanation: Information message

CCL2224I **An immediate disconnection from server** *server* **started**

Explanation: Information message

CCL2225I **A normal disconnection from server** *server* **started**

Explanation: Information message

CCL2226I **Disconnection from server** *server* **using protocol to** *netname* **complete**

Explanation: Information message

CCL2227I **Service trace has started**

Explanation: Information message

CCL2228I Memory mapped trace has started

Explanation: Information message

CCL2229I Client components to trace have been set to *trace components*

Explanation: Information message

CCL2230I Service trace has ended

Explanation: Information message

CCL2231I Terminal on server *server* with TermID *termID* and NetName *netname* has been installed

Explanation: Information message

CCL2232I A request to delete terminal with TermID *termID* and NetName *netname* has been sent to server *server*

Explanation: Information message

CCL2233I A request to purge terminal with TermID *termID* and NetName *netname* has been sent to server *server*

Explanation: Information message

CCL2234W CCIN Timeout occurred (Server=*server name*, LinkId=*link id*)

Explanation: An attempt has been made to connect to a CICS server but no response to the CCIN request has been received by the Client daemon.

System action: The connection to the CICS server is closed. The server will be removed from the list of servers and CICS TG will attempt to reconnect only if a request for that server is received.

User response: Check the CICS server to determine why it did not respond to the CCIN

request. If the problem persists, contact your service organization.

CCL2236I Security settings (user ID and password) have been changed for server *server*

Explanation: Information message

CCL2237E The client information log cannot be opened for writing

Explanation: The configuration file contains an entry for a separate information log, but the file specified could not be opened for writing.

System action: Informational messages will be output to the error log instead.

User response: Correct the entry in the configuration file to avoid seeing this message in future.

CCL2238I Server *server* is configured to use the SNA protocol with NETNAME *NETNAME* and LOCALLUNAME *LUNAME*

Explanation: Information message

CCL2239I Server *server* is configured to use the TCP/IP protocol with NETNAME *NETNAME* and PORT *LUNAME*

Explanation: Information message

CCL3001W Errors detected at or before line *number* in file *file name* with details in the following message

Explanation: Errors were detected at or before the specified line in the configuration file. The following message will highlight the error and its severity.

System action: The Client daemon might terminate.

User response: See following message for details and correct the configuration file and restart the product.

CCL3002E **Cannot find configuration file** *file name*

Explanation: The specified configuration file could not be found.

System action: The Client daemon terminates.

User response: Ensure that the required configuration file exists and is correctly specified.

CCL3003E **Missing = sign after parameter** *parameter*

Explanation: An = sign was expected after a parameter in the configuration file.

System action: The Client daemon terminates.

User response: Correct the configuration file error and restart the product.

CCL3004E **Parameter** *text* **not recognized**

Explanation: An unknown parameter starting with specified text was encountered in the configuration file.

System action: The Client daemon terminates.

User response: Correct the configuration file error and restart the product.

CCL3005E **Parameter** *parameter* **repeated or misplaced**

Explanation: A valid parameter was encountered in the wrong position in the configuration file.

System action: The Client daemon terminates.

User response: Correct the configuration file error and restart the product.

CCL3006E **Parameter** *parameter* **has invalid value** *value*

Explanation: A valid parameter had an incorrect value specified in the configuration file.

System action: The Client daemon terminates.

User response: Correct the configuration file

error and restart the product.

CCL3007E **Inconsistent values for parameters** *parameter 1 and parameter 2*

Explanation: The values of certain parameters specified in the configuration file have dependencies on each other. The values for the two specified parameters are inconsistent.

System action: The Client daemon terminates.

User response: Correct the configuration file error and restart the product.

CCL3008E **Required parameter** *parameter* **missing**

Explanation: The specified parameter is missing from the configuration file and must be provided.

System action: The Client daemon terminates.

User response: Correct the configuration file error and restart the product.

CCL3009E **Not enough memory for Client daemon initialization**

Explanation: There was not enough free memory to process the configuration file.

System action: The Client daemon terminates.

User response: Try altering the values in the configuration file to provide a larger free memory pool or to reduce the overall memory requirements.

CCL3010E **Missing driver definition for protocol** *protocol*

Explanation: The value of the protocol parameter in a Server section of the configuration file must match with the name of a Driver section. No matching driver definition was found for the named protocol.

System action: The Client daemon terminates.

User response: Correct the configuration file error and restart the product.

CCL3011E **Errors detected while reading file**
file name

Explanation: Errors were encountered while reading the configuration file.

System action: The Client daemon terminates.

User response: Correct the configuration file error and restart the product.

CCL3012E **Duplicate section section**
definition

Explanation: A Client, Server, or Driver section in the configuration file has the same name as a previously specified section.

System action: The Client daemon terminates.

User response: Correct the configuration file error and restart the product.

CCL3013E **Unable to access file** *file name,*
rc=rc

Explanation: Errors were encountered while accessing file *file name* .

System action: The Client daemon terminates.

User response: Change the file permissions and restart the product.

CCL3014W **MAXREQUESTS parameter value**
is too high; the maximum
permitted value *max* **has been**
used

Explanation: The MAXREQUESTS parameter in the configuration file has been set to a value higher than the maximum allowed.

System action: The Client daemon sets MAXREQUESTS to the maximum allowed (*max*) and continues.

User response: To avoid this message being logged at startup, correct the configuration file. The CICS Transaction Gateway Desktop Edition is intended for single user use only; CICS Transaction Gateway for Multiplatforms can be configured to handle a higher number of concurrent requests.

CCL3015W **Both 'CTG.INI' and 'ctg.ini'**
configuration files exist, using
default file 'ctg.ini'

Explanation: The default directory contains both uppercase and lowercase versions of the configuration file.

System action: The Gateway daemon will use the default file 'ctg.ini' as no configuration filename has been specified. The contents of file 'CTG.INI' will be ignored.

User response: To avoid this warning being issued, explicitly specify a configuration file or delete file 'CTG.INI' from the default directory. For details of how to specify a configuration file see the Configuration chapter of the Administration Guide.

CCL3016E **The use of Windows credentials**
for security is not supported

Explanation: The USENPI option exists in a server section of the configuration file, USENPI is no longer supported.

System action: The Client daemon terminates.

User response: Edit the configuration file, remove all USENPI statements.

CCL3017E **The TCP62 protocol is not**
supported

Explanation: Parameters in either one or both of the server and driver sections of the configuration file have been configured to use TCP62. This product does not support TCP62.

System action: This message is logged and the Client daemon fails to start.

User response: The CICS Transaction Gateway Information Center provides details and examples of how to configure the Client daemon to use a range of supported network protocols to connect to CICS.

CCL3018W **SECTION CLIENT parameter**
"Application ID" is deprecated

Explanation: The "Application ID" parameter

specified on SECTION CLIENT is deprecated, it is replaced by a new APPLID parameter.

System action: The Application ID specified on SECTION CLIENT will be used when communicating with CICS servers.

User response: Refer to the product information center for details of the new APPLID parameter.

CCL3019W SECTION CLIENT parameter "Application ID" is overridden by SECTION PRODUCT parameter "APPLID"

Explanation: The "Application ID" parameter specified on SECTION CLIENT is deprecated and is overridden by the new APPLID parameter.

System action: The APPLID value specified in SECTION PRODUCT will be used when communicating with CICS servers.

User response: Set the APPLID parameter to the value required and set "SECTION CLIENT = *".

CCL3020W SECTION SERVER = *Server Name* parameter "LUALIASNAMES" is deprecated

Explanation: The LUALIASNAMES parameter specified in the SECTION SERVER is deprecated. On a Windows platform LUALIASNAMES is replaced by parameters LOCALLUALIAS and PARTNERLUALIAS, on a UNIX or Linux platform LUALIASNAMES is replaced by PARTNERLUALIAS.

System action: On a Windows platform the value of parameters LOCALLUALIAS and PARTNERLUALIAS is set to the same value specified for LUALIASNAMES, on a UNIX or Linux platform PARTNERLUALIAS defaults to Y.

User response: Refer to the product information center for details of the parameters LOCALLUALIAS and PARTNERLUALIAS.

CCL3021W SECTION SERVER = *Server Name* parameter "LUALIASNAMES" is overridden by SECTION SERVER parameters "LOCALLUALIAS" and, or "PARTNERLUALIAS"

Explanation: The LUALIASNAMES parameter specified in the SECTION SERVER is deprecated and is overridden. On a Windows platform LUALIASNAMES is replaced by parameters LOCALLUALIAS and PARTNERLUALIAS. On a UNIX or Linux platform LUALIASNAMES is replaced by PARTNERLUALIAS.

System action: When LOCALLUALIAS and, or PARTNERLUALIAS is specified the value of LUALIASNAMES is ignored.

User response: Set the parameters LOCALLUALIAS and PARTNERLUALIAS to the required values.

CCL3022W The name of *section section name* contains characters that are not supported

Explanation: The specified section has a name containing characters that are not in the set of supported characters for server definitions. Supported characters are A-Z and 0-9 and '@', '#', '\$', '-'.

System action: The Client daemon continues.

User response: Edit the name of the specified section in the configuration file so that it contains only supported characters.

CCL3102E Inbound GDS data error (*gds, length, size*)

Explanation: The Client received invalid data from a server. The data does not have a valid CICS data stream header. The *gds* value should be either 0x12F2 or 0x12FF, the *length* value gives the expected data length, and *size* gives the true received length.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL3103E Inbound SYNC/ROLL data error
(length, data)

Explanation: The Client received invalid data from a server. The data was not part of a valid SYNC or ROLLBACK flow. The *length* values gives the length of the received data and *data* contains the first four bytes.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL3104E Inbound FMH43 header error
(length)

Explanation: The Client received invalid data from a server. The data was not a valid CICS FMH43 header. The *length* values gives the length of the received data.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL3105E Inbound CICS datastream error
(type, value1, value2)

Explanation: The Client received invalid data from a server. The data was not a valid CICS data stream. The *type* value indicates which part of the data contained errors, *value1* and *value2* further identify the error. This message may be received if the server does not support sign-on capable terminals.

System action: The message is written to the error log and processing continues.

User response: Use the option to request a non-sign-on capable terminal. If the problem persists, contact your service organization.

CCL3106E Out of memory unpacking CICS datastream

Explanation: The Client was unable to allocate enough memory to save data from a server. Normally such data is a COMMAREA associated

with an emulator terminal.

System action: The message is written to the error log and processing continues.

User response: Try altering the values in the configuration file *ctg.ini*, to provide the Client with a larger free memory pool or to reduce the overall memory requirements.

CCL3107E Inbound FMH5 transaction name incorrect (data)

Explanation: The Client received an invalid transaction attach request from a server. The only valid transaction name is CRSR, used to start ATI requests to a terminal. The *data* value indicates the requested transaction name.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL3108E Inbound FMH5 header error
(length)

Explanation: The Client received invalid data from a server. The data was not a valid CICS FHM5 header. The *length* value gives the length of the received data.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL3109E Cannot connect to server *server* - client is already installed

Explanation: The client name specified in the Client section of the configuration file was already installed on the specified server.

System action: The server rejects the Client daemon's installation request.

User response: Ensure all Client names are unique within the network or specify the name as "*" to enable a suitable unique name to be generated automatically.

CCL3110E Cannot connect to server *server* - server is busy

Explanation: The specified server was busy and could not process the Client daemon installation request.

System action: The server rejects the Client daemon's installation request.

User response: Retry the request sometime later when the server is not so busy. If the problem persists contact your server administrator.

CCL3111E Cannot connect to server *server* - server rejected install request

Explanation: The specified server could not process the Client installation request.

System action: The server rejects the Client daemon's installation request, server continues processing.

User response: Examine the Client error log and, if possible, trace information on both the Client and server to determine the cause of the error.

CCL3112E Cannot connect to server *server* - request rejected by server exit

Explanation: A user exit running on the specified server rejected the request to install the Client definition.

System action: The server rejects the Client's installation request, the Client daemon continues processing.

User response: Check with the server system administrator to determine why the Client installation request was rejected.

**CCL3120E PEM response data error:
ReqType=*Request Type*,
ErrorValue=*value***

Explanation: The Client received the password expiry management (PEM) transaction response from a server. The data contained in the corresponding PEM request contained a data formatting error.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL3121E Cannot connect to server *server* - invalid codepage specified

Explanation: The codepage value sent to the server was rejected.

System action: The server rejects the Client's installation request, the Client daemon continues processing.

User response: If the CCSID parameter has been specified in ctg.ini, check that the value specified is valid. Otherwise, check with the server system administrator to determine why the codepage value was rejected.

CCL3123E Terminal install failed because server '*server*' does not support terminals over TCP/IP

Explanation: CICS TS does not support EPI or cicsterm when using the TCP/IP protocol. The *server* value defines the server that the terminal install failed with.

System action: The server rejects the terminal installation request. The Client daemon logs this error message. The connection between the Client daemon and server remains active.

User response: Configure the Client daemon to use the SNA protocol to connect to CICS TS.

CCL3209E Link to server (*link*) is no longer available

Explanation: The Client tried to communicate with a server whose connection is currently unavailable.

System action: The message is written to the error log and processing continues. The server is ignored.

User response: If the problem persists, contact your service organization.

CCL3221E Unable to find link control structure (*link*)

Explanation: The Client was unable to locate an internal control structure associated with a connection to a server.

System action: The message is written to the error log and processing continues. The server is ignored.

User response: If the problem persists, contact your service organization.

CCL3222E Unable to find conversation control structure (*conv*)

Explanation: The Client was unable to locate an internal control structure associated with a server conversation.

System action: The message is written to the error log and processing continues. The conversation is ignored.

User response: If the problem persists, contact your service organization.

CCL3225E Invalid communications event identifier (*event*)

Explanation: The Client received an invalid internal event from a communications protocol driver.

System action: The message is written to the error log and processing continues. The event is ignored.

User response: If the problem persists, contact your service organization.

CCL3227E Conversation (*conv*) is not in state

Explanation: The Client detected a conversation to a server in an incorrect state.

System action: The message is written to the error log and processing continues. The request is ignored.

User response: If the problem persists, contact your service organization.

CCL3228E Unable to find protocol control structure (*name, driver*)

Explanation: The Client was unable to locate an internal control structure associated with a communications protocol driver.

System action: The message is written to the error log and processing continues. The driver is ignored.

User response: If the problem persists, contact your service organization.

CCL3229E Cannot load protocol driver *driver*

Explanation: The specified communications protocol driver module could not be found or loaded.

System action: No communication can take place using the protocol, the Client daemon continues processing.

User response: Examine the Client error log and, if possible, trace information to determine the cause of the error.

CCL3230E Out of memory creating communications control structure

Explanation: The Client was unable to allocate enough memory to create an internal control structure associated with communications.

System action: The message is written to the error log and processing continues.

User response: Try altering the values in the configuration file *ctg.ini*, to provide the Client with a larger free memory pool or to reduce the overall memory requirements.

CCL3247E Error loading DLL *fullname* (Function = *Function*, Error Code = *Error*)

Explanation: The Client was unable to load a communications protocol driver DLL.

System action: The message is written to the error log and processing continues. The name of

the failing DLL and the loading function and error codes are logged.

User response: Ensure the DLL is available and correctly named. If it is the SNA protocol driver that cannot be loaded as it is unable to load the required libraries from the SNA product, set the environment variable LD_LIBRARY_PATH to include the location of those libraries. If the problem persists, contact your service organization.

CCL3260E EPI application cannot communicate with the Client daemon

Explanation: The Client application cannot communicate with the Client daemon.

System action: This message is written to the log file. The EPI request returns CICS_EPI_ERR_FAILED.

User response: This is expected behavior if the Client daemon has been stopped or restarted while an EPI session is active. First shut down the Gateway daemon if it is running, then shut down the Client daemon, and finally restart the Gateway daemon. If you are running a local mode Java application, shut it down and then restart it. Code any user applications that receive this error to restart the EPI by issuing a CICS_EpiTerminate call, followed by CICS_EpiInitialize to reinitialize the EPI; subsequent requests should complete successfully. Alternatively program applications to end if they receive this error. If the Client daemon had not been stopped or restarted, and the problem persists, contact your service organization.

CCL3261E ECI application cannot communicate with the Client daemon

Explanation: The Client application cannot communicate with the Client daemon.

System action: This message is written to the log file. The ECI request returns ECI_ERR_SYSTEM_ERROR.

User response: This is expected behavior if the Client daemon has been stopped while an application, or the Gateway daemon is active. First shut down the Gateway daemon if it is running, then shut down the Client daemon, and finally restart the Gateway daemon. If you are running a local mode Java application shut it down and then restart it.

CCL3280E Errors occurred while initializing protocol driver *driver*, driver return code *return code*

Explanation: The specified communications protocol driver module reported errors during startup.

System action: No communications can take place using the protocol, the Client daemon continues processing.

User response: Examine the Client error log and, if possible, trace information to determine the cause of the error.

CCL3281E Errors occurred while terminating protocol driver *driver*, driver return code *return code*

Explanation: The specified communications protocol driver module reported errors while it was closing down.

System action: The error is ignored and the Client daemon continues processing.

User response: Examine the Client error log and, if possible, trace information to determine the cause of the error.

CCL3282E Errors occurred while connecting to a server using protocol driver *driver*, driver return code *return code*

Explanation: The specified communications protocol driver module reported errors while it was trying to connect to a server.

System action: The Client daemon cannot communicate with the server, the Client daemon continues processing.

User response: Examine the Client error log and, if possible, trace information to determine the cause of the error.

CCL3283E **Errors occurred while disconnecting from a server using protocol driver *driver*, driver return code *return code***

Explanation: The specified communications protocol driver module reported errors while it was closing a connection with a server.

System action: The error is ignored and Client daemon continues processing.

User response: Examine the Client error log and, if possible, trace information to determine the cause of the error.

CCL3284E **Errors occurred while starting a conversation with a server using protocol driver *driver*, driver return code *return code***

Explanation: The specified communications protocol driver module reported errors while it was starting a new conversation with a connected server.

System action: The request to start the conversation fails and the Client daemon continues processing.

User response: Examine the Client error log and, if possible, trace information to determine the cause of the error.

CCL3285E **Errors occurred while closing a conversation with a server using protocol driver *driver*, driver return code *return code***

Explanation: The specified communications protocol driver module reported errors while it was closing a conversation with a connected server.

System action: The request to close the conversation fails and the Client daemon continues processing.

User response: Examine the Client error log

and, if possible, trace information to determine the cause of the error.

CCL3286E **Errors occurred while sending data to a server using protocol driver *driver*, driver return code *return code***

Explanation: The specified communications protocol driver module reported errors while it was sending data over a conversation with a connected server.

System action: The request that sent the data fails and the Client daemon continues processing.

User response: Examine the Client error log and, if possible, trace information to determine the cause of the error.

CCL3287E **Errors occurred while receiving data from a server using protocol driver *driver*, driver return code *return code***

Explanation: The specified communications protocol driver module reported errors while it was receiving data from a conversation with a connected server.

System action: The request that required the data fails and the Client daemon continues processing.

User response: Examine the Client error log and, if possible, trace information to determine the cause of the error.

CCL3288E **Errors occurred while communicating with a server using protocol driver *driver*, driver return code *return code***

Explanation: The specified communications protocol driver module reported errors while it was communicating or trying to communicate with a server.

System action: The request that initiated the communications fails and the Client daemon continues processing.

User response: Examine the Client error log

and, if possible, trace information to determine the cause of the error.

CCL3292E Error loading DLL *fullname*
(Function = *Function*, OS Error
Data = *Error*)

Explanation: The Client was unable to load a communications protocol driver DLL.

System action: The message is written to the error log and processing continues. The name of the failing DLL, the loading function and error data are logged.

User response: Ensure the DLL is available and correctly named. If the problem persists, contact your service organization.

CCL3299E Internal communications error
(Function = *Function*, Error Code =
***Error*)**

Explanation: An internal Client communications function failed.

System action: The message is written to the error log and the Client terminates. The function name and error code are logged.

User response: If the problem persists, contact your service organization.

CCL4401E TCP/IP not available

Explanation: The TCP/IP protocol driver cannot issue TCP/IP calls on this system. TCP/IP may not be installed or may not be active on the system.

System action: The message is written to the error log and processing continues.

User response: Alter the system configuration to try to activate or install TCP/IP. If the problem persists, contact your service personnel.

CCL4402E TCP/IP out of memory for data buffers

Explanation: The TCP/IP protocol driver could not allocate enough memory to create the internal data buffers.

System action: The message is written to the error log and processing continues.

User response: Try altering the values in the configuration file *ctg.ini*, to provide the Client with a larger free memory pool or to reduce the overall memory requirements.

CCL4403E TCP/IP (to *server*) no resources available: ErrNo=*error* (*label*)

Explanation: The TCP/IP protocol driver could not create a TCP/IP socket. The *server* value identifies the server. The *error* value identifies the TCP/IP error code. The *label* value provides a description of the error.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL4404E TCP/IP unable to resolve IP name '*netname* ' for server '*server* ':
ErrNo=*error* (*label*)

Explanation: The TCP/IP protocol driver could not resolve a server host IP name. The *netname* value identifies the IP name and *server* identifies the server. The *error* value gives the TCP/IP error code. The *label* value provides a description of the error.

System action: The message is written to the error log and processing continues.

User response: Ensure the server IP name or address is correct and can be resolved by the local name server. Use the TCP/IP ping or nslookup commands to see if TCP/IP can resolve the hostname. If the problem persists contact your service organization.

CCL4405E TCP/IP (to *server*) out of memory for control blocks

Explanation: The TCP/IP protocol driver could not allocate enough memory to create the internal control structures; *server* identifies the server.

System action: The message is written to the

error log and processing continues.

User response: Try altering the values in the configuration file `ctg.ini`, to provide the Client with a larger free memory pool or to reduce the overall memory requirements.

CCL4406E TCP/IP (to *server*) **send() API failed, ErrNo = error (label)**

Explanation: The TCP/IP protocol driver could not send data to a server. The *server* value identifies the server. The *error* value gives the TCP/IP error code. The *label* value shows the description of the error.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL4407E TCP/IP (to *server*) **connect() API failed, ErrNo = error (label)**

Explanation: The TCP/IP protocol driver could not connect to a server. The *server* value identifies the server. The *error* value gives the TCP/IP error code. The *label* value shows the description of the error.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL4408E TCP/IP (to *server*) **recv() API failed, ErrNo = error (label)**

Explanation: The TCP/IP protocol driver could not receive data from a server. The *server* value identifies the server. The *error* value gives the TCP/IP error code. The *label* value shows the description of the error.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL4409E TCP/IP (to *server*) **error starting listener**

Explanation: The TCP/IP protocol driver could not start an internal listener function; *server* identifies the server.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL4410E TCP/IP requires WINSOCK **required (found installed)**

Explanation: The TCP/IP protocol driver for Windows requires a WINSOCK DLL of at least version *required*, however version *found* was found to be installed. The *required* and *found* values indicate the version of WINSOCK required and found respectively.

System action: The message is written to the error log and processing continues.

User response: Ensure your TCP/IP product provides a suitable level of the WINSOCK DLL and the system is correctly configured. If the problem persists contact your service organization.

CCL4413I TCP/IP (to *server*) **address=*ipaddr*, port=*port*, socket=*socket***

Explanation: Information message

CCL4414E TCP/IP (to *server*) **link failed: RC=*error***

Explanation: The TCP/IP link failed. The *server* value identifies the server and the *error* value gives the TCP/IP error code.

System action: The message is written to the trace file if tracing is enabled.

User response: If the problem persists, contact your service organization.

CCL4415W TCP/IP (to *server*) **abend received: SenseCode=*sense code***

Explanation: The transaction failed. The *server* value identifies the server and the *sense code* value gives the FMH7 sense data.

System action: The message is written to the trace file if tracing is enabled.

User response: If the problem persists, contact your service organization.

CCL4425E TCP/IP Allocation for memory block failed rc = rc

Explanation: During TCP/IP processing a memory allocation failed.

System action: The message is written to the trace file if tracing is enabled.

User response: None.

CCL4426E TCP/IP (to *server*) Error when attempting to send good ping response, rc= rc

Explanation: An error occurred when attempting to send a good ping response.

System action: The message is written to the trace file if tracing is enabled.

User response: None.

CCL4427E TCP/IP (to *server*) Received unknown ping

Explanation: Received a flow that looks like a ping but cannot be interpreted by the existing criteria.

System action: The message is written to the trace file if tracing is enabled.

User response: None.

CCL4429E TCP/IP connection ping reply (echo), from server *INSERT-0*, session id = *INSERT-1*

Explanation: This is the end of the processing loop. For the receive thread to be restarted, *INSERT-0* needs to be zero and *INSERT-1* needs to be zero or positive. If the rcv thread is not

started, tcp receives will not occur.

System action: The message is written to the trace file if tracing is enabled.

User response: None.

CCL4430E TCP/IP (to *server*) Bad send error number error

Explanation: The TCP/IP send got a bad return code.

System action: The message is written to the trace file if tracing is enabled.

User response: None.

CCL4436E TCP/IP (to *server*) Unable to allocate memory for conversation control block

Explanation: The protocol driver was unable to allocate memory when starting a conversation.

System action: The conversation will not be allocated, this message is logged and the Client daemon continues.

User response: Free up some memory and retry.

CCL4437E TCP/IP (to *server*) select failed, ErrNo = error (error description)

Explanation: The TCP/IP socket select call completed with an error. The *error* value gives the TCP/IP error code. The *error description* value shows the description of the error.

System action: The message is written to the error log and processing continues.

User response: Check the configuration for the given server *server*, specifically check the address of the server machine and the port that the CICS server is using. Check the basic connectivity to the server using the "ping" command to check the address of the server machine is accessible on the network. Check that the CICS server is up and listening on the expected port. If the problem persists, contact your service organization.

CCL4438E TCP/IP (to *server*) select timed out

Explanation: The TCP/IP socket select call timed out.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL4439E TCP/IP (to *server*) select failed because file descriptor not set

Explanation: A TCP/IP socket select call could not complete as the file descriptor had not been set.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL4440E TCP/IP (to *server*) Socket closed

Explanation: The TCP/IP socket being used has closed. This message can be logged during shutdown of a server connection if the CICS server closes the socket when it receives a CCIN uninstall request.

System action: The message is written to the error log and processing continues.

User response: Ignore this error if it occurs during shutdown of the Client daemon or a server connection. Otherwise, if the problem persists, contact your service organization.

CCL4441E TCP/IP (to *server*) Host unreachable

Explanation: The host specified is unreachable.

System action: The message is written to the trace file and processing continues

User response: If the problem persists, contact your service organization.

CCL4443E TCP/IP (to *server*) Repeatedly unable to send to CICS

Explanation: The Client daemon could not send to the CICS region. The send was retried until the retry count was exceeded. This error can be caused by a full local send buffer or a full remote receive buffer.

System action: The link to the server is marked as unavailable and will need to be reconnected before it can be used again.

User response: Re-establish the connection and try increasing the size of the TCP/IP send and receive buffers.

CCL4446E TCP/IP (to *server*) TCP/IP send() API call timed out (errno = *label*)

Explanation: The TCP/IP socket send call did not complete within 5 seconds. The *label* value gives the TCP/IP error code.

System action: The message is written to the error log and processing continues. The send call is expected to complete within 5 seconds and not doing so may indicate an underlying problem in the OS or network hardware.

User response: If the problem persists, contact your service organization.

CCL4450I TCP/IP send to *INSERT-0* timed out; connection closed

Explanation: Information message

CCL4499E TCP/IP (to *server*) data to be received overflows buffer

Explanation: The TCP/IP protocol driver received more data than its buffer space could hold. The *server* value identifies the server, the data is discarded.

System action: The message is written to the error log and processing continues.

User response: If the problem persists, contact your service organization.

CCL4523E TCP/IP Header validation failed with rc = *rc* for server *server*

Explanation: The header information on the

CCL4601E • CCL4608E

data received from the TCP/IP socket did not pass validation tests.

System action: The connection to the server is closed.

User response: If the problem persists, contact your service organization.

CCL4601E Memory allocation of display buffer failed

Explanation: There was insufficient free memory for the Client SNA protocol driver to retrieve LU 6.2 information from the APPC provider.

System action: The SNA protocol driver is not initialized.

User response: Free up some memory and retry.

CCL4602E Memory allocation of receive buffer pool failed

Explanation: There was insufficient free memory for the Client SNA protocol driver to receive data from any servers.

System action: The SNA protocol driver is not initialized.

User response: Free up some memory and retry.

CCL4603E Memory allocation of send buffer failed

Explanation: There was insufficient free memory for the Client SNA protocol driver to send data to any servers.

System action: The SNA protocol driver is not initialized.

User response: Free up some memory and retry.

CCL4605E Begin thread to handle inbound ATI requests failed

Explanation: Creation of thread, to handle inbound allocate requests from ATI to Client terminals, failed.

System action: The SNA protocol driver is not initialized.

User response: If the problem persists, contact your service organization.

CCL4606E Netname *partner LU name* has an invalid length for an alias name

Explanation: The specified *partner LU name* in the 'NETNAME' parameter in your configuration file has an invalid length for this SNA protocol driver and 'PARTNERLUALIAS' parameter setting. The maximum length of an alias name is 8 characters, the minimum length is 1.

System action: The connection to the partner LU is not started.

User response: Refer to the Administration book to determine whether or not this SNA protocol driver requires the use of alias names. Correct the 'NETNAME' and/or 'PARTNERLUALIAS' parameters in the configuration file and retry.

CCL4607E Netname *partner LU name* has an invalid length for a fully qualified name

Explanation: The specified *partner LU name* in the 'NETNAME' parameter in your configuration file has an invalid length for this SNA protocol driver and 'PARTNERLUALIAS' parameter setting. The maximum length of a fully qualified name is 17 characters (*****.*). The minimum length is 1.

System action: The connection to the partner LU is not started.

User response: Refer to the Administration book to determine whether or not this SNA protocol driver supports fully qualified names. Correct the 'NETNAME' and/or 'PARTNERLUALIAS' parameters in the configuration file and retry.

CCL4608E Memory allocation of link data block failed for *server* connection

Explanation: There was insufficient free

memory for the Client SNA protocol driver to open a link to the specified server.

System action: The Client continues.

User response: Free up some memory and retry.

CCL4609E Retrieval of LU 6.2 information for *server* connection failed, APPC return code *primary RC, secondary RC*

Explanation: The Client SNA protocol driver was unable to open a link to the specified server because retrieval of LU6.2 information from the APPC provider failed. The APPC DISPLAY verb failed with the specified return code.

System action: The Client continues.

User response: If the problem persists, contact your service organization.

CCL4610E LU 6.2 information for *server* connection not all returned on retrieval

Explanation: The Client SNA protocol driver was unable to open a link to the specified server because not all of the LU6.2 information could be obtained from the APPC provider.

System action: The Client continues.

User response: If the problem persists, contact your service organization.

CCL4611E Local LU *LU name* is not defined

Explanation: The Client SNA protocol driver was unable to open any link using this LU because it is not defined to IBM Communications Manager/2.

System action: The Client continues.

User response: Define the LU to the Communications SNA Server or correct the 'LocalLUName' parameter in your configuration file *ctg.ini* and retry.

CCL4612E Partner LU *partner LU name* is not defined

Explanation: The Client SNA protocol driver was unable to open any link using this partner LU because it is not defined to IBM Communications Manager/2.

System action: The Client continues.

User response: Define the partner LU to IBM Communications Manager/2 or correct the 'NetName' parameter in your configuration file *ctg.ini* and retry.

CCL4613E Communication Subsystem not loaded, APPC return code *primary RC*

Explanation: APPC could not execute the verb because Communications Manager had not started APPC. Either Communications Manager has not been started or it has not been configured correctly for the application.

System action: The Client continues.

User response: Start Communications Manager if it is not active. If the necessary APPC profiles are not configured, configure them and restart Communications Manager.

CCL4614E Open connection to *server* failed with APPC return code *primary RC, secondary RC*

Explanation: The APPC verb TP_STARTED failed with the specified return code. The connection to the specified server has not been started.

System action: The Client continues.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4615E Close connection to *server* failed with APPC return code *primary RC, secondary RC*

Explanation: The APPC verb TP_ENDED failed with the specified return code.

System action: The Client connection to the

specified server is closed.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4616E Memory allocation of conversation data block failed for server connection

Explanation: There was insufficient free memory for the Client SNA protocol driver to start a conversation with the specified server.

System action: The Client continues.

User response: Free up some memory and retry.

CCL4617E No contention-winner sessions free

Explanation: APPC could not allocate a conversation because no free sessions were available.

System action: The Client continues.

User response: Wait until one or more existing conversations have finished and retry.

CCL4618E SNASVCMG or CPSVCMG is not a valid mode name, APPC return code primary RC, secondary RC

Explanation: An attempt was made to allocate a Client conversation in SNASVCMG or CPSVCMG sessions mode. These are not valid modes.

System action: The Client continues.

User response: Change the 'Modename' parameter in the configuration file ctg.ini and retry.

CCL4619E Mode name mode name is not configured, APPC return code primary RC, secondary RC

Explanation: The specified mode has not been defined.

System action: The Client continues.

User response: Check whether the mode has been configured, and the spelling of the 'ModeName' parameter in the configuration file ctg.ini.

CCL4620E Allocate session to server failed with APPC return code primary RC, secondary RC

Explanation: The APPC verb ALLOCATE failed with the specified return code. A conversation with the specified server has not been started.

System action: The Client continues.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4621E Deallocate session to server failed with APPC return code primary RC, secondary RC

Explanation: The APPC verb DEALLOCATE failed with the specified return code.

System action: The Client continues.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4622E Begin thread to receive data for ATI conversation with server failed

Explanation: A Client thread could not be created to process the queue holding inbound attach requests; _beginthread failed.

System action: The SNA protocol driver is not initialized.

User response: Contact your system administrator.

CCL4623E Begin thread to receive data for conversation with server failed

Explanation: A Client thread could not be created to receive data from the server for the current conversation; `_beginthread` failed.

System action: The conversation is deallocated.

User response: Contact your system administrator.

CCL4624E Inbound ATI initialization failed with return code *return code*

Explanation: An attempt to allocate resources for inbound allocate requests from ATI to Client terminals has failed.

System action: The Client continues but all inbound request processing is terminated.

User response: Check SNA configuration for ATI Side Information Profile entry.

CCL4625E Error processing inbound ATI request, all ATI processing terminated

Explanation: An error occurred during an inbound ATI request.

System action: The Client continues but all inbound request processing is terminated.

User response: Contact your system administrator.

CCL4626E Attach Manager stopped, APPC return code *primary RC, secondary RC*

Explanation: The Client received an inbound ATI attach request but the receive allocate failed because the attach manager was stopped.

System action: The inbound allocate request is rejected.

User response: Start the attach manager.

CCL4628E Attempt to set SNA Logical Unit of Work id for this conversation failed with APPC return code *primary RC, secondary RC*

Explanation: The APPC verb `SET_TP_PROPERTIES` failed with the specified return code.

System action: The property change request is ignored. The client daemon attempts to continue. The SNA Logical Unit of Work (LUW) id for this conversation may not be unique.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4629E Receive_allocate for ATI request failed with APPC return code *primary RC, secondary RC*

Explanation: The APPC verb `RECEIVE_ALLOCATE` failed with the specified return code.

System action: The inbound allocate request is rejected.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4630E Send data to server failed with APPC return code *primary RC, secondary RC*

Explanation: The APPC verb `SEND_DATA` failed with the specified return code.

System action: The Client sends a deallocate abend to the server to terminate the conversation.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4631E Receive data from server failed with APPC return code *primary RC, secondary RC*

Explanation: The APPC verb `RECEIVE_AND_POST` failed with the specified return code and sense code.

System action: The client sends a deallocate abend to the server to terminate the conversation.

User response: An example return code is X'0003' received with sense code X'10086040'. This refers to an invalid attach parameter in the reply flow and can occur if ATTACHSEC=verify is specified on the connection definition in CICS, but the CICS region does not have security enabled. To rectify this, either set ATTACHSEC=local or enable security in CICS. Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4632E Error receiving data from server, APPC what_received return parameter is APPC return code

Explanation: An APPC RECEIVE returned an unexpected data_received or conversation status_received indicator.

System action: The Client sends a deallocate abend to the server to terminate the conversation.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4633E Memory allocation of receive buffer failed

Explanation: There was insufficient free memory for the Client SNA protocol driver to receive data on the current conversation.

System action: The Client sends a deallocate abend to the server to terminate the conversation.

User response: Free up some memory and retry the transaction.

CCL4634E Create semaphore for ReceiveAndPost failed, return code operating system return code

Explanation: An attempt to set post semaphore

for receiving conversation data failed.

System action: The Client sends a deallocate abend to the server to terminate the conversation.

User response: Contact your system administrator.

CCL4635E Allocate session to server failed, TP name TP name unknown, APPC return code primary RC, secondary RC

Explanation: The server rejected the incoming attach because the Client or Client application specified a TP name that the server did not recognize.

System action: The Client continues.

User response: Ensure that the specified TP is installed on the server.

CCL4636E Allocate session to server failed with APPC return code primary RC, secondary RC - retryable

Explanation: APPC could not allocate a conversation but retrying without intervention may succeed. APPC was probably unable to activate a link or a session.

System action: The Client continues.

User response: Retry the operation. If retrying does not succeed, see your APPC Programming Reference information for further information or contact your system administrator.

CCL4637E Allocate session to server failed with APPC return code primary RC, secondary RC - not retryable

Explanation: APPC could not allocate a conversation and intervention is required to correct the problem.

System action: The Client continues.

User response: Refer to your APPC Programming Reference information for further information or contact your system administrator.

CCL4638E **Prepare to receive data from *server* failed with APPC return code**
primary RC, secondary RC

Explanation: The APPC verb PREPARE_TO_RECEIVE failed with the specified return code.

System action: The Client sends a deallocate abend to the server to terminate the conversation.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4639E **Receive data from *server* failed with APPC return code** *primary RC, secondary RC*

Explanation: The APPC verb RECEIVE_IMMEDIATE failed with the specified return code.

System action: The Client sends a deallocate abend to the server to terminate the conversation.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4640E **Error receiving data from *server*, APPC what_received return parameter is APPC return code**

Explanation: An APPC RECEIVE returned an unexpected data_received or conversation status_received indicator.

System action: The Client sends a deallocate abend to the server to terminate the conversation.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4641E **Send data and deallocate to *server* failed with APPC return code**
primary RC, secondary RC

Explanation: The APPC verb SEND_DATA with DEALLOCATE type failed with the specified return code.

System action: The Client sends a deallocate abend to the server to terminate the conversation.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4642E **Invalid number of parameters defined for TP program *EXE name***

Explanation: The TP program was invoked because of an inbound allocate request but the IBM Communications Manager definition for the TP program has an invalid number of parameters defined.

System action: The inbound allocate request is rejected.

User response: Refer to the Administration book, correct the TP definition, and retry.

CCL4643E **Invalid TP name parameter defined for TP program *EXE name***

Explanation: The TP program was invoked because of an inbound allocate request but the IBM Communications Manager definition for the TP program has an invalid TP name parameter defined.

System action: The inbound allocate request is rejected.

User response: Refer to the Administration book, correct the TP definition, and retry.

CCL4648E **Receive data from *server* unsuccessful after partial data received, APPC return code**
primary RC, secondary RC

Explanation: The APPC verb

RECEIVE_IMMEDIATE failed with the specified return code after some data had already been received.

System action: The Client sends a deallocate abend to the server to terminate the conversation.

User response: Try altering the 'SnaRetryCount' parameter in the configuration file ctg.ini and retry. Refer to the Administration book.

CCL4649E Post on receipt of data from server failed with APPC return code primary RC, secondary RC

Explanation: The APPC verb POST_ON_RECEIPT failed with the specified return code.

System action: The Client sends a deallocate abend to the server to terminate the conversation.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4650E Partner LU name netname is not fully qualified

Explanation: The partner LU name specified in the 'NETNAME' parameter in your configuration file was not qualified by the SNA network name (NETID). The format is NETID.LU_NAME.

System action: The connection to the partner LU is not started.

User response: If using alias LU names, set the 'PARTNERLUALIAS' parameter to 'Y' and retry. If not using alias LU names, correct the 'NETNAME' parameter and retry.

CCL4651E Allocate session to server unsuccessful after number retry attempts, APPC return code primary RC, secondary_RC

Explanation: The attempt to allocate a session to the specified server was unsuccessful after the

number of retries specified in your configuration file ctg.ini.

System action: The Client continues.

User response: Contact your system administrator.

CCL4652E Conversation with server failed, APPC return code primary RC, retry

Explanation: A temporary failure prematurely ended the conversation.

System action: The Client continues.

User response: Retry the operation. If retrying does not succeed, see your APPC Programming Reference information for further information, or contact your system administrator.

CCL4653E Conversation with server failed, APPC return code primary RC, no retry

Explanation: A permanent failure prematurely ended the conversation. The link to the server has probably failed.

System action: The Client continues.

User response: Refer to your APPC Programming Reference information for further information or contact your system administrator.

CCL4654E CNOS for mode mode sessions to server failed, APPC return code primary RC, secondary RC

Explanation: The APPC verb CNOS failed with the specified return code. The connection to the specified server has not been started.

System action: The Client continues.

User response: Refer to your APPC Programming Reference information for further information, or contact your system administrator.

CCL4655E Partner LU *server* maximum session limit is zero, APPC return code *primary RC, secondary RC*

Explanation: The APPC verb CNOS failed because the local maximum session limit of the partner LU is 0. The connection to the specified server has not been started.

System action: The Client continues.

User response: Refer to your APPC Programming Reference information for further the particular error, or contact your system administrator.

CCL4656E Mode *mode* not recognized by partner LU, APPC return code *primary RC, secondary RC*

Explanation: The APPC verb CNOS failed as the partner LU does not recognize the specified mode name. The connection to the specified server has not been started.

System action: The Client continues.

User response: Refer to your APPC Programming Reference information for further the particular error, or contact your system administrator.

CCL4657E SNA Server has no active services

Explanation: The Client SNA protocol driver was unable to open a link to the specified server because SNA server has no active services.

System action: The Client continues.

User response: Activate the relevant service via SNA Server Administration or contact your system administrator.

CCL4658E WinAPPStartup failed with return code *Windows APPC RC*

Explanation: The Client SNA protocol driver was unable to open a link to the specified server because it failed to register itself with the currently installed Windows APPC implementation.

System action: The Client continues.

User response: Contact your system administrator.

CCL4659E Local LU *LU name* is not defined on an active service/node

Explanation: The Client SNA protocol driver was unable to open any link using this LU because it is not defined on an active SNA service/node.

System action: The Client continues.

User response: Activate the relevant service SNA service or define the LU to an active SNA service/node or correct the 'LocalLUName' parameter in your configuration file ctg.ini and retry.

CCL4660E Partner LU *partner LU name* is not defined in a connection with local LU *LU name*

Explanation: The Client SNA protocol driver was unable to open any link using this partner LU because it is not defined in a connection with the specified local LU on an active SNA service/node.

System action: The Client continues.

User response: Define a connection between the specified local and partner LUs on an active SNA service/node or correct the 'NetName' parameter in your configuration file ctg.ini and retry.

CCL4661E SNA stack component was not loaded or was terminated, APPC return code *primary RC*

Explanation: A required SNA stack component could not be loaded or was terminated while processing an APPC call. On Windows this error could indicate that the IBM CICS Transaction Gateway service does not have the required Windows domain logon authority to communicate with the Communications Server.

System action: IBM CICS Transaction Gateway continues but SNA connections are not available.

User response: Refer to the documentation for your Communications Server for possible causes of this error. If the problem persists contact your service organization.

CCL4662E **Load of SNA stack library DLL name failed, return code** *Windows return code*

Explanation: LoadLibrary failed, the Client could not load the specified dynamic link library.

System action: The Client SNA protocol driver is not loaded.

User response: Install and configure one of the supported products for SNA transport, if this has not already been done, and ensure that the DLL is in the LoadLibrary search path; then retry. Or contact your system administrator.

CCL4665E **TP program not available on server, APPC return code** *primary RC, secondary RC, retry*

Explanation: APPC could not start a remote TP program retrying without intervention may succeed.

System action: The Client continues.

User response: Retry the operation. If retrying does not succeed, see your APPC Programming Reference information for further information or contact your system administrator.

CCL4666E **TP program not available on server, APPC return code** *primary RC, secondary RC, no retry*

Explanation: APPC could not start a remote TP program and intervention is required to correct the problem. The remote server is probably not available.

System action: The Client continues.

User response: Ensure the remote server is available and if the problem persists see your APPC Programming Reference information for further information or contact your system administrator.

CCL4667E **SNA Server not configured for inbound requests, APPC return code** *primary RC, secondary RC*

Explanation: The APPC verb RECEIVE_ALLOCATE failed as SNA Server has not been correctly configured to receive inbound ATI requests.

System action: The Client continues.

User response: If inbound ATI request processing is required, refer to the Administration book on how to configure SNA Server.

CCL4668E **SNA node not started, APPC return code** *primary RC*

Explanation: The SNA node has not been started.

System action: The Client continues.

User response: Start the SNA node and retry the operation.

CCL4669E **SNA local LU name must be specified as an alias name**

Explanation: Local LU names must be specified as alias names in the configuration file ctg.ini.

System action: The connection to the SNA server is not started.

User response: Specify the local LU name in the configuration file ctg.ini as an alias name, set the parameter 'LOCALLUALIAS' to 'Y' and retry.

CCL4670E **SNA communications subsystem abended, APPC return code** *primary RC, secondary RC*

Explanation: The SNA communications server software has failed with the specified APPC return code.

System action: The connection to the server is not started.

User response: Check the SNA error log for the cause of the abend. Refer to the CICS Transaction

Gateway information center for the location of the SNA error log.

CCL4671E LU alias *LU alias name* is not defined

Explanation: The specified LU alias is not defined in the APPC provider.

System action: The connection to the server is not started.

User response: If using alias LU names, define the LU alias and retry. If not using alias LU names, set the 'LOCALLUALIAS' parameter to 'N' in the configuration file and retry.

CCL4672E Partner LU alias *LU alias name* or mode *mode name* is unknown, APPC return code *primary RC*, *secondary RC*

Explanation: If using alias LU names, either the partner LU alias supplied in the 'Netname' parameter in the configuration file ctg.ini is not defined, or the mode is not defined in the APPC provider. If not using alias LU names, the specified mode is not defined.

System action: The connection to the server is not started.

User response: Check that the parameters in the configuration file ctg.ini match the definitions in the APPC provider.

CCL4673E Conversation with *server* failed, APPC return code *primary RC*, *secondary RC*

Explanation: There has been a failure to allocate a conversation or to receive data from the specified server. The link to the server has probably failed.

System action: The Client continues.

User response: Refer to your APPC Programming Reference information for further information or contact your system administrator.

CCL4674E TP program *TP name* abended on server *server*, APPC return code *primary RC*

Explanation: The specified TP program abended on the given server. If the TP program is CTIN, then this may be because the server has been recycled since running CCIN.

System action: The Client continues. If the TP program is CTIN, the Client will attempt to reconnect to the server and retry the request.

User response: Contact your system administrator.

CCL4675E Wait on semaphore for ReceiveAndPost failed, return code *operating system return code*

Explanation: An attempt to wait on a semaphore for conversation data, after successful completion of the receive verb, failed.

System action: The Client sends a deallocate abend to the server to terminate the conversation.

User response: Contact your system administrator.

CCL4676E Protocol driver failed to notify Client daemon of a data event, return code *internal return code*

Explanation: The protocol driver's attempt to notify the Client of a data event failed.

System action: The protocol driver sends a deallocate abend to the server to terminate the conversation.

User response: Contact your system administrator.

CCL4677E Callback for RECEIVE_AND_POST for unknown convid *APPC conv_id* received

Explanation: The protocol driver's callback function was called for the specified conversation, but it was not in the list of

conversations for which a RECEIVE_AND_POST call had been made.

System action: The event is ignored.

User response: Contact your system administrator.

CCL4678W The environment variable LD_PRELOAD is not set

Explanation: When IBM Communications Server is installed and the SNA protocol is being used to connect to CICS you are required to set LD_PRELOAD. When the Remote API Client is installed LD_PRELOAD is not required. For security reasons LD_ variables are not inherited by the root user.

System action: This warning message is written to the Client daemon error log.

User response: If using the Remote API Client no action is necessary. If using IBM Communications Server, refer to the configuration documentation for the SNA transport product and the IBM Communications Server for Linux configuration section in the product information center. A number of environment variables must be correctly defined for the correct operation of SNA protocol driver. Stop the Client daemon, make sure the named variable is correctly defined in the environment where the Client daemon will run, and then retry.

CCL4679E Environment variable *name* = *value* caused a program error

Explanation: Environment variable *name* caused an error in the Client daemon.

System action: The Client daemon continues, but the SNA protocol is unavailable.

User response: Check that the value of the *name* environment variable is correct. Refer to the IBM Communications Server for Linux configuration section in the product information center for further information.

CCL4680E Server *name* has inconsistent values for parameters *parameter1* and *parameter2*

Explanation: In the configuration file in the SERVER SECTION for server *name*, the combination of values for the specified parameters is not supported by the SNA protocol driver.

System action: The server *name* fails to start.

User response: Correct the errors by running the configuration tool.

CCL4687E Partner LU alias *Partner LU alias name* is not defined

Explanation: The partner LU alias specified in the Netname parameter in the configuration file is not defined in the SNA communications product. Note that the partner LU alias parameter is case sensitive.

System action: The connection to the server is not started.

User response: If using alias LU names, define the partner LU alias in your SNA communications product to match that defined in the configuration file and retry. If not using alias LU names, set the 'PARTNERLUALIAS' parameter to 'N' in the configuration file and retry.

CTG6103E The '*program*' launcher program cannot be found in your PATH

Explanation: The *program* is required to run the command but the *program* is not in the current PATH.

System action: The command terminates with return code 4.

User response: Identify where the *program* is on your system. Add its location to your PATH or for UNIX and Linux systems use CTG_JAVA to define where the *program* is located. Retry the command. See the Problem Determination section in the Administration book for your platform for further information.

CTG6106E Use command 'ctgmsgs' to list the recognized languages as the language *language code* is not recognized

Explanation: CICS Transaction Gateway supports a number of language and code set combinations, but the specified language code is not supported on this platform.

System action: The message language will not be changed.

User response: Run the 'ctgmsgs' command with the '-?' parameter to get a list of supported language and code set combinations. Choose the combination that best meets your requirements and retry with that combination.

CTG6107E Use command 'ctgmsgs' to list the recognized code sets as the code set *code set* is not recognized

Explanation: CICS Transaction Gateway supports a number of language and code set combinations, but the specified code set is not supported on this platform.

System action: The message language will not be changed.

User response: Run the 'ctgmsgs' command with the '-?' parameter to get a list of supported language and code set combinations. Choose the combination that best meets your requirements and retry with that combination.

CTG6109E The JVM is not at the required level to run this application

Explanation: The Java applications supplied with CICS Transaction Gateway require a minimum level of JVM.

System action: The Java application will not start.

User response: See the Administration book for details of the versions of Java supported. Ensure that the correct level of Java is available on your system and add access to it from the PATH or for UNIX and Linux systems use CTG_JAVA to

define where the Java launcher program is located.

CTG6110E Check the value of environment variable 'CTGENVVAR' as the CTGENVVAR file *HFS path* not found

Explanation: CICS Transaction Gateway on z/OS uses the 'ctgenvar' script to initialize environment variables before executing its Java component. If the CTGENVVAR environment variable is set, then the ctgstart script will attempt to run that script. Note that the CTGENVVAR variable is intended for use only when the Gateway daemon is started under a UNIX System Service shell (as opposed to batch).

System action: The CICS Transaction Gateway will not start.

User response: Verify that the file *HFS path* is correct. If this value is as expected, then verify that the file is in place. Note that the value of CTGENVVAR is a HFS path, and therefore case-sensitive.

CTG6111I Using the default configuration script in file *HFS path* to start up the application

Explanation: Information message

CTG6114W CICS Transaction Gateway is already installed in the local code page

Explanation: ctgmsgs4local was called to convert the CICS Transaction Gateway to the local code page, but it is already installed for that code page.

System action: The script terminates.

User response: No further action as CICS Transaction Gateway is in the local code page as requested.

CTG6116E This product has not been properly packaged

Explanation: The script that is running is

incomplete. This indicates that there was an error during packaging.

System action: The script terminates.

User response: Contact your service organization.

CTG6117E Unable to recognize the platform from the 'uname' string *text*

Explanation: The script that is running is unable to identify a supported operating system using the *text* output from the 'uname' command. Either the product is being run on an unsupported platform, or an update to the operating system version has changed this text in an unexpected way.

System action: The script terminates.

User response: Contact your service organization.

CTG6118E This product is packaged for the *target* operating system, but has identified this as the *actual* operating system

Explanation: CICS Transaction Gateway is packaged for a number of different operating systems. This package is for the operating system identified by the string *target* . An operating system version check (uname) shows the operating system you are running to be *actual* and so the installation cannot continue.

System action: The script terminates.

User response: Remove the current installation and install the correct package for your operating system. If you believe that you have installed the correct package, contact your service organization.

CTG6119E This product will not execute on z/OS.e

Explanation: CICS Transaction Gateway for z/OS can only communicate with CICS in a local region (EXCI is the only protocol provided). As CICS will not run under z/OS.e, there cannot be a local CICS region to communicate with. For

this reason CICS Transaction Gateway is not supported on the z/os.e operating system.

System action: The script terminates.

User response: Remove the current installation and install it on a system running a full version of z/OS.

CTG6120E Unable to recognize the platform from the 'uname' string *text*, because TMPDIR *path* maybe be full or unavailable for write access

Explanation: The script is unable to identify the runtime operating system as a command substitution using 'uname' has returned an empty string. This can be due to the temporary file system being inaccessible for write by the executing userid, or simply being full. The temporary file system is located at /tmp by default, but can be overridden by the TMPDIR environment variable.

System action: The script terminates.

User response: Ensure that the userid executing the script has write access to the tmp filesystem, and that the tmp filesystem is not full. If both of these requirements are satisfied and the problem persists, then contact your service organization.

CTG6121E Unable to find Java runtime classes

Explanation: When trying to run a Java application, the Java runtime classes could not be found.

System action: The command line application terminates.

User response: Install the Java Virtual Machine (JVM) correctly.

CTG6122E The CICS Transaction Gateway could not execute the command

Explanation: The CICS Transaction Gateway could not execute the command. Either the CICS TG is not correctly installed, or the environment is not setup correctly.

System action: The application terminates.

User response: Check for exceptions that indicate a problem with the environment. If there are no obvious problems, reinstall the CICS TG.

CTG6123E Detected Operating System is not supported for this application

Explanation: The Operating System you are running, as detected by the JVM, is unsupported.

System action: The application terminates.

User response: Ensure that you are running a supported Operating System.

CTG6124W Source file *filename* not found or not executable

Explanation: The migration tool ctgconvnv cannot find the specified file, or the file does not have execute permission for this user. The syntax of the command is: ctgconvnv ['wrap width'] ['source file name'] where: ['wrap width'] is the width of the generated output and is a value in the range 40 to 120. The default is the terminal screen width. ['source file name'] is the ctgenvvar script to be converted. If the source file name is not fully qualified, the default directory is the <install_path>/bin directory. If the source file name is not specified, a filename of ctgenvvar is assumed. Output is written to stdout. This can be redirected to a file by standard operating system facilities.

System action: The ctgconvnv application terminates.

User response: If the specified file exists but is not executable, correct the file permissions and retry. If the displayed file name is not correct, retry the command using the full HFS file name.

CTG6125I Usage - ctgd {start|stop}

Explanation: Information message

CTG6126E Cannot find ctgd.conf at *location*

Explanation: An attempt was made to start the Gateway daemon in the background, but a valid

configuration file could not be found at the specified location.

System action: The Gateway daemon does not start.

User response: Check that a valid file exists. Refer to the Administration book for details on the format and location of ctgd.conf.

CTG6127E Unable to start the Gateway daemon because the group to which *userid* belongs cannot be resolved

Explanation: An attempt was made to start the Gateway daemon in the background, but it was not possible to resolve the group to which the running user belongs.

System action: The Gateway daemon does not start.

User response: Check that the /etc/group file exists and that the file has read permissions for the user running ctgd. To verify whether the group can be resolved, run the UNIX 'id' command as the user.

CTG6129E ctgd was unable to stop the Gateway daemon on port *port*

Explanation: An attempt to stop the Gateway daemon using the ctgd stop command failed. This might be because the Gateway daemon is not currently running. A detailed error message is written to the console.

System action: The Gateway daemon, if running, might not stop.

User response: Investigate why ctgd was unable to stop the Gateway daemon and ensure that the administration port specified in ctgd.conf matches that of the running Gateway daemon. If the problem persists, manually kill the Gateway daemon processes if it is still running.

CTG6130I Using the configuration script in the CTGENVVAR file *HFS path* to start up the application

Explanation: Information message

CTG6131W Assuming the default location for CTG.INI as the Environment variable 'CICSCLI' is not set

Explanation: The Gateway daemon will attempt to use the default configuration file 'ctg.ini' or 'CTG.INI' in the product bin directory. This is a compatibility feature and not the recommended configuration style.

System action: The ctgstart script continues. If a Gateway daemon configuration file does not exist in the product bin directory, then the initialization will fail.

User response: Verify the intended location of the Gateway daemon configuration file and set the CICSCLI environment variable if required.

CTG6132W Address space sharing will depend upon system defaults as the Environment variable '_BPX_SHAREAS' is not set

Explanation: The ctgstart script has determined that the environment variable '_BPX_SHAREAS' has not been defined.

System action: The ctgstart script continues.

User response: If the environment variable _BPX_SHAREAS is not set, then whether or not the Gateway daemon runs in a single address space will be subject to the system default. Verify your preferred setting for _BPX_SHAREAS and add the appropriate definition to the environment variable configuration. The recommended setting is _BPX_SHAREAS=YES.

CTG6133W Environment variable 'CTG_SWAPPABLE' is set to 'YES', but this is not a recommended configuration

Explanation: The ctgstart script has determined that the environment variable 'CTG_SWAPPABLE' has been defined as 'YES'.

System action: The ctgstart script continues.

User response: Setting CTG_SWAPPABLE to 'YES' allows the Gateway daemon address space to be swapped out by z/OS. This is not a

recommended configuration for the Gateway daemon. Consider either removing CTG_SWAPPABLE from the environment variable configuration (allowing it to default to NO), or explicitly define it with the value 'NO'.

CTG6140E Environment variable CTG_JAVA must specify the 'java' launcher program, CTG_JAVA = *program*

Explanation: The *program* specified by environment variable CTG_JAVA is not a supported *java* launcher.

System action: The command terminates with return code 4.

User response: Check that the environment variable CTG_JAVA is correctly defined.

CTG6141E The java launcher specified by environment variable CTG_JAVA does not exist, CTG_JAVA = *program*

Explanation: The *program* defined by environment variable CTG_JAVA does not exist.

System action: The command terminates with return code 4.

User response: Check that the environment variable CTG_JAVA is correctly defined, check that the *program* exists and that directory permissions allow the user to access the *program*.

CTG6142E The java launcher specified by environment variable CTG_JAVA can not be run, CTG_JAVA = *program*

Explanation: The *program* defined by environment variable CTG_JAVA could not be run.

System action: The command terminates with return code 4.

User response: Check that the *program* has the correct file permissions, ensure that the user can run the *program*.

CTG6400I CICS Transaction Gateway is starting

Explanation: Information message

CTG6401E Provider *provider* failed to register statistic *statistic* due to *reason*

Explanation: An internal error prevented statistic *statistic* from being registered with the CICS TG.

System action: The statistic is not available through ctgadmin, SDSF or the statistics API.

User response: Contact your service organization.

CTG6402E Provider *provider* failed to update statistic *statistic* due to *reason*

Explanation: An internal error prevented statistic *statistic* from being updated.

System action: The statistic is not available through ctgadmin, SDSF or the statistics API.

User response: Contact your service organization.

CTG6403E The statistic group *statistic group* could not be created because proxy class *proxy class* could not be found

Explanation: An internal error prevented statistic group *statistic group* from being created.

System action: No statistics for the statistic group will be available through ctgadmin, SDSF or the statistics API.

User response: Contact your service organization.

CTG6404W Ignoring sotimeout setting in the configuration file

Explanation: The configuration file contains an entry sotimeout. This setting (Handler wakeup timeout field in the Configuration Tool) used to define the time that might have to elapse before the Gateway daemon could shut down. It

became redundant when Version 6 of the CICS TG introduced shutdown commands, and from Version 7 is no longer supported.

System action: The Gateway daemon continues its initialization process.

User response: To suppress this message in future, remove any reference to sotimeout from the configuration file. Open the configuration file in the Configuration Tool and then save it.

CTG6405E The Gateway daemon has terminated abnormally

Explanation: The Gateway daemon terminated abnormally.

System action: This message is output and the Gateway daemon ends.

User response: Consult the log for the causes of this problem, correct the problem, and restart the Gateway daemon.

CTG6406W The maximum worker threads has been reduced to the EXCI logon limit *logonlim*

Explanation: The maximum number of worker threads was previously configured as unlimited, or to a value which exceeded the number of EXCI pipes available to the CICS TG. Requests fail if the number of worker threads exceeds the EXCI pipe limit. The maximum number of worker threads has been reduced.

System action: The maximum number of worker threads has been set to equal the maximum number of EXCI pipes available.

User response: To prevent a warning message for this error choose one of the following options: configure the maximum worker threads to less than or equal to the EXCI pipes logon limit or, increase the EXCI logon limit to greater than, or equal to, the number of worker threads.

CTG6407E Statistics interval length is set to the default as unable to parse 'statint' value *statint value*

Explanation: The statistics interval length

(statint) value could not be parsed

System action: The statistics interval length value is set to the default

User response: The statistics interval length (statint) value should be in the format HHMMSS, where HH is the number of hours in the range 00-24, MM is the number of minutes in the range 00-59 and SS is the number of seconds in the range 00-59

CTG6408E The value *statint value* (HHMMSS) specified for the statistics interval length (statint) is not valid

Explanation: The statistics interval length (statint) value, specified in the configuration file, is not valid. The interval length must be between 1 minute (000100) and 24 hours (240000).

System action: The statistics interval length value is set to the default.

User response: The statistics interval length (statint) value should be in the format HHMMSS, where HH is the number of hours in the range 00-24, MM is the number of minutes in the range 00-59 and SS is the number of seconds in the range 00-59.

CTG6409E Statistics end of day is set to the default as unable to parse 'stateod' value *stateod value*

Explanation: The statistics end of day (stateod) value could not be parsed

System action: The statistics end of day value is set to the default

User response: The statistics end of day (stateod) value should be in the format HHMMSS, where HH is the number of hours in the range 00-23, MM is the number of minutes in the range 00-59 and SS is the number of seconds in the range 00-59

CTG6410E The value *stateod value* (HHMMSS) specified for the statistics end of day (stateod) is not valid

Explanation: The statistics end of day (stateod) value, specified in the configuration file, is not valid. The end of day value must be between midnight (000000) and one second before midnight (235959)

System action: The statistics end of day value is set to the default

User response: The statistics end of day (stateod) value should be in the format HHMMSS, where HH is the number of hours in the range 00-23, MM is the number of minutes in the range 00-59 and SS is the number of seconds in the range 00-59

CTG6411I Interval statistics are active. The statistics interval length is set to *hours hours, minutes minutes and seconds seconds*. The statistics end of day time is set to *hours :minutes :seconds locale* .

Explanation: Information message

CTG6412W The time calculated for the next statistics interval reset is in the past. Recalculating based on the end of day time. The next statistics interval reset will occur at *reset time* .

Explanation: The time calculated for the next statistics interval reset is in the past. This could be due to resetting the system clock.

System action: The time for the next statistics interval will be recalculated based on the end of day time.

User response: Check if the system clock has been moved forwards in time. If not contact your service organization.

CTG6413W The time calculated for the next interval is greater than the statistics interval length setting. Recalculating based on the end of day time. The next statistics interval reset will occur at *reset time* .

Explanation: The time to allow for the next statistics interval is greater than the statistics interval length setting. This could be due to resetting the system clock.

System action: The time for the next statistics interval will be recalculated based on the end of day time.

User response: Check if the system clock has been moved backwards in time. If not contact your service organization.

CTG6414E **The resource group *statistic group* could not be added to the current interval statistics because the corresponding proxy class could not be found**

Explanation: An internal error prevented statistic group *statistic group* from being added to the current interval statistics.

System action: The statistic group will be not be included in the interval statistics.

User response: Contact your service organization.

CTG6481I **-keyringpw=<keyring_pw> - keyring password for use with the ssl protocol**

Explanation: Information message

CTG6482I **-keyring=<keyring> - keyring for use with the ssl protocol**

Explanation: Information message

CTG6483I **-sslport=<port_number> - SSL port for ssl protocol**

Explanation: Information message

CTG6484W **'keyringpw' parameter ignored as supplied without keyring parameter**

Explanation: The keyringpw parameter has been supplied without a keyring parameter also being specified.

System action: CICS Transaction Gateway ignores the keyringpw parameter and continues the startup process.

User response: In order to use SSL both the keyring and keyringpw parameters must be specified.

CTG6485I **Successfully started handler for the *protocol* : protocol on port *port*, bound to address *address***

Explanation: Information message

CTG6486W **Console input has been disabled**

Explanation: The Gateway daemon was unable to read from the console so console input has been disabled. The Gateway daemon cannot be shut down from the console.

System action: The Gateway daemon continues with console input disabled.

User response: To shut down the Gateway daemon on z/OS, enter an MVS MODIFY command specifying the SHUT action. To quiesce the Gateway daemon on Multiplatforms, use the ctgadmin command specifying the shut action.

CTG6488E **Gateway daemon is shutting down, so rejecting the request**

Explanation: A request has been flowed to the CICS Transaction Gateway. This is not allowed while the Gateway daemon is shutting down.

System action: The Gateway daemon continues processing outstanding work during the shutdown process.

User response: Check with your System Administrator as to when the CICS Transaction Gateway will be available again.

CTG6489W **128bitonly cannot be used in conjunction with other cipher suites**

Explanation: The configuration file contains the deprecated parameter 128bitonly as well as explicitly listed cipher suites. When selecting

cipher suites for the SSL protocol handler to use, you must explicitly list each cipher suite in the configuration file.

System action: The handler continues to start using the cipher suites that are explicitly listed. The cipher suites referenced by the 128bitonly parameter are not enabled unless they are explicitly listed in the configuration file.

User response: Remove the 128bitonly parameter from the configuration file and restart the Gateway daemon to remove this warning message.

CTG6490I Normal shutdown of Gateway daemon started by *user* or *systems management program*

Explanation: Information message

CTG6491I Number of active clients is *INSERT-0*

Explanation: Information message

CTG6492I Number of active connections at immediate shutdown request was *INSERT-0*

Explanation: Information message

CTG6493I Q or - for normal shutdown

Explanation: Information message

CTG6494I I for immediate shutdown

Explanation: Information message

CTG6495E No cipher suites available for use by SSL connection

Explanation: None of the cipher suites specified in the configuration file are available for use by the CICS Transaction Gateway.

System action: The SSL protocol handler fails to start.

User response: Ensure that the configuration

file specifies only cipher suites that are available to the level of Java used by the CICS Transaction Gateway.

CTG6496W The use of 128bitonly has been deprecated

Explanation: Requesting that only 128 bit cipher suites are used through the use of 'ciphersuites=128bitonly' has been deprecated and might be removed at a later date.

System action: The SSL protocol handler is started and configured to only use the 128bit cipher suites that are documented in the Administration book.

User response: Use the Configuration Tool to explicitly list the cipher suites you want the SSL protocol handler to use.

CTG6497W Cipher suite *cipher* is not available for SSL connections

Explanation: The specified cipher suite *cipher* cannot be used by the CICS Transaction Gateway for SSL connections.

System action: The handler continues to start using the cipher suites that are supported.

User response: Correct the cipher suites in the configuration file and restart the Gateway daemon to remove this warning message.

CTG6498W Provided password was not used to access ESM key ring

Explanation: A password has been specified using the keyringpw parameter in the configuration file ctg.ini, at the same time as the esmkeyring parameter.

System action: The protocol handler is started but the password provided is not used to try and read the key ring from the ESM provider.

User response: Remove the keyringpw parameter if the key ring is stored in an ESM implementation, or the esmkeyring parameter if the key ring file is stored in the HFS.

CTG6499E **The CICS Transaction Gateway does not support the use of SystemSSL**

Explanation: A protocol string for either SystemSSL or SystemHTTPS is in the configuration file ctg.ini.

System action: The Gateway daemon continues but does not start the specified protocol handler.

User response: Migrate to JSSE-based SSL and HTTPS handlers and then remove the protocol strings from the configuration file, either manually or by loading the file into the Configuration Tool and saving it.

CTG6502I **Initial Connection managers = *n1*,
Maximum Connection managers =
*n2***

Explanation: Information message

CTG6505I **Successfully created the initial
connection manager and Worker
threads**

Explanation: Information message

CTG6506I **Client connected:** [*connection
manager*] - *protocol*

Explanation: Information message

CTG6507I **Client disconnected:** [*connection
manager*] - *protocol*, **reason =** *reason*

Explanation: Information message

CTG6508I **To shut down the Gateway
daemon type**

Explanation: Information message

CTG6509I **Immediate shutdown of Gateway
daemon started by user or systems
management program**

Explanation: Information message

CTG6510I **TCP/IP hostnames will not be
shown**

Explanation: Information message

CTG6511I **Gateway daemon has shut down**

Explanation: Information message

CTG6512I **CICS Transaction Gateway
initialization complete**

Explanation: Information message

CTG6513E **CICS Transaction Gateway failed
to initialize**

Explanation: The CICS Transaction Gateway could not start.

System action: The Gateway daemon process ends with a non-zero return code.

User response: Check the logs for error messages and correct any reported problems.

CTG6516W **Outstanding work in progress:**
[*connection manager*] [*count*]

Explanation: A client has disconnected from connection manager *connection manager*, but *count* work requests are still in progress.

System action: A connection manager will not become available to handle another connection until the work has been finished.

User response: Wait for requests to finish.

CTG6518W **Cleanups for previous requests
are outstanding:** [*name*] [*count*]

Explanation: A client has disconnected from connection manager *name*, but *count* requests which require completion are still outstanding.

System action: The connection manager will attempt to clean up the allocated resources.

User response: For information only.

CTG6524I **Successfully started handler for**

the *protocol* : protocol on port *port*

Explanation: Information message

CTG6525E Unable to start handler for the *protocol* : protocol, port: *port*, because: [RC]

Explanation: Where *protocol* specifies the protocol type, *port* specifies the port and *RC* specifies the cause of the error. The most likely cause is that the protocol handler is attempting to listen on a TCP/IP port that is already in use.

System action: The Gateway daemon is forced to stop because of *protocol* protocol handler error.

User response: Check the configuration settings of the protocol handler.

CTG6526I Initial workers = *count*, Maximum workers = *maximum*

Explanation: Information message

CTG6528W Outstanding work still in progress: [*name*] [*count*]

Explanation: A client has disconnected from connection manager *name*, with work requests in progress. *count* requests have failed to finish within a reasonable time.

System action: The connection manager marks itself as available again.

User response: For information only.

CTG6529W Unable to send reply to Java client: [*name*]

Explanation: The Gateway worker thread *name* was unable to send the reply to the client. This could be because the client was disconnected.

System action: System action is not required.

User response: Check if client is still running.

CTG6531W Unable to disconnect idle Java client - work still in progress: [*name*] - *client*

Explanation: The client at *client* has been idle for too long, but there are still work requests in progress on its behalf.

System action: Connection manager did not disconnect the client.

User response: For information only.

CTG6533E Unable to read the configuration file: [*reason*]

Explanation: Where *reason* specifies the reason for the failure.

System action: The Gateway stops; it will not run without a configuration file.

User response: Check that *ctg.ini* exists in the bin directory, or that the configuration file specified through the CICSCLI environment variable exists. Run the configuration tool if necessary to create a configuration file.

CTG6536I Generic ECI reply calls will not be allowed

Explanation: Information message

CTG6537I Security classes will be required for all connections on all protocols

Explanation: Information message

CTG6542E The CICS Transaction Gateway does not support TCPAdmin for remote administration

Explanation: Configuration file *ctg.ini* contains entries for the TCPAdmin protocol, which is no longer supported.

System action: The Gateway daemon continues its initialization process.

User response: To prevent this message occurring in future, use the Configuration tool to open and then save your configuration file. This removes the invalid entries.

CTG6543E The value specified for the

parameter **parameter is invalid**

Explanation: The specified command line parameter has an invalid value.

System action: The CICS Transaction Gateway fails to start.

User response: Ensure that the value is valid for the given parameter. Refer to the Administration book for details.

CTG6546W Gateway daemon startup parameter *parameter* will be ignored

Explanation: The specified parameter was found during Gateway daemon startup. The parameter has no effect.

System action: The parameter is ignored. The Gateway daemon continues.

User response: Stop using the parameter override or remove the parameter from the configuration file.

CTG6547W Gateway daemon will display symbolic TCP/IP hostnames in messages

Explanation: The display of symbolic hostnames might affect system performance.

System action: The Gateway daemon continues processing.

User response: Deselect the option to display TCP/IP hostnames using the configuration tool, or edit the configuration file and set DNSNAMES=OFF, to avoid this. Refer to the CICS Transaction Gateway Administration book for further details.

CTG6548I This command starts the CICS Transaction Gateway

Explanation: Information message

CTG6549I The following command line options can be specified

Explanation: Information message

CTG6550E Unable to listen on requested port

Explanation: Gateway is unable to recognize port number.

System action: System action is not required.

User response: Check valid port number, protocol and server name.

CTG6551E Unable to create requested connection manager and worker threads

Explanation: Usually this means that the system is running low on resources.

System action: System action is not required.

User response: Reduce the number of connection manager and worker threads initially created.

CTG6552E Connection closed because of error [*error*] whilst accepting a connection from *server*

Explanation: A connection was accepted from *server*, but a problem of type *error* occurred.

System action: Unable to connect.

User response: Check error code.

CTG6553E Error reading request: [*connection manager*]

Explanation: Where *connection manager* specifies the connection manager thread.

System action: Unable to execute request.

User response: Check if connection is still available.

CTG6554E Error in native method: [*worker thread*]

Explanation: Where *worker thread* specifies the worker thread.

System action: System action is not required.

User response: For information only.

CTG6555E Error writing reply: [*worker thread*]

Explanation: Where *worker thread* specifies the worker thread.

System action: Unable to write reply.

User response: Check if connection is still available.

CTG6556E Error copying request on local Gateway: [*error*]

Explanation: A problem has occurred in the CICS Transaction Gateway local mode support.

System action: The CICS Transaction Gateway is unable to process the request.

User response: Check the Gateway protocol settings.

CTG6557E Error whilst executing request: [*name*]

Explanation: An error occurred while processing request with name [*name*]

System action: System action is not required.

User response: Check that request is valid.

CTG6558E No TCP or SSL protocol handlers defined

Explanation: No TCP or SSL protocol handlers are defined in the Gateway daemon configuration file.

System action: The Gateway fails to start.

User response: Ensure that there is at least one TCP or SSL protocol handler definition in the Gateway SECTION of the configuration file.

CTG6559E Unexpected exception occurred: [*type*]

Explanation: An exception of type [*type*] occurred.

System action: System action is not required.

User response: For information only.

CTG6560E Unable to accept request of type *type* : [*name*]

Explanation: Connection manager *name* has received a request *type* but failed to load the required class file to execute the request.

System action: Gateway exits and displays error message.

User response: Make sure that the *type* class is in the CLASSPATH of the Gateway process.

CTG6561E Unable to use class class to provide security to *name*

Explanation: The *class* class could not be successfully loaded to provide security to the connection from *name* .

System action: The connection will be closed.

User response: Make sure that the *class* class is in the CLASSPATH of the Gateway process.

CTG6562E Connection to *name* rejected due to insufficient Connection managers

Explanation: A connection was accepted from *name*, but a connection manager did not become available in time.

System action: The connection was rejected.

User response: Increase the maximum number of Connection managers.

CTG6563E *name* protocol handler exited unexpectedly: [*description*]

Explanation: The *name* protocol handler exited due to problem *description* .

System action: Gateway exits and displays error message.

User response: Check error code.

CTG6564W Restart will be attempted as the protocol protocol handler has been stopped because of persistent errors

Explanation: The *protocol* protocol handler has been stopped because of errors on 50 consecutive socket accept calls. This can happen if the TCP/IP stack is stopped while the Gateway daemon is running. Exceptions from the failed accept attempts have been logged; the exception messages give additional information on the nature of the error.

System action: The Gateway daemon continues without this protocol handler. An attempt to restart the protocol handler will be made within 30 seconds. If this fails further restarts will be attempted every 30 seconds.

User response: Check the logged exceptions and correct the cause of the socket errors.

CTG6565I **-classpath=<classpath> - additional entries to append to JVM classpath**

Explanation: Information message

CTG6566W **Remote client *name* timed out during SSL handshake, connecttimeout is set to *unsigned ms***

Explanation: The attempted connection from the remote client timed out during the SSL handshake

System action: The connection is terminated

User response: Either check why the client was slow to complete the SSL handshake, or increase the SSL protocol handler connecttimeout value

CTG6567I **-requestExits=<exits> - list of classes to be used for request exit monitoring**

Explanation: Information message

CTG6568I **-statsport=<port_number> - TCP/IP port for statistics API requests**

Explanation: Information message

CTG6569E **Unable to open file *filename***

Explanation: File with name *filename* could not be opened.

System action: System action is not required.

User response: Check that file is not in use or read-only.

CTG6570I **Processing file *INSERT-0***

Explanation: Information message

CTG6571I **Creating file *INSERT-0***

Explanation: Information message

CTG6572I **Renaming file *INSERT-0* to *INSERT-1***

Explanation: Information message

CTG6573I **Processing is complete**

Explanation: Information message

CTG6574I **Connection logging has been disabled**

Explanation: Information message

CTG6575I **-port=<port_number> - TCP/IP port for tcp protocol**

Explanation: Information message

CTG6576I **-truncationsize=<number> - Maximum size of trace data blocks in bytes**

Explanation: Information message

CTG6577I **Java details: Version=*INSERT-0*, Vendor=*INSERT-1*, Path=*INSERT-2***

Explanation: Information message

CTG6578I **-dumpoffset=<number> - Byte offset in data to start trace output**

Explanation: Information message

CTG6579I -stack - Switch on exception stack trace

Explanation: Information message

CTG6580E The parameter *parameter* in the configuration file is unrecognized or invalid

Explanation: An unknown parameter starting with the specified text was encountered in the configuration file ctg.ini.

System action: The Gateway daemon terminates.

User response: Correct the configuration file error and restart the Gateway daemon. The configuration tool can be used to correct a badly-formed configuration file.

CTG6581E The Gateway daemon cannot continue

Explanation: An error was detected.

System action: The Gateway daemon terminates.

User response: Examine any other messages and the Gateway error log to determine the cause of the error.

CTG6582E The command line option *option* is unknown

Explanation: An unknown option starting with specified text was encountered on the command line.

System action: The Gateway daemon terminates.

User response: Restart CICS Transaction Gateway with a correct command line option. Use 'ctgstart -?' to obtain a list of supported options, or refer to the Administration book.

CTG6583I -initconnect=<number> - Initial number of connection manager threads

Explanation: Information message

CTG6584I -maxconnect=<number> - Maximum number of connection manager threads

Explanation: Information message

CTG6585I -initworker=<number> - Initial number of worker threads

Explanation: Information message

CTG6586I -maxworker=<number> - Maximum number of worker threads

Explanation: Information message

CTG6587I -trace - Enable standard trace

Explanation: Information message

CTG6588I -noinput - Disable reading of input from the console

Explanation: Information message

CTG6589W The adminport parameter has been ignored as the CICS Transaction Gateway for z/OS does not support it

Explanation: The configuration file ctg.ini contains the adminport parameter, or the Gateway daemon was started and the -adminport command line parameter was specified. The Gateway daemon on z/OS does not support this parameter, which was ignored.

System action: The Gateway daemon continues its initialization process.

User response: To prevent this message occurring in future, remove the adminport parameter from configuration file ctg.ini.

CTG6590I -dnsnames - Use DNS to display symbolic TCP/IP hostnames

Explanation: Information message

CTG6591I Parameter overrides override those in the configuration file

Explanation: Information message

CTG6592W Both 'CTG.INI' and 'ctg.ini' configuration files exist, using default file 'ctg.ini'

Explanation: The default directory contains both uppercase and lowercase versions of the configuration file.

System action: The Gateway daemon will use the default file 'ctg.ini' as no configuration filename has been specified. The contents of file 'CTG.INI' will be ignored.

User response: To avoid this warning being issued, explicitly specify a configuration file or delete file 'CTG.INI' from the default directory. For details of how to specify a configuration file see the Configuration chapter of the Administration book.

CTG6593I ctgstart <options> [<-j>JVMArg1 <-j>JVMArg2...] [<-c>CicscliArg1 <-c>CicscliArg2...]

Explanation: Information message

CTG6594I -adminport=<port_number> - TCP/IP port for local administration

Explanation: Information message

CTG6595I -tfile=<filename> - Trace file name

Explanation: Information message

CTG6596I -x - Enable full detailed trace

Explanation: Information message

CTG6597I The statistics API handler has not been started

Explanation: Information message

CTG6598I -tfilesize=<number> - Maximum trace file size in kilobytes

Explanation: Information message

CTG6599I -quiet - Disable reading/writing to/from console

Explanation: Information message

CTG6601I Request specific trace: *value*

Explanation: Information message

CTG6602I GatewayRequest type = *request*, flow version = *version*, flow type = *type*, Gateway return code = *RC*, length of data following the header = *length*

Explanation: Information message

CTG6603I >

Explanation: Information message

CTG6630E Channel name must not be null

Explanation: The channel name specified by the Java client application is null.

System action: The channel has not been created.

User response: Ensure that the channel name is not null.

CTG6631E Channel name is not between 1 and 16 characters in length

Explanation: The length of the channel name specified by the Java client application is not valid.

System action: The channel has not been created.

User response: Ensure that the channel name is between 1 and 16 characters in length.

CTG6632E The channel name contains non-valid characters

Explanation: The channel name specified by the Java client application contains characters that are not valid.

System action: The channel has not been created.

User response: Ensure that the channel name contains valid characters. Valid characters are uppercase characters (A-Z), lowercase characters (a-z), digits (0-9) and special characters < > \$ @ # / % & ? ! : | \ ' = " ; . - and _.

CTG6633E Container *name* is read-only

Explanation: The Java client application attempted to modify the contents of container *name* but the container has been marked as read-only by CICS.

System action: The contents and CCSid of the container have not been modified.

User response: Ensure that the Java client application does not attempt to modify the contents of read-only containers.

CTG6634E The type of container *name* is not supported

Explanation: The specified container is of a type that is not supported by the IBM CICS Transaction Gateway Java API.

System action: The container has not been added to the channel.

User response: Ensure that the container is created using the Channel.createContainer method.

CTG6635E Container *name* uses unsupported CCSid (*ccsid*)

Explanation: The specified container uses a CCSid that is not supported by the JVM.

System action: The contents of the container have not been modified.

User response: Ensure that the container uses a CCSid that is supported by the JVM.

CTG6636E Container *name* already exists in channel *name*

Explanation: The Java client application attempted to add or move a container to channel *name* but a container with the same name already exists in the channel. Container names must be unique within their containing channel.

System action: The container has not been added to the channel.

User response: Ensure that the name of the container does not match the name of an existing container in the same channel.

CTG6637E Container *name* is not of data type BIT

Explanation: The Java client application attempted to access or modify BIT data in the specified CHAR container.

System action: The contents of the container have not been modified.

User response: Ensure that the data type of the container matches the data type that the Java client application is attempting to access or modify.

CTG6638E Container *name* is not of data type CHAR

Explanation: The Java client application attempted to access or modify CHAR data in the specified BIT container.

System action: The contents of the container have not been modified.

User response: Ensure that the data type of the container matches the data type that the Java client application is attempting to access or modify.

CTG6639E Container name must not be null

Explanation: The container name specified by the Java client application is null.

System action: The container has not been created.

User response: Ensure that the container name is not null.

CTG6640E Container name is not between 1 and 16 characters in length

Explanation: The length of the container name specified by the Java client application is not valid.

System action: The container has not been created.

User response: Ensure that the container name is between 1 and 16 characters in length.

CTG6641E Container name contains non-valid characters

Explanation: The container name specified by the Java Client application contains characters that are not valid.

System action: The container has not been created.

User response: Ensure that the container name contains valid characters. Valid characters are uppercase characters (A-Z), lowercase characters (a-z), digits (0-9) and special characters < > \$ @ # / % & ? ! : | \ ' = " ; . - and _.

CTG6642E Container *name* not found in channel *name*

Explanation: The channel does not contain a container with the specified name.

System action: No system action.

User response: Ensure that the container exists in the channel.

CTG6650E Unable to connect to the Gateway daemon

Explanation: Java Client application unable to

connect to Gateway daemon.

System action: Java Client application is not started.

User response: If running the Gateway daemon, check whether the Java Client application has specified the correct parameters to connect to the Gateway daemon.

CTG6651E Unable to connect to the Gateway daemon: [address = *IP address*, port = *port*] [*error*]

Explanation: Where *IP address* specifies the Gateway daemon TCP/IP address, *port* specifies the TCP/IP port and *error* specifies the cause of the error.

System action: Java Client application is not started.

User response: If you are running the Gateway daemon, check to ensure that the Java Client application specifies the correct parameters when connecting to the Gateway daemon. If using the ctgadmin utility, verify that the command connects to the Gateway daemon port identified by the adminport parameter.

CTG6652E Unable to start the Gateway daemon listener

Explanation: Java Client application could not start the Gateway daemon listener.

System action: Java Client application is not started.

User response: Close this JavaGateway instance and check that Gateway daemon is correctly configured.

CTG6653E Unable to flow request to the Gateway daemon; this JavaGateway instance has been closed

Explanation: The Gateway daemon is unreachable.

System action: Close this JavaGateway instance.

User response: Check if the connections to the

Gateway daemon still exist.

CTG6654E An error occurred while the Gateway daemon was closing:
[*error*]

Explanation: Where *error* specifies the cause of the error.

System action: Close the Gateway daemon.

User response: Check error code.

CTG6655E There was an error reading the reply: [*error*]

Explanation: Where *error* specifies the cause of the error.

System action: Unable to read the reply.

User response: Check error code.

CTG6656E Incorrect data *field* received on the network connection between the Gateway daemon and client application

Explanation: A data flow between the Gateway and client application is corrupted. An internal field contains the wrong data.

System action: The connection is closed. The Gateway daemon continues to process requests on other network connections.

User response: Verify that only CICS TG applications connect to the Gateway daemon port. If this problem persists contact your service organization.

CTG6657E Invalid CICS Transaction Gateway address specified

Explanation: The Gateway daemon URL was incorrectly specified.

System action: Unable to connect to Gateway daemon at this address.

User response: Check Gateway daemon is running and address is right.

CTG6658E Local Gateway support has been terminated

Explanation: Attempts have been made to use the local mode support after the LocalJavaGateway.terminate method has been called.

System action: Unable to make further local mode requests.

User response: For information only.

CTG6659E In use Local Gateways currently exist

Explanation: The LocalJavaGateway.terminate method has been called while local mode applications are still in use.

System action: Wait for requests to finish.

User response: For information only.

CTG6660E Error copying reply: [*error*]

Explanation: Where *error* specifies the cause of the error.

System action: Unable to copy reply.

User response: Check error code.

CTG6661E Cannot change JavaGateway properties when the JavaGateway instance is open

Explanation: A call to a JavaGateway.set method has been made, after the JavaGateway has been opened.

System action: System action is not required.

User response: Close JavaGateway instance.

CTG6662E This JavaGateway instance is already open

Explanation: This JavaGateway instance is already open in the Java Client application.

System action: System action is not required.

User response: Close this JavaGateway instance.

CTG6663E Cannot open a JavaGateway instance when no protocol has been specified

Explanation: No protocol classes have been requested by the Java Client application.

System action: Unable to make further requests to this instance.

User response: Check that the Java Client application specifies a valid protocol parameter.

CTG6664E Protocol *protocol* not supported

Explanation: The JavaGateway URL specified an unsupported protocol.

System action: The JavaGateway object is not initialized.

User response: Ensure the Gateway URL is of the form 'protocol://host', where *protocol* is a supported protocol name. Refer to the Administration book for details of supported protocols.

CTG6665E This JavaGateway instance has been closed

Explanation: The JavaGateway instance was closed by the Java Client application.

System action: Unable to make a further request to this instance.

User response: Start new JavaGateway instance.

CTG6666E Unable to flow request to the Gateway daemon: [error]

Explanation: Where *error* specifies the cause of the error.

System action: Unable to make further requests to this instance.

User response: Check error code.

CTG6667E Error writing request: [error]

Explanation: Where *error* specifies the cause of the error.

System action: Unable to write further requests to this instance.

User response: Check error code.

CTG6668E Initial handshake flow failed: [error]

Explanation: Caused by *error*

System action: Unable to make initial handshake; hence unable to make requests.

User response: Check error code.

CTG6669E JavaGateway cannot open as must specify both client-side and server-side security classes

Explanation: Th server application needs security.

System action: Unable to make requests because of lack of security.

User response: Specify both client and server side security classes.

CTG6670E Exception occurred in the Gateway daemon: [error]

Explanation: Where *error* specifies the cause of the error.

System action: System action is not required.

User response: Check error code.

CTG6671E This JavaGateway instance has yet to be opened

Explanation: The Java Client application has not opened a new JavaGateway instance yet.

System action: System action is not required.

User response: Open JavaGateway instance before making requests.

CTG6672E One or more of the SSL protocol properties have not been defined

Explanation: The Java Client application must specify both the SSL_PROP_KEYRING_CLASS

and SSL_PROP_KEYRING_PW properties when attempting to establish an SSL connection to the CICS Transaction Gateway. The connection is not created.

System action: System action is not required.

User response: Change the application to specify both required properties and then retry.

CTG6673E SocketConnectTimeout cannot be less than 0

Explanation: An attempt has been made to set socket connection timeout with a negative value. The existing value has not been updated.

System action: System action is not required.

User response: Ensure the socket connect timeout is only either a positive value or zero.

CTG6674E Flow to the Gateway daemon was interrupted

Explanation: The Java thread was interrupted while flowing a request to the Gateway daemon.

System action: System action is not required.

User response: Inspect the Java Client application to identify where Thread.interrupt() might be invoked. If the problem persists, contact your service organization with a Java Client application trace.

CTG6686E Unable to initialize JNI library: [error]

Explanation: The ctgjni library could not be initialized. The associated error gives more information about the cause.

System action: The CICS Transaction Gateway fails to start.

User response: If the associated error indicates a problem locating the JNI library, ensure that it is available on the path shown by the error. If the associated error indicates a security exception ensure that the library has the correct access privileges. If the associated error indicates that the EXCI is unavailable, check the JNI log for further details. After the error has been resolved

start the CICS Transaction Gateway.

CTG6687E Key ring was tampered with, or password was incorrect

Explanation: Unable to import key ring.

System action: System action is not required.

User response: Ensure correct key ring and password has been provided.

CTG6699E Failed to open statistics connection: [error]

Explanation: *error* specifies the cause of the failure.

System action: Unable to make statistics requests.

User response: Check error code.

CTG6703E CICS Request: Load of native library DLL name failed with error

Explanation: Where *error* specifies the Java error.

System action: Unable to load native library.

User response: If running on z/OS.e, this message indicates that you have specified the local: protocol. Specify a remote protocol; the local: protocol is not supported on z/OS.e. On operating systems other than z/OS.e, check that the specified native library is available. Also check that the CICS TG application classes on the CLASSPATH are the same version as the native library specified.

CTG6704I BMS Map Converter

Explanation: Information message

CTG6705E key is not a valid AID key

Explanation: The exit key parameter indicated was not a valid AID value. Valid AID keys are enter, clear, PF1 - 24, and PA1 - 3.

System action: Unable to execute command.

User response: Check if AID key is valid.

CTG6706I Classes will be created in package *package*

Explanation: Information message

CTG6707I ScreenHandler beans will be generated

Explanation: Information message

CTG6708I Use this command to generate Map classes from BMS Map files

Explanation: Information message

CTG6709I Arguments: <-p packagename >
<-b> <-k exitAID> bmsfile1
bmsfile2 ...

Explanation: Information message

CTG6710I Processing file *filename* ...

Explanation: Information message

CTG6714W CICS Request: Cics_Rc return code not recognized

Explanation: CICS Server return code *return code* is not valid.

System action: System action is not required.

User response: Check if Cics_Rc is valid.

CTG6716E Unexpected error during initialization: [error]

Explanation: The request resulted in an unexpected Java error *error* .

System action: The request was rejected.

User response: If the problem persists contact your service organisation.

CTG6717E CICS Request: an attempt to convert from a C byte array to a Java String failed using the specified codepage (*code page*) for the conversion

Explanation: The codepage specified for the conversion was *code page* .

System action: A CICS Request attempted to convert between a C byte array and a Java String. The codepage being used for the conversion was *code page* .

User response: Check user codepage.

CTG6718E Unexpected codepage error: [error]

Explanation: An unexpected codepage error *error* occurred.

System action: The request was rejected.

User response: If the problem persists contact your service organisation.

CTG6719E CICS Request: Java default character encoding not obtained: [error]

Explanation: An attempt was made to obtain the Java default character encoding. *error* specifies the Java error.

System action: System action is not required.

User response: Check the return Java error.

CTG6731W ECIRrequest: This call type may return a reply intended for another Java client

Explanation: Either Call_Type ECI_GET_REPLY or Call_Type ECI_GET_REPLY_WAIT have been used. These types of call should be used with caution since they will return the next reply, which may be intended for another Java client.

System action: System action is not required.

User response: Use ECI_GET_REPLY and ECI_GET_REPLY_WAIT call types with caution.

CTG6733W **ECIRequest: The Commarea
outbound length is greater than
Commarea_Length**

Explanation: The amount of data sent from the Java client to the CICS Transaction Gateway, is greater than the size of the Commarea given to CICS.

System action: System action is not required.

User response: Specify larger Commarea.

CTG6734W **ECIRequest: The Commarea
inbound length is greater than
Commarea_Length**

Explanation: The amount of data sent to the Java client from the CICS Transaction Gateway, is greater than the size of the Commarea given to CICS.

System action: System action is not required.

User response: Specify larger Commarea.

CTG6736E **ECIRequest: Message Qualifier is
invalid, outside of range -32767 to
32767**

Explanation: An invalid message qualifier parameter has been provided.

System action: None.

User response: Correct the value passed for message qualifier in the method invocation.

CTG6737I **XA transaction support is enabled**

Explanation: Information message

CTG6738I **XA transaction support is not
enabled**

Explanation: Information message

CTG6740E **ECIRequest: Commarea outbound
length is larger than the
maximum Commarea length**

Explanation: Commarea outbound length can

never be greater than 32768

System action: System action is not required.

User response: Check if Commarea outbound length is not greater than 32768

CTG6749E **ESIRequest: Userid, Password or
Server Name too long**

Explanation: Either the Userid, Password or Server Name entered is longer than required.

System action: System action is not required.

User response: Re-enter the correct Userid, Password or Server name.

CTG6755E **EPIRequest: size field limited to
data.length**

Explanation: The size field may not exceed the length of the data array.

System action: System action is not required.

User response: Specify larger data array.

CTG6762W **termIndex: 0xFFFF is invalid for
calls that are not to a local
Gateway**

Explanation: Unless using a local Gateway, it is invalid to attempt to get an event without specifying a termIndex, since you may obtain events intended for other terminal resources.

System action: System action is not required.

User response: Specify a termIndex.

CTG6764E **The Gateway daemon does not
support running in 64-bit mode**

Explanation: The Gateway daemon was invoked under a 64-bit JVM.

System action: This message is logged and the Gateway daemon fails to start.

User response: Configure the Gateway daemon to use a non-64-bit JVM and restart the Gateway daemon. The JVM used by the Gateway daemon is configured using the ctgjava command on

Windows platforms, or using the PATH environment variable on other platforms. Ensure the correct level of Java is available on your system and the location of Java is configured as appropriate.

CTG6765E **The Gateway daemon is unable to find the CICS TG JNI native library DLL *name***

Explanation: The Gateway daemon Java classes are unable to load the CICS TG JNI native library DLL.

System action: This message is logged. The Gateway daemon fails to start.

User response: This might be because of a corrupted installation. Reinstall the CICS Transaction Gateway. If the problem persists contact your service organization.

CTG6770E **ECIRequest: Invalid Call_Type**

Explanation: Invalid call type passed to ECIRequest

System action: System action is not required.

User response: Check call type.

CTG6771E **ECIRequest: Commarea outbound length is negative**

Explanation: Commarea outbound length must always be a positive integer.

System action: Could not create a Commarea outbound length of this size; default length is set.

User response: Check if Commarea outbound length is a positive integer.

CTG6772E **ECIRequest: eci_timeout value is negative**

Explanation: eci_timeout can only be 0 or a positive integer.

System action: Could not set ECI timeout; default timeout is set.

User response: Check if eci_time is a positive integer.

CTG6773E **ECIRequest: COMMAREA inbound length is negative**

Explanation: The Java client application attempted to set the inbound COMMAREA length to a value less than zero.

System action: The inbound COMMAREA length is not modified.

User response: Ensure that the Java client application does not attempt to set a negative inbound COMMAREA length.

CTG6774E **No JavaGateway or EPIGateway instance has been associated with this terminal**

Explanation: An attempt to connect a terminal failed as no JavaGateway or EPIGateway instance has been associated with that terminal instance.

System action: None.

User response: Create and Configure a JavaGateway or EPIGateway instance and use the appropriate methods or constructors to associate this instance with the instance of the terminal.

CTG6775E **Your created terminal cannot use these extended connect methods, as it is not an extended terminal**

Explanation: An attempt to use the extended Connect Methods on a terminal has failed. The Extended Connect methods can only be used on extended terminal instances.

System action: None.

User response: Use the standard connect method or create an extended terminal instance if you wish to use the extended connect methods.

CTG6776E CicsCpRequest: unable to convert the CICS code page to a Java encoding: [error]

Explanation: An attempt to convert a CICS code page to a Java format, (for example, 850 to Cp850), failed. *error* is the Java error.

System action: System action is not required.

User response: Check the Java error and the CICS Code page.

CTG6777E Security Exception: return code RC from *method* call

Explanation: Return code *RC* was received when attempting to invoke a security type method *method*

System action: None.

User response: Look up the return code to determine the cause of failure.

CTG6779E Cannot find equivalent CCSid for encoding

Explanation: A Java encoding was specified for a terminal, but no equivalent host CCSid is known for this encoding.

System action: None.

User response: Check the documentation for the list of allowed Java encodings that have equivalent CCSid values.

CTG6780E EPIRequest: Invalid Call_Type

Explanation: Invalid call type passed to EPIRequest.

System action: System action is not required.

User response: Check call type.

CTG6781E EPIRequest: Invalid event

Explanation: Invalid event passed.

System action: System action is not required.

User response: Check event.

CTG6782E EPIRequest: Invalid termIndex: 0xFFFF

Explanation: It is invalid to attempt to get an event without specifying a termIndex, since you may obtain events intended for other terminal resources.

System action: System action is not required.

User response: Specify a termIndex.

CTG6783E EPIRequest: termIndex *index* does not exist

Explanation: Where *index* is the invalid termIndex.

System action: System action is not required.

User response: Check that termIndex is available and unique to this connection.

CTG6784E EPIRequest: termIndex *index* belongs to another connection

Explanation: Where *index* is the invalid termIndex.

System action: System action is not required.

User response: Check that termIndex is unique to this connection.

CTG6785E EPIRequest: EPI Initialization failed: [error]

Explanation: Where *error* is the error.

System action: Exit application.

User response: For information only.

CTG6786I There are no free terminals available

Explanation: Information message

CTG6787E Exception [exception] occurred while attempting to load classes from the path: *path*

Explanation: An exception occurred while

screen handler beans where being loaded from the path specified. Possible causes are: the class path is incorrect, a screen handler bean did not have a public constructor, a file or directory could not be accessed.

System action: System action is not required.

User response: Check if class path exist and classes are available.

CTG6788E Exception [exception] occurred while attempting to load the class: class name

Explanation: An exception occurred while the specified class was being loaded. Possible causes are: the class could not be found, the class did not have a public constructor.

System action: System action is not required.

User response: Check exception and that class is available and has a public constructor.

CTG6789E An exception has occurred: exception

Explanation: Where *exception* is the exception.

System action: System action is not required.

User response: Check where exception has occurred.

CTG6790E Too much data received from CICS

Explanation: More data was returned by CICS than could be handled.

System action: The error is returned to the application.

User response: If the problem persists contact your service organization.

CTG6792E Null transaction ID parameter passed to Terminal.send

Explanation: If a transaction ID parameter is passed to send, it must be a valid string.

System action: System action is not required.

User response: Check if a valid transaction ID is given.

CTG6793E The terminal is in the wrong state for the requested action

Explanation: An action was requested when the terminal was in the wrong state, for example: disconnect while the terminal was not idle.

System action: Wait and try again after a fixed period of time.

User response: Wait and try again after a period of time.

CTG6794E Cursor could not be set to row row number, column column number - out of range

Explanation: Either the row *row number* or the column *column number* are off the screen.

System action: System action is not required.

User response: Check if either the row or the column are out of range.

CTG6795E Unsupported datastream received from CICS - command is command

Explanation: The datastream received from CICS could not be parsed.

System action: System action is not required.

User response: For information only.

CTG6796E Transaction transaction failed, return code RC

Explanation: Transaction *transaction* failed to start, did not complete, or timed out.

System action: None.

User response: Check the Read Timeout value for the terminal if the transaction timed out. Check the Client and CICS server logs for more information on why the transaction failed.

CTG6797E Failed to delete terminal

CTG6798E • CTG6983E

Explanation: Terminal could not be deleted.

System action: Wait, then try again.

User response: Check if terminal is still in use.

CTG6798E Unknown attribute type *type*

Explanation: The attribute type *type* is not supported.

System action: System action is not required.

User response: Check attribute type.

CTG6799E Return code *RC* from *method name* call

Explanation: The EPI call *method name* returned the return code *RC* .

System action: System action is not required.

User response: Check return code; if it is not 0 than check for error.

CCL6906I Attach to server statistics shared memory complete, return code *INSERT-0*

Explanation: Information message

CCL6911E Statistic block storage allocation request failed, server statistics unavailable for collection

Explanation: An internal error occurred. The Client was unable to allocate memory to allow server statistics to be gathered.

System action: The message is written to the error log and the Client tries to continue, although attempts to reference this shared memory element may result in a trap. Server statistics will not be available for collection.

User response: Ensure that all orphaned shared memory is purged from the system and that there are sufficient memory resources. The way that you do this depends on your operating system although most UNIX and Linux systems provide the `ipcs` and `ipcrm` utilities. If the

problem persists, contact your service organization.

CCL6931W Storage limit reached for statistic *Stat ID* .

Explanation: Statistics that provide a list of items have predefined storage limits. The named statistic has reached the limit of storage allocated for it.

System action: The message is written to the error log and processing continues. No more entries are added to this statistic

User response: The named statistic lists values recorded before the storage limit is reached.

CTG6981I Successfully initialized JNI library

Explanation: Information message

CTG6982E The Gateway daemon version *version* is unable to load JNI DLL *name* for a different version *version*

Explanation: The version of the CICS TG JNI native library is different from the version of the Gateway daemon java classes.

System action: This message is logged. The Gateway daemon fails to start.

User response: This might be because of a corrupted installation. Reinstall the CICS Transaction Gateway. If the problem persists contact your service organization.

CTG6983E Initialization failed because the CICS TG Java client application classes version *version* attempted to load JNI DLL *name* version *version*

Explanation: The CICS TG Java classes are from a different release, compared to the CICS TG JNI DLL.

System action: This error is thrown to the Java client application. Subsequent calls to the CICS TG API will fail.

User response: Ensure the version of the CICS TG JNI referred to in the Java library path is the same as the CICS TG jar files in the application's CLASSPATH. The CICS TG application trace can be used to display the java library path setting used.

CTG6984E Initialization failed because the CICS TG Java client application classes are unable to find the CICS TG JNI native library file *name* in the specified library path

Explanation: A Java client application is unable to find the CICS TG JNI native library.

System action: This error is thrown to the Java client application. Subsequent CICS TG API calls will fail.

User response: Check that the CICS TG bin directory is in the library path. The CICS TG application trace can be used to display the java library path setting used.

CTG6999E Unable to open configuration file *filename* : [*reason*]

Explanation: The Gateway daemon was unable to open configuration file *filename* due to *reason* .

System action: The Gateway daemon is not started.

User response: Ensure that the specified configuration file exists and that the file permissions are sufficient for the Gateway daemon to read it.

CCL7001I INSERT-0 - CICS 3270 Terminal Emulator

Explanation: Information message

CCL7002I INSERT-0 - CICS 3270 Printer Emulator

Explanation: Information message

CCL7003I Command options are:

Explanation: Information message

CCL7004I -S[=server] - Select a server [or name a server]

Explanation: Information message

CCL7005I -R[=server] - Identical to -S

Explanation: Information message

CCL7006I -T=transid - Specify an initial transaction

Explanation: Information message

CCL7007I -K=keyfile - Select a keyboard binding file

Explanation: Information message

CCL7008I -C=colorfile - Select a color binding file

Explanation: Information message

CCL7009I -M=modelname - Select a model terminal on the server

Explanation: Information message

CCL7010I -N=netname - Select a named terminal on the server

Explanation: Information message

CCL7011I -P=printcmd - Set a command to handle print requests

Explanation: Information message

CCL7012I -F=printfile - Append print requests to a print file

Explanation: Information message

CCL7013I -W - Wait for confirmation before completing

Explanation: Information message

CCL7014I **-Q - Inhibit all output messages**

Explanation: Information message

CCL7015I **-L=modelname - Select a model terminal for printing**

Explanation: Information message

CCL7016I **-O=netname - Select a terminal for printing**

Explanation: Information message

CCL7017I **-U - Print null lines**

Explanation: Information message

CCL7019I **-J - Issue 1 print job per transaction**

Explanation: Information message

CCL7020I **-A - Request a signon incapable terminal**

Explanation: Information message

CCL7021I **-Z - Do not insert spaces between SBCS and DBCS data**

Explanation: Information message

CCL7022I **-B - Print blank lines**

Explanation: Information message

CCL7023I **-E - Use extended default colors**

Explanation: Information message

CCL7025I **Press Clear to continue**

Explanation: Information message

CCL7026I **Press Clear to terminate the emulator**

Explanation: Information message

CCL7027I **The emulator will terminate**

Explanation: Information message

CCL7028E **Cannot open binding file *file name***

Explanation: CICSTERM could not find or open the specified color, or keyboard, binding file.

System action: The emulator terminates.

User response: Ensure the required binding files exist and are correctly specified, then restart the emulator.

CCL7029E **Out of memory processing binding file *file name***

Explanation: CICSTERM ran out of memory while processing a color, or keyboard, binding file.

System action: The emulator terminates.

User response: Reduce the number of bindings specified in the file or try to provide more system free memory; then restart the emulator.

CCL7030E **Error detected at line *number* in binding file *file name* - duplicate bind for item *item***

Explanation: CICSTERM detected an error in a color, or keyboard, binding file. Only one binding is allowed for the specified item.

System action: The emulator terminates.

User response: Correct the binding file and restart the emulator.

CCL7031E **Error detected at line *number* in binding file *file name* - unknown command *command***

Explanation: CICSTERM detected an error in a color, or keyboard, binding file. The specified command was not a valid binding command.

System action: The emulator terminates.

User response: Correct the binding file and restart the emulator.

CCL7032E Error detected at line *number* in binding file *file name* - invalid function *function*

Explanation: CICSTERM detected an error in a keyboard binding file. The specified item was not a valid 3270 keyboard function.

System action: The emulator terminates.

User response: Correct the binding file and restart the emulator.

CCL7033E Error detected at line *number* in binding file *file name* - invalid modifier *modifier*

Explanation: CICSTERM detected an error in a keyboard binding file. The specified item was not a valid workstation keyboard modifier value.

System action: The emulator terminates.

User response: Correct the binding file and restart the emulator.

CCL7034E Error detected at line *number* in binding file *file name* - invalid key *key*

Explanation: CICSTERM detected an error in a keyboard binding file. The specified item was not a valid workstation key name.

System action: The emulator terminates.

User response: Correct the binding file and restart the emulator.

CCL7035E Error detected at line *number* in binding file *file name* - invalid attribute *attribute*

Explanation: CICSTERM detected an error in a color binding file. The specified item was not a valid 3270 screen field attribute.

System action: The emulator terminates.

User response: Correct the binding file and restart the emulator.

CCL7036E Error detected at line *number* in binding file *file name* - invalid color *color*

Explanation: CICSTERM detected an error in a color binding file. The specified item was not a valid workstation display color.

System action: The emulator terminates.

User response: Correct the binding file and restart the emulator.

CCL7037E Unable to communicate with the Client daemon

Explanation: CICSTERM or CICSPRNT could not communicate with the Client daemon. This probably means the Client daemon was unable to start.

System action: The emulator terminates.

User response: Try to start the Client daemon using the CICSCLI /S command.

CCL7038E The Client daemon has not been started

Explanation: The Client daemon is not currently running.

System action: The emulator terminates.

User response: Refer to the information center for details of how to start the product on the platform you are using.

CCL7039I *emulator* - CICS server selection

Explanation: Information message

CCL7040I Please choose the CICS server you require:

Explanation: Information message

CCL7041I Enter the number of the server or Esc to cancel

Explanation: Information message

CCL7042I *INSERT-0 : INSERT-1 - INSERT-2*

Explanation: Information message

CCL7043E **Terminal screen width is invalid - reset to 80 columns**

Explanation: The server returned a terminal definition that contained an invalid screen width for the terminal emulator. (Screen widths are normally 40, 80, or 132 columns.)

System action: The emulator resets the screen width to a standard 80 columns. Output may appear corrupted because the client and server terminal definitions may not match.

User response: Check with the server system administrator to determine why the terminal definition is incorrect.

CCL7044E **Terminal screen size is too large - reset to maximum**

Explanation: The server returned a terminal definition that contained a screen size too large for the terminal emulator. The product of screen width and screen depth cannot exceed 4096.

System action: The emulator resets the screen depth to the maximum number of rows. Output may appear corrupted because the client and server terminal definitions may not match.

User response: Check with the server system administrator to determine why the terminal definition is incorrect.

CCL7045E **Connection lost with server *server***

Explanation: The connection with the specified server has been lost. This would normally indicate that the server is no longer available.

System action: The emulator waits until the server is available again and tries to reconnect.

User response: Wait for the connection to be reestablished or clear the error message and terminate the emulator.

CCL7046E **The Client daemon reported a****resource shortage**

Explanation: The Client daemon returned a resource shortage error. This indicates either a shortage of free memory, or if SNA communications are being used, there may be insufficient LU6.2 sessions available.

System action: The emulator attempts to continue, although further requests to the Client daemon may receive the same error response.

User response: If the Client is very busy, the resource shortage may disappear when the work load decreases. Otherwise, examine the Client log to determine the cause of the error, stop the Client and restart it with more resources.

CCL7047E **The Client daemon is closing down**

Explanation: CICSTERM or CICSPRNT could not be started because the Client daemon is closing down.

System action: The emulator terminates.

User response: Wait for the Client daemon to close down, then restart the emulator.

CCL7048E **Server *server* is undefined**

Explanation: The Client daemon indicated that the specified server name is unknown.

System action: The emulator terminates.

User response: Ensure the required server name is one of those listed in the configuration file, or use the /S option to select from a list of valid names.

CCL7049E **Unable to connect to server *server***

Explanation: The Client daemon was unable to connect to the specified server.

System action: The emulator terminates.

User response: Check for other messages and examine the Client error log to determine the cause of the error.

CCL7050E The MaxBufferSize setting is too small for this terminal

Explanation: The emulator tried to send more data than the Client daemon could accept.

System action: The data is ignored and the emulator continues.

User response: Increase the MaxBufferSize value in the Client section of the configuration file, then stop and restart the product.

CCL7051E Server *server* does not support client terminals

Explanation: Client 3270 emulator terminals are not supported by all CICS servers. The specified server is one of those that does not provide this support. If this is a CICS for iSeries server this message can also be generated if CICS is unavailable.

System action: The emulator terminates.

User response: Select a server that does support client terminals or check with the server system administrator to determine why client terminals are not supported.

CCL7052E Server *server* rejected this terminal install with a security error

Explanation: The specified server is secure and required a valid userid and password before it could install a client terminal.

System action: The emulator terminates.

User response: Enter a valid userid and password when prompted by the Client daemon. Alternatively, to avoid the prompt use the CICSCLI /C command to set the userid and password.

CCL7053E Errors found while communicating with server *server*

Explanation: The Client daemon encountered errors while communicating with the specified server.

System action: The emulator ignores the errors and tries to continue.

User response: Check for other messages and examine the Client error log to determine the cause of the error.

CCL7054I Server *server* is currently unavailable

Explanation: Information message

CCL7055E Terminal install failed - NetName *name* is invalid

Explanation: The server rejected the terminal because the specified NetName was invalid.

System action: The emulator terminates.

User response: Ensure the requested terminal NetName is correct and defined at the server. If the error persists, check with the server system administrator for the cause of the error.

CCL7056E Terminal install failed - Model *model* is invalid

Explanation: The server rejected the terminal because the specified Model was invalid.

System action: The emulator terminates.

User response: Ensure the requested terminal Model is correct and defined at the server. If the error persists, check with the server system administrator for the cause of the error.

CCL7057E Terminal install failed - NetName *name* is already installed

Explanation: The server rejected the request because another terminal was already installed with the same NetName.

System action: The emulator terminates.

User response: Ensure the requested terminal NetName is correct. If the error persists, check with the server system administrator for the cause of the error.

CCL7058E Terminal install failed - rejected by the server

Explanation: The server was unable to accept the terminal and rejected the installation request.

System action: The emulator terminates.

User response: Check with the server system administrator to determine why the terminal was rejected.

CCL7059E Terminal install failed - reason unknown

Explanation: The server rejected the terminal.

System action: The emulator terminates.

User response: Check with the server system administrator to determine why the terminal was rejected.

CCL7060W DBCS datastream errors detected

Explanation: The terminal emulator detected errors in the DBCS datastream.

System action: The terminal emulator ignores the errors and tries to continue.

User response: Check with the server system administrator to determine why incorrect data is being generated. Client trace information may be helpful when trying to find these errors.

CCL7061E Unable to open or create print output file

Explanation: The terminal emulator was unable to open or create the local print file specified by the PrintFile option in the configuration file.

System action: The print request is ignored and the terminal emulator continues.

User response: Ensure the name of the print file you specified in the configuration file is correct and accessible.

CCL7062E Unable to write to the print output file

Explanation: The terminal was unable to write to the local print file specified by the PrintFile option in the configuration file.

System action: The print request is ignored and the emulator continues.

User response: Ensure the name of the print file you specified in the configuration file is correct and accessible.

CCL7063E Unable to execute print command (errno = *errno*)

Explanation: The terminal emulator is unable to execute the local print command specified by the PrintCommand option in the configuration file. The *errno* value indicates the return code from the operating system.

System action: The print request is ignored and the emulator continues.

User response: Ensure the print command you specified in the configuration file is correct. A fully qualified filename should be used.

CCL7064I CICS printer is active - netname %s1 terminal id %s2

Explanation: Information message

CCL7065I Press Esc to terminate

Explanation: Information message

CCL7066I Continue with CICS printer closedown?

Explanation: Information message

CCL7067I CICS printer closedown cancelled - processing continues

Explanation: Information message

CCL7068I Press Enter to continue or Esc to cancel printer closedown

Explanation: Information message

CCL7069E Server *server* has reported a transaction ABEND

Explanation: The specified server reported an ABEND while trying to run the current transaction. This may have been caused by a communications failure between the Client daemon and server.

System action: The terminal emulator ignores the errors and tries to continue.

User response: Examine any other messages and the Client error log and server error log to determine the cause of the error.

CCL7070E Terminal screen depth is invalid - reset to 24 rows

Explanation: The server returned a terminal definition that contained an invalid screen depth for the terminal emulator. (Screen depths are normally between 10 and 60 rows).

System action: The terminal emulator has reset the screen depth to a standard 24 rows. Output may appear corrupted because the terminal emulator and server terminal definitions may not match.

User response: Check with the server system administrator to determine why the terminal definition is incorrect.

CCL7071E Screen data exceeds the display space available

Explanation: The server returned more data than the terminal emulator can accept.

System action: The excess data is ignored and the terminal emulator tries to continue.

User response: Check with the server system administrator to determine why such a large amount of data is being generated. Client trace information may be helpful when trying to find these errors.

CCL7072E The Client daemon has reached its MaxServers limit

Explanation: The Client daemon can only

communicate simultaneously with the number of servers specified by the MaxServers option in the Client section of the configuration file (ctg.ini). This limit has been reached.

System action: The emulator terminates.

User response: Use the CICSCLI /X command to close connections to servers that are no longer required, or increase the "MaxServers" setting in the configuration file; then stop and restart the product.

CCL7073E Terminal install failed - requested terminal is not a 3270 device

Explanation: The server rejected the terminal because the server terminal definition does not represent a 3270 device.

System action: The emulator terminates.

User response: Ensure the requested terminal Model or NetName is correct and correctly defined at the server. If the error persists, check with the server system administrator for the cause of the error.

CCL7074E Terminal install failed - server is busy

Explanation: The server rejected the terminal emulator because the server was too busy.

System action: The emulator terminates.

User response: Retry the request sometime later when the server is not so busy.

CCL7080I Continue with emulator closedown?

Explanation: Information message

CCL7081I &File

Explanation: Information message

CCL7082I &Print

Explanation: Information message

CCL7083I E&xit

Explanation: Information message

CCL7084I &Settings

Explanation: Information message

CCL7085I &Font

Explanation: Information message

CCL7086I &Autosize

Explanation: Information message

CCL7087I &Save on exit

Explanation: Information message

CCL7098E **An internal processing error has occurred**

Explanation: The terminal emulator detected an unexpected internal error.

System action: Error information is written to the Client error log.

User response: Obtain the Client error log - and, if possible, trace information - and refer the problem to your support organization.

CCL7099I **Refer to the error log for more details**

Explanation: Information message

CCL7101E **Internal transport error (Function = *Function*, Error Code = *Error*)**

Explanation: An internal emulator function failed.

System action: The message is written to the error log and the emulator terminates. The function name and error code are logged.

User response: If the problem persists, contact your service organization.

CCL7103E **Internal Windows function error (Function = *Function*)**

Explanation: An internal Windows function failed.

System action: The message is written to the error log and the emulator terminates. The function name is logged.

User response: If the problem persists, contact your service organization.

CCL7104E **Emulator has insufficient memory to create an internal control structure**

Explanation: The emulator is unable to allocate enough memory to create an internal control structure.

System action: The message is written to the error log and the emulator terminates.

User response: Try altering the system configuration to provide more free memory when the emulator is started, then restart the emulator.

CCL7105E **Emulator has received incorrect sized data block(s) (*size1*, *size2*)**

Explanation: The emulator has received incorrect data from the Client daemon.

System action: The message is written to the error log and the emulator continues.

User response: If the problem persists, contact your service organization.

CCL7106E **Emulator ATI table error**

Explanation: The emulator has received too many outstanding ATI requests from the server.

System action: The message is written to the error log and the emulator continues.

User response: Try to establish why the emulator is receiving large numbers of ATI requests while being unable to initiate them. If the problem persists contact your service organization.

CCL7108E No CICS servers are available

Explanation: There no CICS servers available for the emulator to use.

System action: The emulator terminates.

User response: Check that there are one or more valid CICS servers in the configuration file. If the workload manager is active, check that at least one of the servers the emulator tried to connect to is available. Examine the Client trace to see which servers the workload manager attempted to connect to.

CCL7109E CICS_EpiAddTerminal rejected by the EPI user exit

Explanation: The EPI user exit (CICSEPIX) returned CICS_EXIT_DONT_ADD_TERMINAL and rejected the CICS_EpiAddTerminal request.

System action: The EPI terminal is not added.

User response: Check with the server system administrator to determine why the terminal was rejected.

CCL7136E A minimum screen size of *number* rows by *number* columns is required

Explanation: The screen size available is too small for cicsterm. The minimum size is 25 rows by 80 columns.

System action: The terminal emulator has not been started.

User response: Change the screen size of your terminal before starting cicsterm and retry.

CCL7252W CICS EPI user exits are not activated

Explanation: One or more required exits have not been initialized.

System action: The Client daemon will continue but exits will not be used.

User response: Define the exits as defined in the "Writing your own user exits section" of the

CICS Transaction Gateway for Multiplatforms Information Center.

CCL7257E Application with PID = *PID* cannot determine whether Workload Manager is active

Explanation: The application *PID* cannot determine whether Workload Manager is active.

System action: This message is written to the log, and Workload Manager is switched off in application *PID* .

User response: If the problem persists, contact your service organization.

CCL7273E Error accessing application message queue. Error code=*error code* . The application with process Id *PID* must be restarted.

Explanation: The Client daemon cannot communicate with the Client application's IPC message queue. This is an unrecoverable situation. The Client application with process Id *PID* must be restarted.

System action: No action.

User response: Restart the Client application.

CCL7628E All affinity regions for program are invalid

Explanation: A user application specified a program which has no affinity with any of the active regions defined to the Workload Manager.

System action: It is possible to define program affinity to CICS regions on the Workload Manager, to indicate that a program is defined on a particular CICS region. If using the biasing algorithm, the Workload Manager sends requests only to regions with the correct affinity.

User response: A trace point prior to this one gives the name of the program. Check that the program has affinity with one of the regions defined to the Workload Manager, and that at least one of the CICS regions with affinity is running. The 'cicscli -l' command will indicate if the connection to the CICS server is active.

CCL7629E Unable to read Workload Manager INI file *filename* (reason = error)

Explanation: The Workload Manager is unable to read the specified file, which contains working configuration data. The file is created and maintained by the Workload Manager.

System action: The message is written to the log and the Load Manager is disabled.

User response: Make sure that the INI file being read exists on the disk, and that it is not read-protected. If the problem persists, contact your service organization.

CCL7639W Duplicate parameter *parameter* found [line *number*] - ignored

Explanation: This is a warning. A duplicate parameter heading has been detected in the Workload Manager section of the configuration file ctg.ini.

System action: The Client continues to read the configuration file ctg.ini, but all duplicate parameters are ignored.

User response: Remove any duplicate parameters from the configuration file ctg.ini.

CCL7640W Duplicate section *section* found [line *number*] - ignored

Explanation: This is a warning. A duplicate section heading has been detected in the Workload Manager section of the configuration file ctg.ini.

System action: The Client continues to read the configuration file ctg.ini, but all duplicate sections are ignored.

User response: Remove any duplicate sections from the configuration file ctg.ini.

CCL7643E Incomplete or missing section *section*

Explanation: The Workload Manager found an invalid section in the configuration file ctg.ini, the specified section could not be read.

System action: The Client continues but the Workload Manager is disabled.

User response: Remove or correct the invalid section in the configuration file ctg.ini.

CCL7644E Cannot find parameter *parameter*

Explanation: A mandatory parameter could not be found in the Workload Manager INI file.

System action: The Workload Manager will be disabled.

User response: Make sure that the Workload Manager INI file is valid.

CCL7645E Parameter *parameter* has invalid value *value*

Explanation: The specified parameter has an invalid value.

System action: The Workload Manager will be disabled.

User response: Make sure that the Workload Manager INI file is valid.

CCL7646E Value for parameter *parameter* is too large (Max = *size*)

Explanation: The value of the specified parameter is too large.

System action: The Workload Manager will be disabled.

User response: Make sure that the value specified is within the required range.

CCL8001I *INSERT-0* - CICS Transaction Gateway Client Daemon Control Program

Explanation: Information message

CCL8002I Command options are:

Explanation: Information message

CCL8003I -S[=server] - Start Client daemon

[and connect to a server]

Explanation: Information message

CCL8004I -X[=server] - Shutdown Client daemon [or server connection]

Explanation: Information message

CCL8005I -I[=server] - Abort Client daemon [or server connection]

Explanation: Information message

CCL8006I -L - List active server connections

Explanation: Information message

CCL8007I -D[=size] - Enable service tracing [and set size limit]

Explanation: Information message

CCL8008I -O - Disable all service tracing

Explanation: Information message

CCL8009I -C=server - Specify server for security changes

Explanation: Information message

CCL8010I Requires one or both of -U and -P options

Explanation: Information message

CCL8011I -U[=userid] - Set userid to be used with a server

Explanation: Information message

CCL8012I -P[=password] - Set password to be used with a server

Explanation: Information message

CCL8015I -W - Wait for confirmation before completing

Explanation: Information message

CCL8016I -Q - Inhibit all output messages

Explanation: Information message

CCL8017I -V - Display Client daemon version information

Explanation: Information message

CCL8018I -Y - Controlled restart of Client daemon

Explanation: Information message

CCL8019I -J - Immediate restart of Client daemon

Explanation: Information message

CCL8020I -M[=components] - Specify Client components to trace

Explanation: Information message

CCL8021I -B - Specify memory mapped tracing

Explanation: Information message

CCL8023I *INSERT-0* performed no action

Explanation: Information message

CCL8024E Unable to communicate with the Client daemon

Explanation: CICSCCLI could not communicate with the Client daemon process. This probably means that the Client daemon was unable to start.

System action: CICSCCLI performs no action.

User response: Try starting the Client daemon using the CICSCCLI /S command.

CCL8025I The Client daemon has not been started

Explanation: Information message

CCL8026I Client trace is enabled

Explanation: Information message

CCL8027I Client trace is disabled

Explanation: Information message

**CCL8029I *INSERT-0* for *INSERT-1* Version
*INSERT-2***

Explanation: Information message

**CCL8030E Unable to change Client trace
state**

Explanation: The Client daemon was unable to set the requested trace state.

System action: Service tracing remains unchanged.

User response: Examine any other messages and the Client error log to determine the cause of the error.

CCL8031E Server *server* is unknown

Explanation: The specified server name was not defined in the configuration file.

System action: CICSCLI performs no action.

User response: Correct the server name and reissue the command. If required, add the server to the configuration file, then stop and restart the product to activate the changes.

CCL8032E Unable to complete stop request

Explanation: The Client daemon was unable to disconnect from the server.

System action: CICSCLI performs no action.

User response: Try using CICSCLI /I; if this fails examine any other messages and the Client error log to determine the cause of the error.

**CCL8033I Connection to server *server* is
stopping**

Explanation: Information message

CCL8034I The Client daemon is stopping

Explanation: Information message

**CCL8035I The Client daemon is already
started**

Explanation: Information message

**CCL8036E Unable to perform start request,
see Client error log**

Explanation: The Client daemon was unable to complete the start request.

System action: CICSCLI performs no action.

User response: Examine any other messages and the Client error log to determine the cause of the error.

**CCL8037I Connection to server *server* is
being started**

Explanation: Information message

CCL8038I The Client daemon is starting

Explanation: Information message

CCL8039I The Client daemon is stopping

Explanation: Information message

**CCL8040I No servers are being used by the
Client daemon**

Explanation: Information message

**CCL8041I The Client daemon is using the
following servers:**

Explanation: Information message

CCL8042I Server *server* (using *protocol* to

netname) is available

Explanation: Information message

CCL8043I Server *server* (using protocol to *netname*) is unavailable

Explanation: Information message

CCL8044I Server *server* (using protocol to *netname*) is connecting

Explanation: Information message

CCL8045I Server *server* (using protocol to *netname*) is stopping

Explanation: Information message

CCL8046E Unable to perform list request

Explanation: The Client daemon was unable to perform a request to list the current server connection details.

System action: CICSCLI performs no action.

User response: Examine any other messages and the Client error log to determine the cause of the error.

CCL8047I -T - Display Client daemon startup parameters

Explanation: Information message

CCL8050I Requested security information has been updated

Explanation: Information message

CCL8051E Invalid request to change security information, server name is required

Explanation: A CICSCLI command was issued with the /U or /P options but without the /C or /S option specifying a server name.

System action: CICSCLI performs no action.

User response: Reissue the command with the correct options.

CCL8052E Unable to make the requested security information changes

Explanation: The Client daemon was unable to perform the request to change the userid and password information for a secure server.

System action: CICSCLI performs no action.

User response: Examine any other messages and the Client error log to determine the cause of the error.

CCL8053E Connection to server *server* is already started

Explanation: The Client daemon was unable to perform a request to connect to the named server, because a connection with the server already exists.

System action: CICSCLI performs no action.

User response: If the connection is no longer required, use the CICSCLI /X command to close it.

CCL8054E Connection to server *server* has not been started

Explanation: The Client daemon was unable to perform the request to start a connection with the specified server.

System action: CICSCLI performs no action.

User response: Examine any other messages and the Client error log to determine the cause of the error.

CCL8055E The Client daemon has reached its MaxServers limit

Explanation: The Client daemon can only communicate simultaneously with the number of servers specified by the MaxServers option in the Client section of the configuration file. This limit has been reached.

System action: CICSCLI performs no action.

User response: Use the CICSCLI /X command to close connections to servers that are no longer required. Alternatively, increase the MaxServers setting in the configuration file; then stop and restart the product.

CCL8056E *Size is an incorrect trace size limit value*

Explanation: The CICSCLI /D command to start Client tracing can include an optional numeric value specifying the maximum size of data to be traced. The specified size is either not a numeric value or is outside the permitted range of 1 - 32767.

System action: CICSCLI performs no action.

User response: Reissue the command with the correct value.

CCL8057I *Client daemon error and security pop-ups have been enabled*

Explanation: Information message

CCL8058I *Client daemon error and security pop-ups have been disabled*

Explanation: Information message

CCL8059I *Error and security pop-ups were already enabled*

Explanation: Information message

CCL8060I *Error and security pop-ups were already disabled*

Explanation: Information message

CCL8061E *Unable to set error and security pop-up state*

Explanation: The Client daemon was unable to set the security and error pop-up state requested by a CICSCLI /E or CICSCLI /N command.

System action: Error and security pop-ups remain unchanged.

User response: Examine any other messages and the Client error log to determine the cause of the error.

CCL8062I *Option *option* ignored because the Client daemon is already started*

Explanation: Information message

CCL8063E *Option *option* cannot be used unless starting the Client daemon*

Explanation: The /F option to specify a configuration file was included on a CICSCLI command, but this option can be specified only when starting the Client daemon with the CICSCLI /S command.

System action: CICSCLI performs no action.

User response: If a different configuration file is required, the product must be stopped and restarted.

CCL8064E *A request to restart the Client daemon timed out*

Explanation: A request to restart the Client daemon timed out while waiting for the Client daemon to stop. Cicscli waits up to 60 seconds for the Client daemon to stop. It can take some time to stop the Client daemon, because all active conversations with the server must complete first. On some network protocols a specific sequence of data flows must be exchanged with the server during shutdown; this can also take some time.

System action: The restart command terminates, and the Client daemon will not be restarted.

User response: Wait for the cclclnt process to terminate; this means that the Client daemon has stopped. Then issue a start request.

CCL8065E *Unable to change status of Client trace to memory buffer*

Explanation: The Client daemon was unable to set the requested trace state.

System action: Service tracing to the memory

buffer remains unchanged.

User response: Examine any other messages and the Client error log to determine the cause of the error.

CCL8066E Unable to start Client trace to memory buffer

Explanation: The Client daemon was unable to start the trace to the memory buffer. This is probably because sufficient storage could not be allocated.

System action: Service tracing to the memory buffer was not started.

User response: Examine any other messages and the Client error log to determine the cause of the error.

CCL8067E A request to start the Client daemon timed out

Explanation: A request to start the Client daemon has timed out because another instance of the Client daemon is currently shutting down. Cicscli waits for up to 60 seconds for the Client daemon instance to shut down. For shutdown to complete, all active server conversations must complete and network connections must have ended and this may take some time.

System action: The start command terminates and the Client daemon will not start.

User response: Wait for the cclclnt process to terminate, showing the Client daemon has stopped, and reissue the start request.

CCL8069E Userid *userid* exceeds the maximum supported length

Explanation: The length of the userid string supplied exceeds the maximum supported by the CICS Client.

System action: CICSCLI performs no action.

User response: Reissue the command with a valid string.

CCL8070E Password *password* exceeds the

maximum supported length

Explanation: The length of the password string supplied exceeds the maximum supported by the CICS Client.

System action: CICSCLI performs no action.

User response: Reissue the command with a valid string.

CCL8072I The Client daemon is trying to restart ...

Explanation: Information message

CCL8073I You must be user root to use the '-f' command

Explanation: Information message

CCL8074I Build Level *INSERT-0, INSERT-1*

Explanation: Information message

CCL8075I -R - Register Client daemon startup parameters

Explanation: Information message

CCL8076I [-A<CICSCLIParam1> [-A<CICSCLIParam2> ...]]

Explanation: Information message

CCL8079W Failed to obtain precise product version information, rc=*return code*

Explanation: The fixpack version of the installed product could not be determined as there was an error reading information from the product information file product.xml. This might be because the product.xml file could not be found or could not be read.

System action: The version of the cicscli utility is displayed.

User response: Check that the file product.xml exists in the product installation directory and that the user issuing the cicscli command has permission to read product.xml. If the problem

persists, contact your service organization.

CCL8080E **Invalid request to start a server connection, server name is required**

Explanation: A CICSCLI command was issued with the /S option but without specifying a server name.

System action: CICSCLI performs no action.

User response: Reissue the command with a server name.

CCL8081E **Invalid request to shut down a server connection, server name is required**

Explanation: A CICSCLI command was issued with the /X option but without specifying a server name.

System action: CICSCLI performs no action.

User response: Reissue the command with a server name.

CCL8082E **Invalid request to abort a server connection, server name is required**

Explanation: A CICSCLI command was issued with the /I option but without specifying a server name.

System action: CICSCLI performs no action.

User response: Reissue the command with a server name.

CCL8083I **Client daemon startup parameters are: 'parameters '**

Explanation: Information message

CCL8090I **-S=server - Start a server connection**

Explanation: Information message

CCL8091I **-X=server - Shut down a server connection**

Explanation: Information message

CCL8092I **-I=server - Abort a server connection**

Explanation: Information message

CTG8200E **No message available for message id ID**

Explanation: The message id indicated could not be located within the message file.

System action: No action has been taken.

User response: Contact your service organization with the message id.

CTG8201E **Unable to open a connection to Gateway daemon on port port : exception message = exception**

Explanation: An exception was thrown when trying to open a connection to the Gateway daemon on the specified port. The exception message is provided.

System action: The requested action cannot be completed.

User response: The exception may provide more information into the problem.

CTG8202E **Exception occurred trying to authenticate with gateway gateway : exception message = exception**

Explanation: An exception was thrown when trying to authenticate the client with the Gateway daemon at the specified URL. The exception message is provided.

System action: The requested action cannot be completed.

User response: The exception may provide more information into the problem.

CTG8203E Unexpected gateway rc=RC trying to authenticate with gateway gateway

Explanation: An unexpected return code was received when trying to authenticate the client with the Gateway daemon at the specified URL.

System action: The requested action cannot be completed.

User response: The return code may provide more information into the problem.

CTG8204E Gateway gateway has security protocols in place not recognized by this program

Explanation: CTGAdmin cannot handle the security protocols in place on the Gateway daemon at the specified URL.

System action: The requested action cannot be completed.

User response: Obtain a later version of ctgadmin.jar, or disable the protocols at Gateway gateway .

CTG8205E Unknown security protocol protocol in gateway gateway reported a failure rc=RC

Explanation: The indicated security protocol, unknown to CTGAdmin, reported a failure of the specified return code.

System action: The requested action cannot be completed.

User response: Obtain a later version of ctgadmin.jar, or disable the protocols at Gateway protocol .

CTG8206E Unexpected gateway rc=RC trying to send request to gateway gateway

Explanation: An unexpected return code was received trying to send the request from the authenticated client.

System action: The requested action cannot be completed.

User response: The return code may provide more information into the problem.

CTG8207E Unable to initialize with authenticated gateway gateway : rc = RC

Explanation: An unexpected return code was received trying to initialize with the Gateway daemon from an authenticated client at the specified URL.

System action: The requested action cannot be completed.

User response: The return code may provide more information into the problem.

CTG8208E Exception occurred trying to send request to gateway gateway : exception message = exception

Explanation: An exception was thrown when trying to send a request from an authenticated client to the Gateway daemon at the specified URL. The exception message is provided.

System action: The requested action cannot be completed.

User response: The exception may provide more information into the problem.

CTG8209E Unexpected rc=RC sending request to gateway gateway

Explanation: An unexpected return code was received trying to send a request to the Gateway daemon from an authenticated client at the specified URL.

System action: The requested action cannot be completed.

User response: The return code may provide more information into the problem.

CTG8210E The gateway gateway reports no management server located

Explanation: The Gateway daemon at the specified URL was unable to locate its management server.

System action: The requested action cannot be completed.

User response: Obtain a trace from the Gateway daemon and contact your service organization.

CTG8211E The gateway *gateway* sent back information which this client cannot read

Explanation: CTGAdmin cannot handle information sent by the Gateway daemon at the specified URL.

System action: The requested action cannot be completed.

User response: Ensure that the Java level on the Java client is at least as high as that on the Gateway daemon. Also check that you have the latest version of ctgadmin.jar installed.

CTG8212E Option *option* must be specified

Explanation: The mandatory option was not specified.

System action: The requested action cannot be completed.

User response: Reissue the command, specifying the required option.

CTG8213E An unexpected error occurred processing option *option*

Explanation: An unexpected error occurred when processing the specified option.

System action: The requested action cannot be completed.

User response: Contact your service organization.

CTG8214E Option *option* cannot be specified more than once

Explanation: The specified option appears more than once in the request; this is not allowed.

System action: The requested action cannot be completed.

User response: Correct the request and try again.

CTG8215E Option *option* is not valid

Explanation: The specified option is not recognized.

System action: The requested action cannot be completed.

User response: Correct the request and try again.

CTG8216E Option *option* must be prefixed with -

Explanation: A parameter was provided but did not start with a minus sign.

System action: The requested action cannot be completed.

User response: Correct the request and try again.

CTG8217E Unexpected internal error

Explanation: An unexpected internal error occurred.

System action: The requested action cannot be completed.

User response: If the problem persists, contact your service organization.

CTG8218I The command completed successfully

Explanation: Information message

CTG8219E Action *action* specified is not valid

Explanation: CTGAdmin did not recognize the specified action.

System action: The requested action cannot be completed.

User response: Correct the request and try again. Refer to help for a list of valid actions.

CTG8220E No recognized options for action *action* specified

Explanation: This action requires options, but no valid ones were specified.

System action: The requested action cannot be completed.

User response: Correct the request and try again.

CTG8221E Parameter for option *option* must be numeric

Explanation: The parameter for the specified option was not a number and is required to be a number.

System action: The requested action cannot be completed.

User response: Correct the request and try again.

CTG8222E Option *option*, parameter *parameter* outside range of *min -max*

Explanation: The parameter for the specified option was not within the range specified by *min -max*.

System action: The requested action cannot be completed.

User response: Correct the request and try again.

CTG8223E Cannot specify the option *fulldatadump* with *dumpoffset* or *truncationsize*

Explanation: For the trace action you cannot specify the *fulldatadump* option with the *dumpoffset* option or the *truncationsize* option.

System action: The requested action cannot be completed.

User response: Correct the request and try again. Specify *fulldatadump* if you want the complete contents of the dump. Specify *dumpoffset*, or *truncationsize*, or both, if you want a partial dump.

CTG8224E IP address or hostname not authorized for gateway *gateway*

Explanation: The client machine is not authorized to perform administrative commands on the specified URL.

System action: The requested action cannot be completed.

User response: Check that the host from which you are issuing the command appears in the list of authorized hosts at the Gateway daemon that you are attempting to administer (configuration tool, TCPAdmin panel).

CTG8225E Option *option*, parameter *parameter* is below minimum value of *min*

Explanation: The parameter for the specified option was too low and cannot be used.

System action: The requested action cannot be completed.

User response: Correct the request and try again.

CTG8226E Gateway daemon does not accept administration requests on port *port*

Explanation: The specified port does not refer to the local Admin port but refers to a standard TCP protocol handler.

System action: The requested action cannot be completed.

User response: specify the port for the local admin handler and reissue the request.

CTG8227E Exception returned from gateway *gateway* : exception message = *exception*

Explanation: An exception was returned from the Gateway daemon at the specified URL.

System action: The requested action cannot be completed.

User response: The exception message will provide more information on the problem.

CTG8228E Specified option(s) *option* unknown for requested action

Explanation: The specified option is not recognized.

System action: The requested action cannot be completed.

User response: Correct the request and try again.

CTG8229E Unexpected rc=*RC* trying to authenticate with gateway *gateway*

Explanation: An unexpected return code was received when trying to authenticate the client with the Gateway daemon at the specified URL.

System action: The requested action cannot be completed.

User response: The return code may provide more information into the problem.

CTG8230E The gateway *gateway* sent back an exception that this client cannot read

Explanation: The Gateway daemon at the URL specified tried to send an exception back which cannot be handled by CTGAdmin.

System action: The requested action cannot be completed.

User response: Ensure that the Java level on the Java client is at least as high as that on the Gateway daemon. Also check that you have the latest version of ctgadmin.jar installed.

CTG8231E JNI Trace file name must be specified

Explanation: For the trace action, the jnifile option was specified but no parameter was given. A parameter must be specified.

System action: The requested action cannot be completed.

User response: Correct the request and try again.

CTG8232I CTGAdmin performed no action

Explanation: Information message

CTG8233E Cannot specify *action* and its short form *abbreviation* together

Explanation: You entered options for an action, but at least one of the options has been specified in its full form and its short form. This is equivalent to specifying the same option twice.

System action: The requested action cannot be completed.

User response: Correct the request by specifying either the full version of the option or its short form.

CTG8234I Immediate shutdown was requested

Explanation: Information message

CTG8235I Normal shutdown was requested

Explanation: Information message

CTG8236E Option *option*, unexpected parameter specified

Explanation: The option specified had an associated parameter, but the option doesn't require a parameter.

System action: The requested action cannot be completed.

User response: Correct the request and try again.

CTG8237E The use of tfile or tf is not supported on z/OS

Explanation: The CICS Transaction Gateway does not support setting the Gateway trace file through SDSF.

System action: The Gateway daemon continues running.

User response: If trace is required to go to a

specific file, specify the file in the configuration file used when the Gateway daemon starts.

CTG8238E The use of jnifile or jf is not supported on z/OS

Explanation: The CICS Transaction Gateway does not support setting the JNI trace file through SDSF.

System action: The Gateway daemon continues running.

User response: If JNI trace is required to go to a specific file, specify the file in the CTG_JNI_TRACE environment variable.

CTG8239I Response received from CICS Transaction Gateway

Explanation: Information message

CTG8240E Cannot specify multiple options for stats action

Explanation: The 'stats' action accepts only one of the '-gs', '-si', '-rg' options.

System action: The requested action cannot be completed

User response: Correct the request and try again

CTG8241W IDs *ids* not recognized

Explanation: A request for statistical information specified one or more IDs that do not exist.

System action: The requested action completes.

User response: Correct the IDs and try again.

CTG8242W ID *id* specified multiple times

Explanation: ID *id* was specified more than once in a request for statistics.

System action: The requested action completes.

User response: Remove the duplicate ID to remove this message.

CTG8243W Statistic *id* is part of resource group *resource group*

Explanation: The command contained a request for statistic *id* and its resource group *resource group*

System action: The requested action completes and the results for *resource group* are displayed.

User response: Remove either the statistic ID or the resource group ID and rerun the command.

CTG8244W Unable to retrieve statistic *id*

Explanation: An error occurred in the CICS TG while attempting to retrieve statistic *id*

System action: The requested action completes.

User response: Check the CICS TG error logs for details of the error.

CTG8245W Unable to retrieve all statistics for resource group *id*

Explanation: An error occurred in the CICS TG while retrieving one or more statistics as part of the request for statistics in resource group *id*.

System action: The requested action completes.

User response: Check the CICS TG error logs for details of the error.

CTG8246E The administration request was sent to an incorrect Gateway daemon port

Explanation: The administration request was rejected as it was sent to an incorrect Gateway daemon port.

System action: The requested action cannot be completed.

User response: Resend the request to the Gateway daemon port identified by the `adminport` parameter.

CTG8247W Null response received from Gateway daemon during *INSERT-0* dump request

Explanation: The Gateway daemon received the dump request but returned a null response

System action: The Gateway daemon returned an invalid response to the dump request, which might not be completed.

User response: Check the Gateway daemon log for earlier messages indicating the dump request completed successfully.

CTG8248E The *INSERT-0* dump type is unsupported in the remote JVM

Explanation: The remote JVM executing the Gateway daemon does not support the *INSERT-0* dump type.

System action: The requested action cannot be completed.

User response: Correct the request by requesting a dump type supported by the remote JVM.

CTG8249E The Gateway daemon encountered a serious error while processing *INSERT-0* dump type (Response code=*INSERT-1*)

Explanation: The request to the Gateway daemon completed but an error was detected.

System action: The Gateway daemon returned an error during the dump request, which might not be completed.

User response: Check the Gateway daemon log for earlier messages indicating the dump request completed successfully.

CTG8250I The Gateway daemon successfully processed the *INSERT-0* dump request

Explanation: Information message

CTG8251E An invalid response was returned from the Gateway daemon during *INSERT-0* dump request

Explanation: The Gateway daemon received the dump request but returned an invalid response

System action: The Gateway daemon returned an invalid response to the dump request, which might not be completed.

User response: Check the Gateway daemon log for earlier messages indicating the dump request completed successfully.

CTG8268I Invoking dump request

Explanation: Information message

CTG8269I Dump request complete

Explanation: Information message

CTG8270W Statistical type *ids* not recognized

Explanation: A request for statistical information specified one or more statistical types that do not exist.

System action: The invalid types are ignored and the requested action completes. If the statistical type list contained no valid statistical types then all results will be displayed.

User response: Correct the specified statistical types and try again.

CTG8273E The '-all' dump option cannot be combined with other dump options

Explanation: If the '-all' dump option is specified it must be the only option.

System action: The requested action cannot be completed.

User response: Correct the request and try again.

CTG8274W Some dump types are unsupported in the remote JVM

Explanation: The remote JVM executing the Gateway daemon does not support some dump types.

System action: The requested action completes, but some dump types are not performed.

User response: Check the Gateway daemon log for earlier messages indicating which dump types are unsupported.

CTG8275E Cannot communicate with the Windows Service Control Manager (Error Code=*error code*)

Explanation: An error occurred during communications with the Windows Service Control Manager. The Windows OpenSCManager function returned the specified error.

System action: The requested action does not succeed.

User response: If the problem persists, contact your service organization.

CTG8276E Insufficient security permissions to communicate with the CICS TG service

Explanation: The current user has insufficient authority for the administration of Windows Services.

System action: The requested action does not succeed.

User response: Check the Windows application event log for further information on the error. Ensure that the current user has the required level of authority.

CTG8277E Cannot communicate with the CICS TG service (Error Code=*error code*)

Explanation: An error occurred during communications with the IBM CICS Transaction Gateway service. The Windows OpenService function returned the specified error.

System action: The requested action does not succeed.

User response: Check the Windows application event log for further information on the error. If the problem persists, contact your service organization.

CTG8278E Cannot start the CICS TG service (Error Code=*error code*)

Explanation: An error occurred while the IBM CICS Transaction Gateway service was starting. The Windows Service Control Manager returned the specified error.

System action: The requested action does not succeed.

User response: Check the Windows application event log for further information on the error. If the problem persists, contact your service organization.

CTG8279I The CICS TG service is already started

Explanation: Information message

CTG8280E The CICS TG service failed to start (Error Code=*error code*), check the Windows application event log

Explanation: An error occurred while the IBM CICS Transaction Gateway service was starting. The service returned the specified error.

System action: The requested action does not succeed.

User response: Check the Windows application event log for further information on the error. If the problem persists, contact your service organization.

CTG8281I The CICS TG service is starting

Explanation: Information message

CTG8282E The CICS TG service failed to start (Windows error code=*error code*), check the Windows application event log

Explanation: An error occurred while the IBM CICS Transaction Gateway service was starting. Windows returned the specified error.

System action: The requested action does not succeed.

User response: Check the Windows application event log for further information on the error. If the problem persists, contact your service organization.

CTG8284I IBM CICS Transaction Gateway Desktop Edition Informational Dump VProduct Version Build Level Build Level Dump Data

Explanation: Information message

CTG8285I IBM CICS Transaction Gateway Informational Dump VProduct Version Build Level Build Level Dump Data

Explanation: Information message

CTG8286I Informational dump complete

Explanation: Information message

CTG8400I Using configuration file *filename*

Explanation: Information message

CTG8401I The following ciphers are enabled:

Explanation: Information message

CTG8402I Request Monitoring Exit *class_name* is enabled

Explanation: Information message

CTG8403E Request monitoring exit *class_name* failed to initialize with exception *exception_message*

Explanation: During initialization the request monitoring exit *class_name* failed with exception *exception_message* .

System action: The Gateway daemon or Java

Client classes will continue to initialize with the exit disabled.

User response: Correct the error in the exit identified by the *exception_message* and restart the Gateway daemon to enable this exit.

CTG8404I JSSE version unknown

Explanation: Information message

CTG8405I JSSE provider info: *INSERT-0*

Explanation: Information message

CTG8406E Request Monitoring Exit *class_name* has failed with exception *exception_message*

Explanation: The request monitoring exit *class_name* failed with exception *exception_message* when the `exitFired` method was called.

System action: Processing continues with the exit disabled.

User response: Correct the error in the exit identified by the *exception_message* and restart the Gateway daemon to enable this exit.

CTG8407I -applid=<applid> - the Gateway daemon applid

Explanation: Information message

CTG8408I -applidqualifier=<qual> - the Gateway daemon applid qualifier

Explanation: Information message

CTG8409E *Stream* log is defined but an output destination has not been specified

Explanation: The specified log handler has failed to start. The destination has not been specified.

System action: This message is written to the error stream. The log that failed to start is now written to `stdout` or `stderr`, depending on

whether the error or information log was being configured.

User response: Check the configuration file to ensure that the destination is specified correctly and has valid parameters.

CTG8410E *Stream log is configured to an unknown destination*

Explanation: The specified log handler has failed to start. The destination is either unknown, or unsupported on this platform.

System action: This message is written to the error stream. The log that failed to start is now written to stdout or stderr, depending on whether the error or information log was being configured.

User response: Check the configuration file to ensure that the destination is specified correctly and has valid parameters.

CTG8411E *Logging to file is not supported on z/OS*

Explanation: The file logging handler has failed to start because it is unsupported on this platform.

System action: This message is written to the error stream. The log stream that failed to start is now written to stdout or stderr, depending on whether the error or information log was being configured.

User response: Correct the configuration file to use a log handler that is supported on this platform.

CTG8412E *Stream log to destination has missing parameters*

Explanation: The specified log failed to start because expected parameters are missing.

System action: This message is written to the error stream. The log that failed to start is now written to stdout or stderr, depending on whether the error or information log was being configured.

User response: Add the parameters which are missing, and ensure that they have valid values.

CTG8413E *Stream log to destination has invalid parameters*

Explanation: The specified log failed to start because the parameters supplied are outside permitted values.

System action: This message is written to the error stream. The log that failed to start is now written to stdout or stderr, depending on whether the error or information log was being configured.

User response: Correct the errors in the configuration file.

CTG8414E *Stream log unable to write to file: name*

Explanation: The log to file failed to start because it was unable to write to the specified file.

System action: This message is written to the error stream. The log that failed to start is now written to stdout or stderr, depending on whether the error or information log was being configured.

User response: Check that the file system is not full, and that the CICS Transaction Gateway has the necessary permissions to write to the file.

CTG8415W *Log to file has mismatched parameter: parameter, so using largest value (value) to continue*

Explanation: Both information and error log streams are configured to log to the same file, but have different parameters specified for the file properties.

System action: This message is written to the error stream. The file handler will start using the larger of the two values.

User response: The reason gives more information about the mismatched parameters. Correct the configuration file by either specifying different filenames for the two log streams, or

ensure that both streams have the same parameters if you want to log both streams to the same file.

CTG8416W **Unknown log stream:** *stream*

Explanation: An incorrect log stream was specified.

System action: This message is written to the error stream.

User response: Check the configuration file to ensure the log stream is specified correctly. Refer to the Administration book for details of the supported stream types.

CTG8417I **The statsrecording parameter is not specified so SMF Recording is not enabled**

Explanation: Information message

CTG8418W **The statsrecording parameter has been ignored as it is not valid on this platform**

Explanation: The configuration file `ctg.ini` contains the `statsrecording` parameter, a distributed platform Gateway daemon does not support the `statsrecording` parameter.

System action: The Gateway daemon continues its initialization process.

User response: To prevent this message occurring in future, remove the `statsrecording` parameter from configuration file `ctg.ini`. Edit the file manually, or use the Configuration tool to open and then save the file, this removes the parameter.

CTG8419I **SMF Recording is not enabled**

Explanation: Information message

CTG8420E **The CICS Transaction Gateway does not support HTTP protocols**

Explanation: Either the HTTP or HTTPS protocol handler is specified in the configuration file.

System action: The Gateway daemon continues, but does not start the handler for the specified protocol.

User response: Migrate to the TCP or SSL handlers, and then remove references to the protocol from the configuration file by loading the file into the Configuration Tool and then saving it, or by editing the file in a text editor.

CTG8421I **SMF Recording is enabled**

Explanation: Information message

CTG8422E **The CICS TG Gateway daemon failed to initialize the SMF recording facility with an internal exception** *exception*

Explanation: An internal error occurred in the logic that initializes SMF recording.

System action: The Gateway daemon will continue to run. Statistics will not be recorded to SMF.

User response: If this problem persists contact your service organization.

CTG8423E **An internal exception** *exception* **occurred whilst processing data for an SMF record**

Explanation: An internal error occurred in the logic that processes data to send to SMF.

System action: The Gateway daemon will continue to run. The current set of statistics will not be written to SMF and the data will be lost. The Gateway daemon will continue to attempt to write to SMF.

User response: If this problem persists contact your service organization.

CTG8424E **The command line override parameter is only supported on z/OS**

Explanation: The command line override 'parameter' was specified on a distributed

platform Gateway daemon when it is only supported on z/OS.

System action: This message is logged and the Gateway daemon fails to start.

User response: Restart the Gateway daemon without specifying the command line override.

CTG8425E **The name parameter exceeds the maximum length of len characters**

Explanation: Parameter 'NAME' has a maximum length of 'LEN' characters. This has been exceeded.

System action: This message is output and the Gateway daemon fails to start.

User response: Reconfigure the parameter to be within the allowed limit and restart the Gateway.

CTG8426I **The applid is *applid***

Explanation: Information message

CTG8427I **The applid qualifier is *applidqualifier***

Explanation: Information message

CTG8428I **Client connection associated with Connection Manager *Connection Manager* has client applid *Client Applid* and qualifier *Client Applid Qualifier***

Explanation: Information message

CTG8429I **Established new IPIC connection to CICS server *CICS Server name as defined in the INI file*. with: negotiated session limit=*Number of simultaneous sessions allowed on this connection*, CICSAPPLID=*Applid of the CICS region*. CICSAPPLIDQUALIFIER=*Applid Qualifier of the CICS region*., HOSTNAME=*Hostname used when connecting to CICS*., PORT=*Port number used when connecting to CICS*., sockets=*Number of sockets used by this connection***

Explanation: Information message

CTG8430I **Closed IPIC connection to CICS server *CICS Server name as defined in the INI file*.**

Explanation: Information message

CTG8431E **Handshake failure for IPIC connection to CICS server *server name* response code=*CICS response code*, reason=*reason [numeric reason code]***

Explanation: The physical socket connection was established, but CICS rejected the connection.

System action: Any requests that were attempting to use the CICS server *server name* will fail with ECI_ERR_NO_CICS. Requests received after this point will also fail until the server retry interval has expired, at which point the connection will be retried.

User response: Check that the IPIC server *server name* is defined correctly in the ctg.ini file (remote mode) or Server URL (local mode) and check the reason text *reason* . Also check the console log for the CICS server and topic "DFHIS1011" in the CICS Information Center for an explanation of the reason.

CTG8432W CICS has purged IPIC conversation *conversation ID* to CICS server *server name* with mirror task *task*, Transaction ID: *XID or IPIC LUW token (if relevant)*

Explanation: The CICS connection defined as *server name* has been purged at the CICS end. This has caused all work on that connection to be aborted. The work can be identified with the transaction token *XID or IPIC LUW token (if relevant)* and mirror transaction *task* .

System action: CICS Transaction Gateway ends the work and frees each purged conversation to CICS.

User response: No user response required.

CTG8433E Connection failure for IPIC connection to CICS server *server name* reason = *reason text*

Explanation: The physical socket connection could not be established to the target CICS server.

System action: Any requests that were attempting to use the CICS server *server name* will fail with ECI_ERR_NO_CICS. Requests that are received after this point will also fail until the server retry interval has expired, at which point the connection will be retried.

User response: Check that the IPIC server is defined correctly in the ctg.ini file (remote mode) or server URL (local mode) and check the reason text *reason text* .

CTG8434E IPIC server *server name* configuration paramter error; Reason: *reason text*

Explanation: There is an error in the specified IPIC server definition in the configuration file.

System action: The CICS Transaction Gateway fails to start.

User response: Edit the configuration file to correct the IPIC server definition. Refer to the CICS Transaction Gateway Information Center for details of the IPIC server parameters.

CTG8435W Duplicate IPIC Server Definition *server name* in ctg.ini

Explanation: The named IPIC server definition is a duplicate of another definition within the ctg.ini file.

System action: The Gateway daemon will ignore the duplicate server definition.

User response: Check that the IPIC server is defined correctly in the ctg.ini file.

CTG8436W An IPIC Server Definition in the INI file is not of the form SECTION IPICSERVER=NAME

Explanation: The IPIC Server definition was not created properly in the configuration file.

System action: The Gateway will ignore the server definition.

User response: Check that each IPIC server is defined correctly in the INI file.

CTG8437I Shutdown statistics: *INSERT-0*

Explanation: Information message

CTG8438W The initial number of worker threads is greater than maximum number of worker threads

Explanation: The value specified for the initial number of worker threads in the configuration settings is greater than the value specified for the maximum number of worker threads.

System action: The initial number of worker threads is reset to be the value of maximum number of worker threads.

User response: Adjust the configuration settings to ensure that the value of the initial number of worker threads (initworker) is set less than or equal to the value of the maximum number of worker threads (maxworker).

CTG8439W The initial number of connection manager threads is defined to be greater than the maximum number of connection manager threads

Explanation: The value specified for the initial number of connection manager threads in the configuration settings is greater than the value specified for maximum number of connection manager threads.

System action: The initial number of connection manager threads is reset to be the value of maximum number of connection manager threads.

User response: Adjust the configuration settings to ensure that the value of the initial number of connection manager threads (initconnect) is set less than or equal to the value of the maximum number of connection manager threads (maxconnect).

CTG8440E Duplicate Logical CICS Server definition *server name* detected in the configuration file

Explanation: The Logical CICS server definition is a duplicate of another definition in the configuration file

System action: The CICS Transaction Gateway fails to start

User response: Edit the configuration file to define unique Logical CICS server definitions

CTG8441E Multiple definitions of *property name* detected in the configuration file at line *line number*

Explanation: The configuration file contains more than one definition for the property

System action: The CICS Transaction Gateway fails to start

User response: Edit the configuration file to correct the duplicate property

CTG8442W Both the statsport parameter and the configuration for statistics API handler are specified

Explanation: The configuration file contains the statsport configuration parameter and the configuration for the statistics API handler

System action: The CICS Transaction Gateway starts using the configuration for the statistics API handler

User response: Edit the configuration file to remove the statsport parameter

CTG8443E A logical CICS server definition in the configuration file is not specified in the correct format:
SECTION
LOGICALSERVER=NAME

Explanation: The logical CICS server definition is not specified correctly in the configuration file.

System action: The CICS Transaction Gateway fails to start.

User response: Check that each logical CICS server is defined correctly in the configuration file.

CTG8444E Multiple SERVER fields in logical CICS server definition *server name*

Explanation: The definition for the logical CICS server contains multiple SERVER definitions.

System action: The CICS Transaction Gateway fails to start.

User response: Correct the definition in the configuration file to contain a single SERVER definition.

CTG8445E Error reading logical CICS server definition *server name* from the configuration file: Reason = *reason*

Explanation: An error occurred while reading the logical CICS server definition.

System action: The CICS Transaction Gateway fails to start.

User response: Check that the logical CICS server is defined correctly in the configuration file.

CTG8446E CICS Request Exit *name* failed to initialize with exception *message*

Explanation: During initialization the CICS request exit *name* failed with exception *message*

System action: The CICS Transaction Gateway fails to start.

User response: Correct the error identified by the exception *message* and restart the CICS Transaction Gateway

CTG8447I CICS Request exit *name* installed successfully

Explanation: Information message

CTG8448I Command line options updated *property* to be *value*

Explanation: Information message

CTG8449E Unexpected configuration parameter *property* at line *line number* in the configuration file

Explanation: The configuration parameter *property* is unknown, is in the wrong section or is unsupported on this platform.

System action: The CICS Transaction Gateway does not start

User response: Edit the configuration file to correct the error and start the CICS Transaction Gateway

CTG8450E An ENDSECTION label is expected before line *line number* in the configuration file

Explanation: The configuration file parser encountered the start of a new section when the previous section had not been terminated with an ENDSECTION label.

System action: The CICS Transaction Gateway does not start.

User response: Edit the configuration file to correct the error and start the CICS Transaction Gateway.

CTG8451W Specified retry count is less than zero

Explanation: The retry count specified by the CICS request exit was less than zero

System action: The CICS Transaction Gateway continues to start and uses zero as the retry count value

User response: Update the CICS request exit to specify a retry count greater than or equal to zero and restart the CICS Transaction Gateway

CTG8452W Exception *exception* thrown by CICS request exit

Explanation: The request failed and threw an exception in the CICS request exit

System action: The CICS Transaction Gateway continues and returns an error to the Java client application

User response: Check why the CICS request exit threw an exception and update the exit

CTG8453I Logical CICS servers disabled due to CICS request exit configuration

Explanation: Information message

CTG8454E Value *parameter value* is not valid for configuration parameter *parameter name* at line *line number*

Explanation: The configuration file parameter value is not valid

System action: The CICS Transaction Gateway fails to start

User response: Edit line *line number* of the configuration file to correct the parameter value. Refer to the CICS Transaction Gateway Information Center for details of the valid values.

CTG8455I Successfully started the local administration handler on port *port*

Explanation: Information message

CTG8456I Successfully started the local statistics API handler on port *port*

Explanation: Information message

CTG8457I Successfully started the local administration handler on port *port*, bound to address *address*

Explanation: Information message

CTG8458I Successfully started the statistics API handler on port *port*, bound to address *address*

Explanation: Information message

CTG8459W statsport parameter found in configuration file

Explanation: The configuration of the statistics API handler has been increased to include the IP address to bind to and the maximum number of clients that can connect, to allow statistics API programs to connect from remote clients. The specified configuration file contains the old style configuration.

System action: The statistics API handler is started on the specified port and accepts requests from the loopback device only.

User response: Update the configuration file to use the new configuration parameters for the statistics API.

CTG8460E Parameter *parameter name* on line *line number* of the configuration file is not supported on this platform

Explanation: The configuration file parameter *parameter name* is not supported by the CICS Transaction Gateway for this platform.

System action: The CICS Transaction Gateway does not start.

User response: See the Information Center for information about which configuration parameters are supported on this platform. Correct the configuration file so that it does not use parameter *parameter name* and start the Gateway daemon.

CTG8461I Successfully initialized trace plug-in '*plug-in name*'

Explanation: Information message

CTG8462E Failed to initialize trace plug-in '*plug-in name*' with reason code = '*reason code*'

Explanation: During initialization of the CICS Transaction Gateway trace mechanism, the trace plug-in *plug-in name* was found to be unusable.

System action: The CICS Transaction Gateway will fail to start.

User response: If you have specified an alternative to the default trace implementation using the gateway.T.plugin java system property, then remove this property from your configuration and start the CICS Transaction Gateway. If you have been asked to use an alternative trace plug-in by IBM, then contact your service organization providing full details of the message.

CTG8463E Section *section name* is not supported on this platform

Explanation: The configuration file section *section name* is not supported by the CICS Transaction Gateway for this platform.

System action: The CICS Transaction Gateway does not start.

User response: See the Information Center for information about which configuration sections are supported on this platform. Correct the configuration file so that it does not use section *section name* and start the Gateway daemon.

CTG8464W Information left in RRS after transaction completion for XID *xid*

Explanation: An XA transaction has been successfully completed in CICS but Gateway information has been left in RRS.

System action: The CICS Transaction Gateway continues.

User response: Manually commit the UR in RRS that is associated with the XID listed.

CTG8465E Multiple definitions of section type *section name* detected in the configuration file at line *line number*

Explanation: The configuration file contains more than one definition for the *section name* section

System action: The CICS Transaction Gateway does not start

User response: Correct the configuration file so that there is only a single definition for the section *section name* .

CTG8466E Multiple definitions of section type *section type* with name *section name* detected in the configuration file at line *line number* . Previous definition at line *line number* .

Explanation: The configuration file contains a previous section of type *section type* with name *section name* .

System action: The CICS Transaction Gateway does not start.

User response: Correct the configuration file so there is only a single definition for section *section type* with name *section name* .

CTG8467E Duplicate server definition with name *section type* detected in the configuration file at line *section name* . Previous definition at line *line number* .

Explanation: The configuration file contains a

previous server definition with name *section type* .

System action: The CICS Transaction Gateway does not start.

User response: Correct the configuration file so there is only a single server definition with name *section type* .

CTG8468W CICS request exit attempted to retry an XA request to server *current server name* having already attempted to use server *previous server name* which uses a different protocol

Explanation: When retrying a request the CICS request exit returned a server that uses a different protocol than the CICS server the request was originally sent to.

System action: The request fails, an error is returned to the client application and this message written to the log.

User response: Update the CICS request exit to ensure that CICS servers that are used for retrying XA requests all use the same protocol as the original CICS server.

CTG8469I Request monitoring is active

Explanation: Information message

CTG8470E Information and error message logging to console is not supported

Explanation: The Gateway daemon failed to start because a log destination of console is not supported on this platform.

System action: The Gateway daemon fails to start. If the error log destination is configured to use console, this is ignored and this message is written to the default error log.

User response: Edit the configuration file (ctg.ini) to use a file log destination.

CTG8471E *log stream log destination does not contain valid file*

Explanation: The Gateway daemon failed to start because the log destination does not specify a valid output file.

System action: The Gateway daemon fails to start. This message is written to the error log.

User response: Edit the configuration file (ctg.ini) to specify a valid file destination.

CTG8472E *The CICS request exit `class_name` has thrown an exception while processing a command event with data `exit_event_data` . Exception: `exception_message` .*

Explanation: The CICS request exit class has thrown an exception from the eventFired method while processing a command event.

System action: Processing continues.

User response: Review the stack trace and correct the error in the CICS request exit.

CTG8473E *The CICS request exit `class_name` has thrown an exception while processing event `exit_event` . Exception: `exception_message` .*

Explanation: The CICS request exit class has thrown an exception from the eventFired method while processing event.

System action: Processing continues.

User response: Review the stack trace and correct the error in the CICS request exit.

CTG8474E *The CICS request exit `class_name` does not implement method `method_name`*

Explanation: The specified CICS request exit class does not implement a required interface method.

System action: Gateway daemon initialization fails.

User response: Update the CICS request exit to

implement the required methods of interface com.ibm.ctg.ha.CICSRequestExit or remove the CICS request exit from the Gateway daemon configuration and restart the Gateway daemon.

CTG8475W *The name of `section` `section name` contains characters that are not supported*

Explanation: The specified section has a name containing characters that are not in the set of supported characters for server definitions. Supported characters are A-Z and 0-9 and '@', '#', '\$', '.'.

System action: The Gateway daemon continues.

User response: Edit the name of the specified section in the configuration file so that it contains only supported characters.

CCL8501I *INSERT-0 - CICS Client BMS Conversion Utility*

Explanation: Information message

CCL8502I *Command options are:*

Explanation: Information message

CCL8503I */C - To generate C++ files*

Explanation: Information message

CCL8504I */B - To generate Visual Basic files*

Explanation: Information message

CCL8520I *Issue 'INSERT-0 /?' to list command options*

Explanation: Information message

CCL8521E *Unknown command option `option`*

Explanation: An invalid command option was specified when using the CICSBMSC command.

System action: CICSBMSC performs no action.

User response: Reissue the command with the correct option.

CCL8522E Input file *filename* must have .bms or .BMS suffix

Explanation: The CICSBMSC command requires a BMS source file as input.

System action: CICSBMSC performs no action.

User response: Reissue the command with the correct BMS source file name.

CCL8523E Unable to open BMS input file *filename*

Explanation: The CICSBMSC command could not find, or could not read the supplied BMS source file.

System action: CICSBMSC performs no action.

User response: Check that the BMS source file name is correct, that the file is available and can be read.

CCL8524E Unable to open output files

Explanation: The CICSBMSC command is unable to create output files.

System action: CICSBMSC processing ends.

User response: Check that there is sufficient disk space available for the CICSBMSC output files.

CCL8525I Processing BMS map *INSERT-0*

Explanation: Information message

CCL8526E Error in BMS source file *filename* line *number*

Explanation: The CICSBMSC command has found an error in an input BMS source file.

System action: CICSBMSC processing ends.

User response: Check the BMS macro source at the line indicated.

CCL8527I C++ output files *INSERT-0* and *INSERT-1* created

Explanation: Information message

CCL8528I Visual Basic file *INSERT-0* created

Explanation: Information message

CCL8529I Processing is complete

Explanation: Information message

CTG8687E Unauthorised Switch_Context failed CTXSWCH Return code RC=*INSERT-0* LUW(*INSERT-1*) Context switch to(*INSERT-2*) from(*INSERT-3*) Call_Type = *INSERT-4* Extend_Mode = *INSERT-5*

Explanation: An RRS context switch for a non-XA transaction has failed. The context switch failure will occur if the target RRS context is already in use by another worker thread. The CTXSWCH return code is given as a hexadecimal value and can be found in "MVS Programming: Resource Recovery" (SA22-7616).

System action: The Gateway daemon will continue to process requests.

User response: Check the CICS TG log for further RRS-related error messages that might indicate a problem with the RRS registration. The target CICS region might show mirror transactions waiting on resource type RRMSEXIT. Restarting the Gateway daemon should allow any such mirror transactions to end. If no obvious RRS registration errors are found, gather any CICS TG logs, traces, dumps and RRS errors and contact IBM Software Support.

CTG8688I JNI has been configured to trace via DFHTREX only

Explanation: Information message

CTG8689E LUW token creation has failed for context token (*INSERT-0*)

Explanation: While processing an extended mode ECI request, CICS TG has been unable to create a new LUW token to represent the transaction.

System action: The current ECI request will fail. The Gateway daemon will terminate with ABEND code 1273.

User response: Gather any CICS TG logs, traces, dumps, and RRS errors and contact IBM Software Support. Refer to the User Response information for CICS TG ABEND code 1273 for further guidance.

CTG8800I CTGJAVA - CICS Transaction Gateway Java location tool

Explanation: Information message

CTG8801I CTGCFG - CICS Transaction Gateway Configuration Tool launcher

Explanation: Information message

CTG8803I CTGADMIN - CICS Transaction Gateway Administration Tool launcher

Explanation: Information message

CTG8804E The CICS Transaction Gateway is not installed properly

Explanation: The CICS Transaction Gateway is not installed correctly. Registry keys are created at install time.

System action: The command line application terminates.

User response: Reinstall the CICS Transaction Gateway.

CTG8805E Unable to read from the registry

Explanation: The program tried to read from the registry but cannot read the appropriate registry key. This is normally caused by security settings in the registry.

System action: The command line application terminates.

User response: Get your system administrator to give read permission to the registry key HKEY_LOCAL_MACHINE\SOFTWARE\IBM\CICS Transaction Gateway.

CTG8806E Unable to write to the registry

Explanation: The program tried to write to the registry but cannot write the appropriate registry key. This is normally caused by security settings in the registry.

System action: The command line application terminates.

User response: Logon as a user who has permissions to write to either HKEY_LOCAL_MACHINE\SOFTWARE\IBM\CICS Transaction Gateway. Then repeat the command line option.

CTG8807E Unable to find a JVM

Explanation: When trying to run a Java application the application launcher was unable to find a JVM.

System action: The command line application terminates.

User response: Install a JVM, or use the CTGJAVA command to point to the Java Virtual Machine to run.

CTG8808E Unable to find Java runtime classes

Explanation: When trying to run a Java application, the Java runtime classes could not be found.

System action: The command line application terminates.

User response: Install the Java Virtual Machine (JVM) correctly.

CTG8809E An unexpected error occurred locating a JRE

Explanation: An unexpected failure occurred while attempting to retrieve the location of the JRE.

System action: The process terminates.

User response: If the problem persists, contact your service organization.

CTG8810I **Current JVM set to:** *INSERT-0*

Explanation: Information message

CTG8811I **-s=filename - Specifies the JVM to use**

Explanation: Information message

CTG8812I **-v - View the current JVM being used**

Explanation: Information message

CTG8813I **-a - Autoconfigure the JVM**

Explanation: Information message

CTG8814I **CTGJAVA - CICS Transaction Gateway Java location tool**

Explanation: Information message

CTG8815E **File does not exist** *Registry string to the file.*

Explanation: The file specified does not exist.

System action: The command line application terminates.

User response: Specify the fully qualified path to a Java Virtual Machine.

CTG8816I **-j<argument> - argument to pass to the JVM**

Explanation: Information message

CTG8817I **-c<argument> - argument to pass to the 'cicscli' command**

Explanation: Information message

CTG8818E **The JVM specified is unsupported**

Explanation: The Java Virtual Machine pointed to by the CTGJAVA command is of an unsupported level.

System action: The command line application terminates.

User response: Specify a supported Java Virtual Machine to CTGJAVA.

CTG8819E **The currently set JVM is unsupported**

Explanation: When trying to run a Java application, the Java Virtual Machine was found to be of an unsupported level.

System action: The command line application terminates.

User response: Use the CTGJAVA command to point to a supported Java Virtual Machine.

CTG8820E **This application is not supported on this Operating System**

Explanation: The JVM detected an attempt to run an application on an unsupported operating system.

System action: The application terminates.

User response: Ensure that you are running a supported Operating System.

CTG8821E **The CICS Transaction Gateway was unable to start the Client daemon**

Explanation: The Client daemon must be started before the Gateway daemon. The 'ctgstart' command called 'cicscli' to start the Client daemon and the Client daemon failed to start.

System action: The 'ctgstart' command terminates, the Gateway daemon is not started.

User response: Examine any other console

messages and the Client error log to determine the cause of the error.

CTG8822E **An attempt to open an internal pipe for command *command line* failed with return code *rc***

Explanation: The operating system returned the specified error to the application.

System action: The command line application terminates.

User response: Retry the start attempt. If problems persist, contact your Service organization.

CTG8823E **The Java Virtual Machine launcher specified is unsupported**

Explanation: The Java Virtual Machine (JVM) launcher, javaw.exe, is unsupported.

System action: The command line application terminates.

User response: Configure the JVM launcher using the CTGJAVA utility to use "java.exe".

CTG8986I **CICS Transaction Gateway start type is *Start Type***

Explanation: Information message

CTG8987E **Invalid value *Invalid value* specified for the START paramter**

Explanation: The START parameter is specified with an invalid value.

System action: This message is output and the Gateway daemon does not start.

User response: Specify a valid value for START, or remove the START paramter, and restart the Gateway daemon. See the CICS TG Information Center for valid values of START.

CTG8989E **The *Handler* protocol handler is specified with no corresponding protocol definition *Definition***

Explanation: The protocol handler definition is not defined in the Gateway daemon configuration file.

System action: This message is output and the Gateway daemon does not start.

User response: Edit the Gateway daemon configuration file to specify a valid protocol handler definition.

CTG8990E **The required *Handler* protocol handler parameter *Parameter* is not specified**

Explanation: The protocol handler parameter is not defined in the configuration file or, if available, as a command line override.

System action: This message is output and the Gateway daemon does not start.

User response: Correct the configuration and retry.

CCL9042E **Message queue header does not contain valid eye-catcher, eye-catcher = *eye-catcher (eye-catcher)***

Explanation: The Inter-Process Communication (IPC) transport mechanism attempted to retrieve data from a message queue that was of an unrecognized type.

System action: The process continues.

User response: Investigate the Windows event log and the Client daemon log files for error messages and restart the Client application. If the problem persists, contact your IBM service organization.

CCL9091E **Unable to communicate with Client daemon**

Explanation: The Client application was unable to communicate with the Client daemon process.

System action: This message is written to the log file. The Client application will continue.

User response: Check the Windows application event log and in the product logs to find out

why the Client daemon was not running. If the problem persists, contact your IBM service organization.

CCL9092E Unable to locate pipe entry for session type *session type* and session id *session* in pipe table

Explanation: The Inter-Process Communication (IPC) layer could not locate the specified named pipe entry.

System action: This message is written to the log file. The process will continue.

User response: Investigate the Windows event log and the Client daemon log files for error messages and restart the Client daemon and any Client applications. If problems persist, contact your IBM service organization.

CCL9093E Unable to locate named pipe for session id *session*

Explanation: An internal error has occurred. The interprocess communication could not locate a named pipe for the specified session id.

System action: This message is written to the log file. Processing continues.

User response: Restart the IBM CICS Transaction Gateway service. If the problem persists, contact your IBM service organization.

CCL9094E Session type mismatch, session type *session type*, expected session type *session type*

Explanation: An internal processing error has occurred. The interprocess communication encountered an unexpected session type.

System action: This message is written to the log file. Processing continues.

User response: Restart the IBM CICS Transaction Gateway service. If the problem persists, contact your IBM service organization.

CCL9095E A Client application was unable to connect to the Client daemon

Explanation: The Interprocess communication failed to connect the client application to the Client daemon after a number of attempts.

System action: This message is written to the log file. The Client application is unable to use the Client daemon services.

User response: Restart the IBM CICS Transaction Gateway service and retry. If the problem persists, contact your IBM service organization.

CCL9098E Unrecognised request type *request type*

Explanation: The Inter-Process Communication (IPC) layer did not recognise request type in the data received from another process.

System action: This message is written to the log file. Processing continues.

User response: This is an internal error. If the problem persists, contact your IBM service organization.

CCL9116E Unable to send message to queue *number*

Explanation: An error occurred writing an internal message to a message queue. In UNIX and Linux systems, the Client daemon communicates internally using message queues. In systems other than AIX, the default configuration settings for these queues are too small to allow for large client data flows.

System action: The Client daemon is unable to operate.

User response: Change the configuration settings of the message queues to allow for large client data flows. The way that you do this depends on your operating system. Refer to the UNIX and Linux Administration book for further information on configuring message queues. If this does not resolve the problem, contact your service organization.

CCL9117I This may be because you have not configured your system for large message queues

Explanation: Information message

CCL9118I Refer to the Administration book for further information

Explanation: Information message

CCL9124E Unable to determine maximum file descriptors (Function = *function*, Error code=*error*), using default value

Explanation: On a UNIX or Linux platform, a call to determine the maximum number of file descriptors available to a process has returned an error code. The *error* value indicates the value from the *function* system call.

System action: The message is written to the error log. The Client daemon continues using a default number of file descriptors.

User response: If the problem persists, contact your service organization.

CCL9125E Error releasing file descriptors (Errno = *error*), continuing processing

Explanation: On a UNIX or Linux platform, an attempt to close all inherited file descriptors failed, for at least one descriptor. The *error* value indicates the return code from the operating system.

System action: The message is written to the error log and the Client continues. Applications using the Client may encounter problems restarting after termination if this error has occurred.

User response: If the problem persists, contact your service organization

CCL9126E Required message queue (key *queue key*) already exists, will not reuse

Explanation: The Client API component tried to create an Operating System message queue that already exists. The queue key is *queue key* and is based on the process ID (PID) of the Client API component. In the case of the CICS TG, this will be the Gateway daemon PID and in the case of the CICS UC, this will be the Client application PID.

System action: The message is written to the client log and user application requests can no longer be processed.

User response: Shut down and then restart the Gateway daemon, or restart the Client application, as appropriate.

CCL9303E Memory allocation failed with DosAllocMem return code: *error*

Explanation: The Client was unable to allocate the amount of memory required to hold the returned data. The *error* value indicates the system return code.

System action: The message is written to the error log and the function fails.

User response: Try reducing the amount of data requested and try again. If the problem persists contact your service organization.

CCL9304E Invalid call type (*call_type*)

Explanation: An invalid value (as indicated by *call_type*) was specified for the ECI call type.

System action: The message is written to the error log and the function fails.

User response: Correct the program so that a supported value is specified for the ECI call type.

CCL9305E The commarea size specified exceeds the length of data supplied

Explanation: The value specified for the length of the commarea is larger than the amount of data supplied.

System action: The message is written to the error log and the call fails.

User response: Correct the program so that the specified length of the commarea is equal to, or exceeds, the amount of data being passed to the ECI.

CCL9306E The value specified for Extended Mode (*value*) is invalid

Explanation: An invalid value (as indicated by *value*) has been specified for the ECI extended mode parameter.

System action: The message is written to the error log and the call fails.

User response: Correct the program so that a supported value is specified for the extended mode parameter.

CCL9307E The ECI version specified (*version*) is not valid

Explanation: An invalid value (as indicated by *version*) has been specified for the ECI version parameter.

System action: The message is written to the error log and the function fails.

User response: Correct the program so that a supported value is specified for the ECI version parameter.

CCL9337E Error during ECI call and commarea not returned

Explanation: The call to the ECI has completed but an error was detected. No data is returned to the calling application in the commarea.

System action: The message is written to the trace file if tracing is enabled.

User response: Information on the type of error can be determined from earlier entries in the trace log.

CCL9343I CICS_EciListSystems call completed successfully

Explanation: Information message

CCL9344E Error during CICS_EciListSystems call, return code *rc*

Explanation: An error was encountered during a CICS_EciListSystems call. The value of *rc* indicates the failing error code.

System action: The message is written to the trace file if tracing is enabled.

User response: Refer to the Information Center for an explanation of the error codes.

CTG9500E Maximum number of Java arguments exceeded

Explanation: The maximum number of parameters that can be passed to the CICS TG has been exceeded.

System action: The CICS TG service terminates.

User response: Reduce the number of arguments and try again.

CTG9501E Cannot load JNI dll: *jvm.dll*

Explanation: The service cannot find the Java native runtime dll and is unable to start the CICS TG.

System action: The CICS TG service terminates.

User response: Inspect the Windows application event log for other messages. Make sure the Java Virtual Machine (JVM) for use with the product is correctly defined, run 'ctgjava'.

CTG9502E Service could not allocate memory

Explanation: Upon starting the CICS TG service enough resources could not be allocated.

System action: The CICS TG service terminates.

User response: Free up some memory and try again

CTG9503E Service could not create Java Virtual Machine JNI return code *error label*

Explanation: An error occurred whilst trying to

start the Java Virtual Machine for the Gateway service to run under. The options used when trying to create the JVM will be output when this condition occurs.

System action: The CICS TG service terminates.

User response: Check the options supplied and try again with different parameters

CTG9504I JVM Option:

Explanation: Information message

CTG9505E Cannot find class *name*

Explanation: When attempting to start the CICS TG service the required Java class could not be found. This class can be found in the supplied 'ctgserver.jar'.

System action: The CICS TG service terminates.

User response: Update your classpath and try again.

CTG9506E No main() method found in class com.ibm.ctg.server.JGate

Explanation: When attempting to start the CICS TG service the main() method cannot be found within the JGate class.

System action: The CICS TG service terminates.

User response: Reinstall the CICS Transaction Gateway.

CTG9507E Out of memory

Explanation: Whilst running the CICS TG service no free memory was available for new resources.

System action: The CICS TG service terminates.

User response: If this occurs under expected load then more memory may be needed to satisfy the performance needs for the specific CICS TG installation.

CTG9508E StartServiceControlDispatcher failed

Explanation: An attempt to start the Windows service control dispatcher failed.

System action: The CICS TG service terminates.

User response: Restart the machine and try to start the service again.

CTG9509I Issue ctgservice -? to list command options

Explanation: Information message

CTG9511E Unrecognized option: *option*

Explanation: The option specified is not supported by the ctgservice command line executable.

System action: The usage statement is displayed

User response: Adjust the parameters you pass to ctgservice.exe

CTG9512I CTGSERVICE - CICS Transaction Gateway Windows Service Utility

Explanation: Information message

**CTG9513I (C) Copyright IBM Corporation
Copyright dates . All rights reserved.**

Explanation: Information message

CTG9514I [-A<Param1> [-A<Param2> ...]]

Explanation: Information message

CTG9515I Command options are:

Explanation: Information message

CTG9516I -R Register Gateway daemon parameter overrides

Explanation: Information message

CTG9517I -T Display Gateway daemon parameter overrides

Explanation: Information message

CTG9518I Gateway daemon parameter overrides are: 'overrides '

Explanation: Information message

CTG9520I The following parameter overrides are valid:

Explanation: Information message

CTG9526I Running *INSERT-0*

Explanation: Information message

CTG9527I Stopping *INSERT-0*

Explanation: Information message

CTG9533E **OpenService failed: *error message text***

Explanation: During an uninstall of the CICS TG service the service could not be opened.

System action: The command line application terminates.

User response: Check to see if the service is installed and try again.

CTG9535E *program* Failed to create AppParameters sub-key

Explanation: Whilst attempting to create the specified subkey to be used by the CICS TG in the registry an error occurred.

System action: The command line application terminates.

User response: Restart the machine and try again.

CTG9536E *program* Failed to create AppParameters string

Explanation: Whilst attempting to create the specified string to be used by the CICS TG in the registry an error occurred.

System action: The command line application terminates.

User response: Restart the machine and try again.

CTG9537E *program* Failed to save value of AppParameters

Explanation: Whilst attempting to store the parameters to be used by the CICS TG in the registry an error occurred.

System action: The command line application terminates.

User response: Restart the machine and try again.

CTG9539E **OpenSCManager failed: *error message text***

Explanation: During an install of the CICS TG service the Service Control Manager could not be opened.

System action: The command line application terminates.

User response: Restart the machine and try again.

CTG9540I **Service *INSERT-0* updated**

Explanation: Information message

CTG9541E **Starting service *service name* from the command line is not supported**

Explanation: The named service cannot be started from the command line.

System action: The command line application terminates.

User response: Use the Windows Service Control Manager to start the named service.

CTG9542E **Internal error, Function = *Function*, Error Code = *Error***

Explanation: An internal function has returned an error.

System action: The service failed to complete the requested task.

User response: If the problem persists, contact your service organization.

CTG9544E **Function** *Function failed, Error Code = Error*

Explanation: An internal function has returned an error.

System action: The service failed to complete the requested task.

User response: If the problem persists, contact your service organization.

CCL9547E **Client daemon startup parameter**
'overrides' is not valid

Explanation: An invalid Client daemon startup option was defined for the IBM CICS Transaction Gateway service.

System action: The service fails to start. This message is written to the Windows Application Event log.

User response: Change or remove the invalid parameter value and retry.

CCL9548E **Client daemon failed to start, see**
the Client error log

Explanation: The IBM CICS Transaction Gateway service was unable to start the Client daemon.

System action: The service fails to start. This message is written to the Windows application event log.

User response: Examine the Client daemon error log to determine the cause of the error. The error log is in the location specified in the configuration file, or in the <product_data_path> if there was an error in the configuration.

CTG9549E **Gateway daemon failed to start,**

see the Gateway error log

Explanation: The IBM CICS Transaction Gateway service was unable to start the Gateway daemon.

System action: The service fails to start. This message is written to the Windows application event log.

User response: Examine the Gateway daemon error log to determine the cause of the error. The error log is in the location specified in the configuration file, or in the <product_data_path> if there was an error in the configuration.

CTG9550I **Service** *INSERT-0* **has started**

Explanation: Information message

CTG9551I **Service** *INSERT-0* **has stopped**

Explanation: Information message

CTG9552E **Service** *service name* **failed to start:**
Parameter list *values*

Explanation: A problem occurred when the named service was being started. The *values* are the parameters that were defined for the service.

System action: The service failed to start. This message is written to the Windows application event log.

User response: Inspect the Windows application event log for other messages that help to explain the problem. Address the cause of failure and retry the command.

CTG9553E **Windows function** *function* **failed,**
error code *error*, **Windows message**
details

Explanation: The service process received an unexpected return code from the Windows function.

System action: This message is written to the Windows application event log. The service will terminate.

User response: Inspect the Windows application

event log, the message *details* might indicate the cause of failure, where possible take corrective action. If the problem persists, contact your service organization.

CTG9554W Service *service name* received invalid parameter *value*

Explanation: An invalid value was defined for use with the named service.

System action: The named service fails to start. This message is written to the Windows application event log.

User response: Change or remove the invalid parameter value and retry.

CTG9555E Invalid parameters were defined; service *service name* was not started

Explanation: One or more invalid parameter values were defined for use with the named service.

System action: The named service fails to start. This message is written to the Windows application event log.

User response: Inspect the Windows application event log for messages detailing invalid parameters, and change or remove the invalid parameter values.

CTG9556E Service *service name* could not establish communications with the Client daemon

Explanation: A problem occurred when starting the Client daemon.

System action: The named service ends. This message is written to the Windows application event log.

User response: Check the parameter values that were passed to the Client daemon are valid. To do this issue the `cicscli` command with the parameters that were used. Change any parameters if necessary and restart the service. If parameters are valid inspect the Client error log for other messages.

CTG9557E Function *function* gave an unexpected return value *data*

Explanation: The service process received an unexpected return value from the function.

System action: This message is written to the Windows application event log. The service will terminate.

User response: Inspect the Windows application event log, other messages might indicate the cause of failure, where possible take corrective action. If the problem persists, contact your service organization.

CTG9558I JVM parameter overrides are: *INSERT-0*

Explanation: Information message

CTG9559W The Client daemon is already running

Explanation: The CICS Transaction Gateway service has detected that the Client daemon is already running.

System action: The CICS Transaction Gateway service continues to start, and writes this message to the Windows application event log.

User response: Look in the Windows application event log and in the product logs to find out why the Client daemon was left running. If the problem persists, contact your IBM service organization.

CTG9560E The Client daemon terminated unexpectedly

Explanation: The Client daemon terminated unexpectedly while the CICS Transaction Gateway service was running.

System action: The CICS Transaction Gateway service shuts down immediately, and writes this message to the Windows application event log.

User response: Look in the Client daemon error log to find out why the Client daemon terminated. The location of the error log is defined in the configuration file. If the problem

persists, contact your IBM service organization.

CTG9561E Exception occurred in the Gateway daemon: [error]

Explanation: Where *error* specifies the cause of the error.

System action: System action is not required.

User response: Check error code.

CTG9600I ResultSet not supported

Explanation: Information message

CTG9601E This connection is already closed

Explanation: The connection has already been closed and so cannot be closed again.

System action: System action is not required.

User response: You are attempting to close a connection that has already been closed.

CTG9602I RecordFactory not supported

Explanation: Information message

CTG9603E Connection manager has returned an invalid connection

Explanation: The connection returned by the connection manager is not of the correct type for use with the CICS resource adapters. This is an internal error.

System action: System action is not required.

User response: The CICS Resource Adapter was returned a connection that was either null, or not of the correct type.

CTG9604I This form of execute is not supported

Explanation: Information message

CTG9605E ManagedConnection cannot be used because it is in an unrecoverable state

Explanation: Due to an unrecoverable error the ManagedConnection object that your application is using has become damaged and cannot be used for any further interactions with CICS.

System action: None.

User response: If the problem persists, contact your service organization.

CTG9607E Transaction failed to commit, and was rolled back instead

Explanation: The transaction failed to commit on CICS and so a rollback was carried out to back out any work done within the transaction.

System action: The system tried to commit the transaction but cannot do so. The transaction has therefore been rolled back.

User response: If the problem persists, contact your service organization.

CTG9608E XA transaction in progress on ECIManagedConnection, cannot process local transaction request

Explanation: An ECIManagedConnection can process only one transaction at a time. An XA transaction has already been started on the ECIManagedConnection, so a local transaction may not start until the XA transaction is finished.

System action: System action is not required.

User response: Review the Java Client application that tried to start a second transaction. Refer to the Programming Guide for further details.

CTG9609E Local transaction in progress on ECIManagedConnection, cannot process XA transaction request

Explanation: An ECIManagedConnection can process only one transaction at a time. A local transaction has already been started on the ECIManagedConnection, so an XA transaction may not start until the local transaction is finished.

System action: System action is not required.

User response: Review the Java Client application that tried to start a second transaction. Refer to the Programming Guide for further details.

CTG9610E Local transaction already started on ECIManagedConnection

Explanation: An ECIManagedConnection can process only one transaction at a time. A local transaction has already been started on the ECIManagedConnection, so you may not start another.

System action: System action is not required.

User response: Review the Java Client application that tried to start a second transaction. Refer to the Programming Guide for further details.

CTG9611E Local transaction not started on ECIManagedConnection

Explanation: A local transaction has not been started on the ECIManagedConnection so any calls to commit() or rollback() will fail.

System action: System action is not required.

User response: The local transaction has not been started, so cannot commit.

CTG9612E XA transaction already started on ECIManagedConnection

Explanation: An ECIManagedConnection can process only one transaction at a time. An XA transaction has already been started on the ECIManagedConnection, so you may not start another.

System action: System action is not required.

User response: Review the Java Client application that tried to start a second transaction. Refer to the Programming Guide for further details.

CTG9613E An XAException occurred processing XA request

Explanation: An XAException occurred processing XA request.

System action: System action is not required.

User response: Check the JNI and application server logs for further details.

CTG9614E Incorrect input flag received on XA request

Explanation: An input flag that has been received on an XA_START or XA_END Request is not valid. The request has failed.

System action: System action is not required.

User response: Check the application Server logs for further details.

CTG9615E Transaction not active on ECIManagedConnection

Explanation: The LUW token specified on the end transaction command does not match the LUW token of the active transaction.

System action: The transaction remains active.

User response: If the problem persists, contact your service organization with trace data for the transaction manager and the CICS Transaction Gateway resource adapter.

CTG9616E Transaction failed to commit or roll back

Explanation: A request was made, but failed, possibly due to a communications error.

System action: System action is not required.

User response: Due to an error, this transaction cannot be committed or rolled back.

CTG9617E ECI connection closed

Explanation: Access to this ECICConnection is no longer allowed as it has been closed. A new ECICConnection must be acquired for any further interaction to occur with CICS.

System action: System action is not required.

User response: This connection is closed, no further work through this handle is allowed.

CTG9618E **Cannot associate connection with ECIManagedConnection as supplied Connection is null**

Explanation: When trying to associate a connection with an ECIManagedConnection the connection type must be ECIConnection. This is an internal error.

System action: No system action is performed.

User response: If the problem persists contact your service organization.

CTG9619E **Cannot associate connection with ECIManagedConnection as not of type ECIConnection**

Explanation: When trying to associate a connection with an ECIManagedConnection the connection type must be ECIConnection. This is an internal error.

System action: System action is not required.

User response: The Connection handle is not of the correct type.

CTG9620E **Connection returned by connection manager not of type javax.resource.cci.Connection**

Explanation: The connection manager has not returned an Connection object from the allocateConnection() method.

System action: System action is not required.

User response: The Connection handle returned is not of the correct type.

CTG9621E **ConnectionSpec supplied is not of type ECIConnectionSpec**

Explanation: The ECI resource adapter only supports ConnectionSpec objects of the type ECIConnectionSpec.

System action: System action is not required.

User response: The ConnectionSpec handle is not of the correct type.

CTG9622E **ConnectionRequestInfo object is not of type ECIConnectionRequestInfo**

Explanation: The ConnectionRequestInfo object being supplied to the ECI resource adapter is not of the correct type. This is an internal error.

System action: System action is not required.

User response: If the problem persists contact your service organization.

CTG9623E **Connection manager supplied is null**

Explanation: The connection manager supplied to the ECIConnectionFactory is null. In a managed environment this is a internal error to the environment. In a nonmanaged environment you must not supply a null connection manager to the ECIConnectionFactory constructor.

System action: System action is not required.

User response: The connection manager handle is null.

CTG9624E **ECIInteraction closed**

Explanation: You cannot do any further work with this ECIInteraction object as it has been closed.

System action: System action is not required.

User response: The ECIInteraction has been closed, further work is not allowed.

CTG9625E **ECIConnection used during interaction is null**

Explanation: The ECIConnection object associated with the current ECIInteraction object is null so the interaction cannot continue. This is an internal error.

System action: System action is not required.

User response: The Connection handle is null for this interaction.

CTG9626E Connection associated with ECInteraction not of type ECConnection

Explanation: The ECConnection that is being used by your ECInteraction is not of the correct type. This is an internal error.

System action: System action is not required.

User response: The Connection handle for this interaction is not of the correct type.

CTG9627E IOException occurred when writing to the Output Record

Explanation: An I/O exception was thrown by the record when the resource adapter attempted to populate it with the relevant information. The record exception is linked to this one.

System action: None.

User response: Review the linked exception for an indication of any corrective action needed.

CTG9628E InteractionSpec passed to execute() not of type ECInteractionSpec

Explanation: Only InteractionSpec objects of the type ECInteractionSpec can be passed to the ECInteraction.execute() method.

System action: System action is not required.

User response: InteractionSpec passed in to the execute method is not of the correct type.

CTG9629E InteractionSpec passed to execute() is null

Explanation: When calling ECInteraction.execute() a non null ECInteractionSpec must be supplied at all times.

System action: System action is not required.

User response: Null was passed to the ECInteractionSpec.execute method.

CTG9630E IOException occurred in communication with CICS

Explanation: An IOException occurred during an interaction with a CICS server. The CICS exception is linked to this one.

System action: None.

User response: Review the linked exception for an indication of any corrective action needed.

CTG9631E Error occurred during interaction with CICS

Explanation: During an interaction with a CICS server a known error has occurred in the underlying CICS Transaction Gateway runtime. The supplied error code can be referenced in the CICS Transaction Gateway documentation. This is an internal error.

System action: None.

User response: The supplied error code can be referenced in the CICS Transaction Gateway documentation.

CTG9632E SYNC_SEND/ SYNC_SEND_RECEIVE not supported when reply to previous SYNC_SEND is pending

Explanation: A reply to a SYNC_SEND interaction is pending. No further SYNC_SEND or SYNC_SEND_RECEIVE interactions are allowed until a SYNC_RECEIVE is executed.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9633E SYNC_RECEIVE attempted without corresponding SYNC_SEND

Explanation: A SYNC_SEND interaction must be executed before a reply from CICS can be received using SYNC_RECEIVE.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9634E Input record does not implement the Streamable interface

Explanation: Input record objects supplied to the ECIInteraction.execute() method must support the javax.resource.cci.Streamable interface.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9635E Input record is null

Explanation: When specifying an InteractionVerb of SYNC_SEND or SYNC_SEND_RECEIVE a non null input Record object must be supplied as input to CICS.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9636E Output record does not implement the Streamable interface

Explanation: Output record objects supplied to the ECIInteraction.execute() method must support the javax.resource.cci.Streamable interface.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector

Architecture" section of the Programming Guide for further details.

CTG9637E Output record is null

Explanation: When specifying an InteractionVerb of SYNC_RECEIVE or SYNC_SEND_RECEIVE a non null output Record object must be supplied to hold the reply from CICS.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9638E Transaction Abend occurred in CICS: Abend Code=

Explanation: During the Interaction with CICS a Transaction Abend occurred on the server. The supplied code is that which is returned by CICS.

System action: None.

User response: Check the abend code in the CICS documentation for an indication of any corrective action needed.

CTG9639E Unable to perform action, connection is closed

Explanation: The EPICConnection has been closed and can no longer be used.

System action: System action is not required.

User response: Connection is closed, no further work possible.

CTG9640E No connection was returned by connection manager

Explanation: The connection manager did not return an EPICConnection for use by the application.

System action: System action is not required.

User response: No connection returned by the connection manager.

CTG9641E Invalid connection was returned by connection manager

Explanation: An unusable Connection object was returned by the connection manager.

System action: System action is not required.

User response: Invalid connection was returned by connection manager.

CTG9642E Invalid ConnectionSpec was supplied, must be an EPIConnectionSpec instance

Explanation: Only ConnectionSpecs of the type EPIConnectionSpec are accepted in the getConnection() method of EPIConnectionFactory.

System action: System action is not required.

User response: Invalid ConnectionSpec was supplied, must be an EPIConnectionSpec instance.

CTG9643E Unable to perform action, interaction is closed

Explanation: The EPIInteraction is closed so no further work can be carried out on it.

System action: System action is not required.

User response: Connection is closed, no further work possible.

CTG9644E No EPIInteractionSpec supplied

Explanation: An EPIInteractionSpec must be supplied when EPIConnection.execute() is called.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9645E Invalid InteractionSpec was supplied, must be an EPIInteractionSpec instance

Explanation: An EPIInteractionSpec must be supplied when EPIConnection.execute() is called.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9646E Invalid connection passed to associate connection method by application server

Explanation: When attempting to associate a Connection object with an EPIManagedConnection the incorrect type was used. The EPI resource adapter supports only EPIConnection objects being passed to the associateConnection() method.

System action: None.

User response: If the problem persists, contact your service organization.

CTG9647E A null connection was passed to associate connection method by application server

Explanation: When attempting to associate a Connection object with an EPIManagedConnection the incorrect type was used. The EPI resource adapter only supports EPIConnection objects being passed to the associateConnection() method.

System action: System action is not required.

User response: A null connection was passed to associateConnection() method by the application server.

CTG9648I Local transactions not supported

Explanation: Information message

CTG9649I XAResource not supported

Explanation: Information message

CTG9650E Streamable input record information does not contain the correct amount of data

Explanation: The amount of data stored in the input Record object does not correspond to the amount required for the currently defined screen size.

System action: System action is not required.

User response: Streamable input record information does not contain the correct amount of data.

CTG9651E Input record is of unknown type and cannot be used

Explanation: Input record objects supplied to EPIInteraction.execute() must implement the javax.resource.cci.Streamable interface in order to be used by the EPI resource adapter.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9652E Output record is of unknown type and cannot be used

Explanation: Output record objects supplied to EPIInteraction.execute() must implement the javax.resource.cci.Streamable interface in order to be used by the EPI resource adapter.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9653E No output record was given and

one was expected

Explanation: When executing a SYNC_RECEIVE or SYNC_SEND_RECEIVE an output record must be supplied to hold the reply from CICS.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9654E No input record was given and one was expected

Explanation: When executing a SYNC_SEND or SYNC_SEND_RECEIVE an input record must be supplied to hold the data to be sent to CICS.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9655E LogonLogoff class threw an exception which is linked to this one

Explanation: An exception was thrown while the LogonLogoff class specified was being accessed. The original exception can be accessed using the getCause method.

System action: None.

User response: Review the linked exception for an indication of any corrective action needed.

CTG9656E LogonLogoff class could not be found

Explanation: The LogonLogoff class specified could not be found in the current CLASSPATH. Check that any required files are in your systems CLASSPATH.

System action: None.

User response: Check that the LogonLogoff class specified is on the system CLASSPATH.

CTG9657E LogonLogoff class found but could not be instantiated

Explanation: An error occurred whilst trying to instantiate the LogonLogoff class specified.

System action: None.

User response: Check that the name of the LogonLogoff class is specified correctly in the Java Client application.

CTG9658E No authority to create LogonLogoff Class

Explanation: The Java Client application does not have the required authorization to create the specified LogonLogoff class.

System action: None.

User response: Refer to the "Writing LogonLogoff classes" section of the Programming Guide for further details on how to grant Java security permissions.

CTG9659E Unable to build security information to pass to LogonLogoff class

Explanation: The resource adapter was unable to create a Subject object required to be passed to the registered LogonLogoff class. This could be because the required Java Security permissions are not available.

System action: None.

User response: Refer to the "Writing LogonLogoff classes" section of the Programming Guide for further details on how to grant Java security permissions.

CTG9660E Adapter unable to function as no Connection Manager passed to resource adapter

Explanation: A Request to create a connection factory has failed because a null connection manager was passed to the

createConnectionFactory method.

System action: None.

User response: If the problem persists, contact your service organization.

CTG9661I EPI is not transactional, so this action is not supported

Explanation: Information message

CTG9662I You can only have one interaction per connection

Explanation: Information message

CTG9663E A connection which is not the active connection attempted to run on the resource adapter

Explanation: The EPICconnection that your component is trying to use is not the current Connection being processed by the EPI resource adapter. The current working Connection must be closed before work can continue. This is an internal error.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9664E Transaction ended, but unable to determine server state

Explanation: The CICS transaction has ended, but it is not possible to determine if resources were committed or rolled back.

System action: None.

User response: If the problem persists, contact your service organization.

CTG9665E ExecuteTimeout property cannot be negative

Explanation: The ExecuteTimeout property on

the ECInteractionSpec must be a positive value.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9666E Value of InteractionVerb is not one of: SYNC_SEND, SYNC_RECEIVE, SYNC_SEND_RECEIVE

Explanation: A valid value for InteractionVerb was not passed within the ECInteractionSpec.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9667E FunctionName is empty, cannot send request

Explanation: In order to carry out an Interaction with CICS a FunctionName must be provided in the ECInteractionSpec. This FunctionName maps to the program that is to be executed on the CICS server.

System action: System action is not required.

User response: The function name in the ECInteractionSpec is empty, request not valid.

CTG9668E XA transaction in progress but no XID is available

Explanation: An ECI request is being executed as part of an XA transaction but no Xid is available on the associated XAResource object. The transaction is not in a valid state.

System action: System action is not required.

User response: Check the application server logs for further details.

CTG9669E XA transaction not started on ECIManagedConnection

Explanation: An attempt was made to end an XA transaction which had not been started.

System action: System action is not required.

User response: Check the application server logs for further details.

CTG9670E Incorrect XID received

Explanation: The Xid received on an XA_END flow is not the same as that received on the XA_START flow. The XA_END flow has failed.

System action: System action is not required.

User response: Check the application server logs for further details.

CTG9671E Screenable input record does not match the current screen definition

Explanation: The input record provided does not match the EPI resource adapter's representation of the screen.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9672E Transaction cannot begin - connection is closed

Explanation: A LocalTransaction cannot be started because it was obtained from a Connection which is now closed.

System action: System action is not required

User response: Check that begin() is not being invoked after the connection is closed.

CTG9673E SYNC_RECEIVE requires COMMAREA length, reply length or both to be defined

Explanation: A SYNC_RECEIVE interaction requires the COMMAREA length or reply length to be specified in the InteractionSpec, and neither have been specified.

System action: None.

User response: Review the Java Client application to resolve the problem. Refer to the "Programming using the J2EE Connector Architecture" section of the Programming Guide for further details.

CTG9674E No SSL key ring was provided for SSL protocol

Explanation: The SSL protocol was specified for the connectionURL, but no key ring has been specified.

System action: System action is not required.

User response: No SSL key ring was provided for SSL protocol.

CTG9675E EPI resource adapter failed trying to populate Screenable record

Explanation: An exception was thrown by the screenable record when the EPI resource adapter tried to populate it with the relevant information. The record exception is linked to this one.

System action: None.

User response: Review the linked exception for an indication of any corrective action needed.

CTG9676E IOException occurred when reading the Input Record

Explanation: An I/O exception was thrown by the record when the resource adapter attempted to read the information. The record exception is linked to this one.

System action: None.

User response: Review the linked exception for an indication of any corrective action needed.

CTG9677E Local transactions cannot be used

Explanation: An application has attempted to obtain a local transaction from a connection which cannot be provided. This will result from using the ECI resource adapter configured for local: connections on WebSphere for z/OS.

System action: A NotSupportedException was thrown back to the application.

User response: The application needs to be changed to not use local transactions obtained from the connection instance, or the resource adapter needs to be configured for remote connections to a CICS TG daemon.

CTG9678E EPI resource adapter failed trying to read streamed EPI data, linked exception has details

Explanation: During a read of the terminal, an exception has occurred

System action: The terminal read could not continue.

User response: Retrieve the linked exception from the CICSUserInputException and diagnose the error from the original message.

CTG9679E EPI resource adapter failed trying to write streamed EPI data

Explanation: During a write to the terminal an exception has occurred. The original exception has been linked.

System action: The terminal write could not continue.

User response: Retrieve the linked exception from the CICSUserInputException and diagnose the error from the original message.

CTG9680E Connected to a CICS TG that does not support XA transactions

Explanation: The application has attempted to execute an XA transaction using a CICS TG that does not support XA transactions.

System action: System action is not required.

User response: Either re-configure the application to connect to a CICS TG that

supports XA transactions, or do not use XA transactions within this application.

CTG9681E Connected to a CICS TG that is not enabled to support XA transactions

Explanation: The application has attempted to execute an XA transaction using a CICS TG that does not have XA transaction support enabled.

System action: System action is not required.

User response: Enable support for XA transactions within the CICS TG, or re-configure to connect to a CICS TG that has XA transaction support enabled, or do not use XA transactions with this application.

CTG9683E Mapped Output Record is not an ECICChannelRecord

Explanation: The Output MappedRecord object that has been passed to the Resource Adapter is not of the correct type. It cannot be treated as a Channel object because it is not of type ECICChannelRecord.

System action: The request is rejected.

User response: Check the type of the input record.

CTG9684E XA error 'XA return code ' occurred processing XA request type 'request type '

Explanation: CICS TG returned error code *XA return code* for an XA request type *request type* .

System action: This message is logged. The application server makes the appropriate response depending on the request type and error code. The application server might roll back a transaction if an error occurs.

User response: Check the CICS TG and application server logs for further information about the cause of the error.

CTG9685E Heuristic XA error 'XA return code ' occurred processing XA request

type 'request type '

Explanation: The CICS TG returned heuristic error code *XA return code* for an XA request type *request type* .

System action: This message is logged. The application server makes the appropriate response depending on the request type and error code.

User response: Check the CICS TG and application server logs for further information about the cause of the error. Check that all units of recovery associated with the Gateway resource manager in RRS and the application server are in a completed state. See the Gateway daemon information center for information about the resource manager name used by Gateway daemons.

CTG9700E Binding operation *operation* is not in the specified WSDL

Explanation: The specified WSDL does not contain binding operation *operation* .

System action: The WSIFOperation object has not been created.

User response: Ensure that the operation name specified on the createOperation call matches the name of a binding operation contained in the WSDL.

CTG9701E ECI Binding operation *operation* is not in the specified WSDL

Explanation: Binding operation *operation* does not contain an extensibility element for this resource adapter.

System action: The WSIFOperation object has not been created.

User response: Ensure that binding operation *operation* contains an extensibility element for this resource adapter.

CTG9702E There is a duplicate extensibility element *name* in *operation*

Explanation: Binding operation *operation*

contains multiple extensibility elements with name *name* .

System action: The WSIFOperation object has not been created.

User response: Ensure that all extensibility elements in binding operation *operation* have unique names.

CTG9703E No matching part was found for InteractionSpec property *name*

Explanation: The InteractionSpec does not contain a part for property *name* .

System action: The Interation fails to execute.

User response: Ensure that the resource adapter has been deployed correctly. If the problem persists, contact your service organization.

CTG9704E An exception occurred while updating InteractionSpec property *name* with message *message*

Explanation: A call to update InteractionSpec property *name* threw an exception with message *message* .

System action: The Interation fails to execute.

User response: Inspect the text of message *message* to identify the cause of the exception. Ensure that the resource adapter has been deployed correctly. If the problem persists, contact your service organization.

CTG9705E No matching part was found for ConnectionSpec property *name*

Explanation: The ConnectionSpec does not contain a part for property *name* .

System action: The ConnectionSpec has not been created.

User response: Ensure that the resource adapter has been deployed correctly. If the problem persists, contact your service organization.

CTG9706E An exception occurred while updating ConnectionSpec property *name* with message *message*

Explanation: A call to update ConnectionSpec property *name* threw an exception with message *message* .

System action: The ConnectionSpec has not been created.

User response: Inspect the text of message *message* to identify the cause of the exception. Ensure that the resource adapter has been deployed correctly. If the problem persists, contact your service organization.

CTG9710E Binding operation *operation* is not in the specified WSDL

Explanation: The specified WSDL does not contain binding operation *operation* .

System action: The WSIFOperation object has not been created.

User response: Ensure that the operation name specified on the createOperation call matches the name of a binding operation contained in the WSDL.

CTG9711E EPI Binding operation *operation* is not in the specified WSDL

Explanation: Binding operation *operation* does not contain an extensibility element for this resource adapter.

System action: The WSIFOperation object has not been created.

User response: Ensure that binding operation *operation* contains an extensibility element for this resource adapter.

CTG9712E There is a duplicate extensibility element *name* in operation

Explanation: Binding operation *operation* contains multiple extensibility elements with name *name* .

System action: The WSIFOperation object has not been created.

User response: Ensure that all extensibility elements in binding operation *operation* have unique names.

CTG9713E No matching part was found for InteractionSpec property *name*

Explanation: The InteractionSpec does not contain a part for property *name* .

System action: The Interaction fails to execute.

User response: Ensure that the resource adapter has been deployed correctly. If the problem persists, contact your service organization.

CTG9714E An exception occurred while updating InteractionSpec property *name* with message *message*

Explanation: A call to update InteractionSpec property *name* threw an exception with message *message* .

System action: The Interaction fails to execute.

User response: Inspect the text of message *message* to identify the cause of the exception. Ensure that the resource adapter has been deployed correctly. If the problem persists, contact your service organization.

CTG9715E No matching part was found for ConnectionSpec property *name*

Explanation: The ConnectionSpec does not contain a part for property *name* .

System action: The ConnectionSpec has not been created.

User response: Ensure that the resource adapter has been deployed correctly. If the problem persists, contact your service organization.

CTG9716E An exception occurred while updating ConnectionSpec property *name* with message *message*

Explanation: A call to update ConnectionSpec property *name* threw an exception with message *message* .

System action: The ConnectionSpec has not been created.

User response: Inspect the text of message *message* to identify the cause of the exception. Ensure that the resource adapter has been deployed correctly. If the problem persists, contact your service organization.

CTG9717E An exception occurred while updating part *part* from InteractionSpec property *name* with message *message*

Explanation: A call to update part *part* from InteractionSpec property *name* threw an exception with message *message* .

System action: The Interaction fails to execute.

User response: Inspect the text of message *message* to identify the cause of the exception. Ensure that the resource adapter has been deployed correctly. If the problem persists, contact your service organization.

CTG9800E No message for id *ID* could be located

Explanation: The message id indicated could not be located within the message file.

System action: No action taken.

User response: Contact your service organization with the message id.

CTG9801E A request to get or set an attribute was unknown to MBean *name*

Explanation: A Management Bean received a request to get or set an attribute but this attribute is unknown to it.

System action: No action taken.

User response: Ensure the management bean being invoked supports this attribute.

CTG9802E An invalid value was given for an attribute for MBean *name*

Explanation: A request to set an attribute on a management bean failed because the value provided was not valid.

System action: The attribute doesn't change.

User response: Ensure the value provided is valid for the attribute.

CTG9803E An unexpected exception was received from MBean *name*

Explanation: A management bean throw an exception which was not expected by the system.

System action: No action taken.

User response: The linked exception may provide more information on the problem.

CTG9804E An unknown exception occurred whilst trying to set the trace file to *filename*

Explanation: An exception occurred whilst trying to set the Gateway trace file to the specified file. This might be because the specified file is a directory, is not writeable or cannot be opened for another reason.

System action: The Gateway trace file is not changed.

User response: Check that the file specified is a valid file and is in a writeable location.

CTG9805E An unknown exception occurred whilst trying to set the JNI trace file to *filename*

Explanation: An exception occurred whilst trying to set the JNI trace file to the specified file. This might be because the specified file is a directory, is not writeable or cannot be opened for another reason.

System action: The JNI trace file is not changed.

User response: Check that the file specified is a valid file and is in a writeable location.

CTG9806E An error occurred when setting *value*

Explanation: When trying to make multiple changes to the Gateway daemon *value* failed.

System action: The Gateway daemon settings are not changed.

User response: Check that the parameters used in the change are correct.

CTG9807E An error occurred setting attribute *requested value* . A further error occurred while attempting to roll back attribute *associated value* .

Explanation: When trying to make multiple changes to the Gateway daemon the attempt to set attribute *requested value* failed. When trying to undo the changes already made, the roll back of attribute *associated value* failed.

System action: The Gateway daemon settings are left in an undefined state.

User response: Check that the parameters used in the change are correct.

CTG9808E The value of *setting* is not valid for *value*

Explanation: When trying to change a Gateway trace setting, the value provided was not valid.

System action: The Gateway daemon setting is not changed.

User response: Check that the value used for the setting is correct.

CTG9809E The value of *setting* is not valid for *value*

Explanation: When trying to change a Gateway JNI trace setting, the value provided was not valid.

System action: The Gateway daemon setting is not changed.

User response: Check that the value used for the setting is correct.

CTG9810E **The trace file *filename* specified a directory**

Explanation: When trying to set the Gateway trace file, the value specified a directory.

System action: The trace file is not changed.

User response: Specify a trace file that is not a directory.

CTG9811E **The trace file *filename* specified an invalid path**

Explanation: When trying to set the Gateway trace file, the value specified an invalid path.

System action: The trace file is not changed.

User response: Specify a trace file using a valid path for the platform.

CTG9812E **The trace file *filename* specified an unwritable file**

Explanation: When trying to set the Gateway trace file, the value specified a file on which the Gateway daemon does not have write permissions, or a directory in which the Gateway daemon cannot create or write a file.

System action: The trace file is not changed.

User response: Specify a trace file which is writeable, or in a location which is writeable, by the Gateway daemon.

CTG9814E **The JNI trace file *filename* specified an invalid path**

Explanation: When trying to set the JNI trace file, the value specified an invalid path.

System action: The JNI trace file is not changed.

User response: Specify a JNI trace file using a valid path for the platform.

CTG9815E **The JNI trace file *filename* specified an unwritable file**

Explanation: When trying to set the JNI trace file, the value specified a file on The CICS

Transaction Gateway does not have write permissions to the specified JNI trace file, or the CICS TG does not have permission to create, or write, a file in the specified directory.

System action: The JNI trace file is not changed.

User response: Specify a trace file which can be written to by the CICS Transaction Gateway.

CTG9816E **Attempt to change JNI trace filename while trace running**

Explanation: The user attempted to change the JNI trace filename whilst trace was running.

System action: This message is output. Tracing continues to the previous filename.

User response: Deactivate JNI trace before changing the filename.

Product library and related literature

The CICS Transaction Gateway product library contains information on administration, messages and programming; this information is available in this information center, and is also available in PDF form. IBM Redbooks® publications provide a further source of information about working with CICS Transaction Gateway.

CICS Transaction Gateway books

The books in the library cover administration, programming and messages.

- *CICS Transaction Gateway: Messages, SC34-7061-02* describes the error messages that can be generated by the CICS Transaction Gateway.

Sample configuration documents

Several sample configuration documents are available in portable document format (PDF).

These documents give step-by-step guidance for configuring CICS Transaction Gateway for communication with CICS servers, using various protocols. They provide detailed instructions that extend the information in the CICS Transaction Gateway library.

Visit the following Web site:

www.ibm.com/software/cics/ctg

and follow the **Library** link.

IBM Redbooks publications

IBM Redbook titles are available on a wide range of subjects relevant to CICS Transaction Gateway programming, installation, operation and troubleshooting.

The following International Technical Support Organization (ITSO) Redbook publication contains many examples of client/server configurations:

- *CICS Transaction Gateway V5 - The WebSphere Connector for CICS, SG24-6133* describes how to use the different protocols (TCP/IP, TCP62, APPC and EXCI) for communication with CICS, and how to securely connect a Java client application to a CICS region.

- *Revealed! Architecting Web Access to CICS*, SG24-5466 is intended for IT architects who select, plan, and design SOA solutions that make use of CICS assets
- *Enterprise JavaBeans for z/OS and z/OS CICS Transaction Server V2.2*, SG24-6284 describes the EJB and the way it has been implemented within the CICS architecture, also describes how to set up and configure a CICS region to support EJBs
- *Java Connectors for CICS: Featuring the J2EE Connector Architecture*, SG24-6401 provides information on developing J2EE applications.
- *Systems Programmer's Guide to Resource Recovery Services (RRS)*, SG24-6980-00 describes how to use RRS in various scenarios.
- *Communications Server for z/OS V1R2 TCP/IP Implementation Guide*, SG24-6517-00 provides information on using Communications Server for z/OS V1R2, including load balancing.
- *Redpaper: Transactions in J2EE*, REDP-3659-00 discusses transactions in the J2EE environment, including one-phase commit and two-phase commit XA transactions.
- *Exploring Systems Monitoring for CICS Transaction Gateway V7.1 for z/OS*, SG24-7562-00 looks at product installation and customization, and also covers systems monitoring for CICS Transaction Gateway using IBM Tivoli® OMEGAMON® XE, and statistics provided by CICS Performance Analyzer

The ITSO Redbooks are available from various sources. For the latest information, see:

www.ibm.com/redbooks/

Other useful information

Other sources of useful information include the CICS Transaction Server information center and associated publications.

The CICS Transaction Server for z/OS® V4.1 information center is located at:

<http://publib.boulder.ibm.com/infocenter/cicsts/v4r1/index.jsp>

CICS Transaction Server publications

The CICS Transaction Server books on security, inter-product communication and problem determination also provide a useful source of information.

CICS Transaction Server for z/OS RACF Security Guide, SC34-7003

CICS inter-product communication

The following books describe the intercommunication facilities of the CICS server products:

- *CICS Family: Interproduct Communication*, SC34-6853
- *CICS Transaction Server for z/OS CICS External Interfaces Guide*, SC34-7019
- *CICS Transaction Server for z/OS: Intercommunication Guide*, SC34-7018
- *CICS TS for VSE: Intercommunication Guide*, SC33-0701
- *CICS Transaction Server for iSeries: Intercommunication*, SC41-5456

The first book above is a CICS family book containing a platform-independent overview of CICS inter-product communication.

CICS problem determination

The following books describe the problem determination facilities of the CICS server products:

- *Transaction Server for Windows Problem Determination*, GC34-6210
- *CICS Transaction Server for z/OS Problem Determination Guide*, SC34-7034
- *CICS TS for VSE 2.3 Problem Determination Guide*, SC33-0716
- *CICS Transaction Server for iSeries: Problem Determination*, SC41-5453
- *TXSeries for Multiplatforms: Problem Determination Guide*, SC34-6636

Accessibility

Accessibility features help users with a physical disability, for example restricted mobility or limited vision, to use information technology products successfully. CICS Transaction Gateway provides accessibility by enabling keyboard-only operation.

For more information about the IBM commitment to accessibility, visit the [IBM Accessibility Center](#).

Glossary

This glossary defines the terms and abbreviations used in CICS Transaction Gateway and in the information centers.

A

abnormal end of task (abend)

The termination of a task, job, or subsystem because of an error condition that recovery facilities cannot resolve.

Advanced program-to-program communication (APPC)

An implementation of the SNA/SDLC LU 6.2 protocol that allows interconnected systems to communicate and share the processing of programs. The Client daemon uses APPC to communicate with CICS server systems.

APAR See *Authorized program analysis report*.

API See *application programming interface*.

APPC See *Advanced program-to-program communication*.

application programming interface (API)

A functional interface that allows an application program that is written in a high-level language to use specific data or functions of the operating system or another program.

APPLID

1. On CICS Transaction Gateway: The application identifier that is used to identify CICS Transaction Gateway connections on the CICS server and tasks in a CICSplex. See also *APPLID qualifier* and *fully-qualified APPLID*.
2. On CICS Transaction Server: The name by which a CICS system is known in a network of interconnected CICS systems. CICS Transaction Gateway application identifiers do not need to be defined in SYS1.VTAMLST. The CICS APPLID is specified in the APPLID system initialization parameter.

APPLID qualifier

Optionally used as a high-level qualifier for the APPLID to form a fully-qualified APPLID. See also *APPLID* and *fully-qualified APPLID*.

ARM See *automatic restart manager*.

Authorized program analysis report (APAR)

A request for correction of a defect in a current release of an IBM-supplied program.

ATI See *automatic transaction initiation*.

attach In SNA, the request unit that flows on a session to initiate a conversation.

Attach Manager

The component of APPC that matches attaches received from remote computers to accepts issued by local programs.

autoinstall

A method of creating and installing resources dynamically as terminals log on, and deleting them at logoff.

automatic restart manager (ARM)

A z/OS recovery function that can improve the availability of specific batch jobs or started tasks, and therefore result in faster resumption of productive work.

automatic transaction initiation (ATI)

The initiation of a CICS transaction by an internally generated request, for example, the issue of an EXEC CICS START command or the reaching of a transient data trigger level. CICS resource definition can associate a trigger level and a transaction with a transient data destination. When the number of records written to the destination reaches the trigger level, the specified transaction is automatically initiated.

B

bean A definition or instance of a JavaBeans component. See also *JavaBeans*.

bean-managed transaction

A transaction where the J2EE bean itself is responsible for administering transaction tasks such as committal or rollback. See also *container-managed transaction*.

BIND command

In SNA, a request to activate a session between two logical units (LUs).

business logic

The part of a distributed application that is concerned with the application logic rather than the user interface of the application. Compare with *presentation logic*.

C

CA See *certificate authority*.

callback

A way for one thread to notify another application thread that an event has happened.

certificate authority (CA)

In computer security, an organization that issues certificates. The certificate authority authenticates the certificate owner's identity and the services that the owner is authorized to use. It issues new certificates and revokes certificates from users who are no longer authorized to use them.

change-number-of-sessions (CNOS)

An internal transaction program that regulates the number of parallel sessions between the partner LUs with specific characteristics.

channel

A channel is a set of containers, grouped together to pass data to CICS. There is no limit to the number of containers that can be added to a channel, and the size of individual containers is limited only by the amount of storage that you have available.

CICS connectivity components

A generic reference to the Client daemon, EXCI, and the IPIC protocol.

CICS connectivity components

The Client daemon, the EXCI (External CICS Interface), and the IPIC (IP Interconnectivity) protocol are collectively called the 'CICS connectivity components'. The Client daemon handles the TCP/IP and the SNA protocols.

CICS Request Exit

An exit that is invoked by the CICS Transaction Gateway for z/OS at run time to determine which CICS server to use.

CICS server name

A defined server known to CICS Transaction Gateway.

CICS TS

Abbreviation of CICS Transaction Server.

class

In object-oriented programming, a model or template that can be instantiated to create objects with a common definition and therefore, common properties, operations, and behavior. An object is an instance of a class.

CLASSPATH

In the execution environment, an environment variable keyword that specifies the directories in which to look for class and resource files.

Client API

The Client API is the interface used by Client applications to interact with CICS using the Client daemon. See External Call Interface, External Presentation Interface, and External Security Interface.

Client application

The client application is a user application written in a supported programming language that uses one or more of the CICS Transaction Gateways APIs.

Client daemon

The Client daemon manages TCP/IP, SNA, and, on Windows, named pipe connections to CICS servers on UNIX, Linux, and Windows. It processes ECI, EPI, and ESI requests, sending and receiving the appropriate flows to and from the CICS server to satisfy Client application requests. It can support concurrent requests to one or more CICS servers. The CICS Transaction Gateway initialization file defines the operation of the Client daemon and the servers and protocols used for communication.

client/server

Pertaining to the model of interaction in distributed data processing in which a program on one computer sends a request to a program on another computer and awaits a response. The requesting program is called a client; the answering program is called a server.

CNOS See *Change-Number-of-Sessions*.

code page

An assignment of hexadecimal identifiers (code points) to graphic characters. Within a given code page, a code point can have only one meaning.

color mapping file

A file that is used to customize the 3270 screen color attributes on client workstations.

COMMAREA

See *communication area*.

commit phase

The second phase in a XA process. If all participants acknowledge that they are prepared to commit, the transaction manager issues the commit request. If any participant is not prepared to commit the transaction manager issues a back-out request to all participants.

communication area (COMMAREA)

A communication area that is used for passing data both between programs within a transaction and between transactions.

configuration file

A file that specifies the characteristics of a program, system device, server or network.

connection

In data communication, an association established between functional units for conveying information.

In Open Systems Interconnection architecture, an association established by a given layer between two or more entities of the next higher layer for the purpose of data transfer.

In TCP/IP, the path between two protocol application that provides reliable data stream delivery service.

In Internet, a connection extends from a TCP application on one system to a TCP application on another system.

container

A container is a named block of data designed for passing information between programs. A container is a "named COMMAREA" that is not limited to 32KB. Containers are grouped together in sets called channels.

container-managed transaction

A transaction where the EJB container is responsible for administration of tasks such as committal or rollback. See also *bean-managed transaction*.

control table

In CICS, a storage area used to describe or define the configuration or operation of the system.

conversation

A connection between two programs over a session that allows them to communicate with each other while processing a transaction.

conversation security

In APPC, a process that allows validation of a user ID or group ID and password before establishing a connection.

D**daemon**

A program that runs unattended to perform continuous or periodic systemwide functions, such as network control. A daemon can be launched automatically, such as when the operating system is started, or manually.

data link control (DLC)

A set of rules used by nodes on a data link (such as an SDLC link or a token ring) to accomplish an orderly exchange of information.

DBCS See *double-byte character set*.

default CICS server

The CICS server that is used if a server name is not specified on an ECI, EPI, or ESI request. The default CICS server name is defined as a product wide setting in the configuration file (ctg.ini).

dependent logical unit

A logical unit that requires assistance from a system services control point (SSCP) to instantiate an LU-to-LU session.

deprecated

Pertaining to an entity, such as a programming element or feature, that is supported but no longer recommended, and that might become obsolete.

digital certificate

An electronic document used to identify an individual, server, company, or some other entity, and to associate a public key with the entity. A digital certificate is issued by a certificate authority and is digitally signed by that authority.

digital signature

Information that is encrypted with an entity's private key and is appended to a message to assure the recipient of the authenticity and integrity of the message. The digital signature proves that the message was signed by the entity that owns, or has access to, the private key or shared secret symmetric key.

distinguished name

The name that uniquely identifies an entry in a directory. A distinguished name is made up of attribute:value pairs, separated by commas. The format of a distinguished name is defined by RFC4514. For more information, see <http://www.ietf.org/rfc/rfc4514.txt>. See also *realm name* and *identity propagation*.

distributed application

An application for which the component application programs are distributed between two or more interconnected processors.

distributed identity

User identity information that originates from a remote system. The distributed identity is created in one system and is passed to one or more other systems over a network. See also *distinguished name* and *realm name*.

distributed processing

The processing of different parts of the same application in different systems, on one or more processors.

distributed program link (DPL)

A link that enables an application program running on one CICS system to link to another application program running in another CICS system.

DLC See *data link control*.

DLL See *dynamic link library*.

domain

In the Internet, a part of a naming hierarchy in which the domain name consists of a sequence of names (labels) separated by periods (dots).

domain name

In TCP/IP, a name of a host system in a network.

domain name server

In TCP/IP, a server program that supplies name-to-address translation by mapping domain names to IP addresses. Synonymous with name server.

dotted decimal notation

The syntactical representation for a 32-bit integer that consists of four 8-bit numbers written in base 10 with periods (dots) separating them. It is used to represent IP addresses.

double-byte character set (DBCS)

A set of characters in which each character is represented by 2 bytes. Languages such as Japanese, Chinese and Korean, which contain more symbols than can be represented by 256 code points, require double-byte character sets. Because each character requires 2 bytes, the typing, display, and printing of DBCS characters requires hardware and programs that support DBCS. Contrast with *single-byte character set*.

DPL See *distributed program link*.

dynamic link library (DLL)

A collection of runtime routines made available to applications as required.

dynamic server selection

The mapping of a logical server name to an actual server name at run time using logical server definitions, default CICS server definitions, or custom function in a CICS request exit. Previously known as *server name remapping*.

E

EBCDIC

See *extended binary-coded decimal interchange code*.

ECI See *external call interface*.

EJB See *Enterprise JavaBeans*.

emulation program

A program that allows a host system to communicate with a workstation in the same way as it would with the emulated terminal.

emulator

A program that causes a computer to act as a workstation attached to another system.

encryption

The process of transforming data into an unintelligible form in such a way that the original data can be obtained only by using a decryption process.

enterprise bean

A Java component that can be combined with other resources to create J2EE applications. There are three types of enterprise beans: entity beans, session beans, and message-driven beans.

Enterprise JavaBeans (EJB)

A component architecture defined by Sun Microsystems for the development and deployment of object-oriented, distributed, enterprise-level applications (J2EE).

environment variable

A variable that specifies the operating environment for a process. For example, environment variables can describe the home directory, the command search path, the terminal in use, and the current time zone.

EPI See *external presentation interface*.

ESI See *external security interface*.

Ethernet

A local area network that allows multiple stations to access the transmission medium at will without prior coordination, avoids contention by using carrier sense and deference, and resolves contention by using collision detection and transmission. Ethernet uses carrier sense multiple access with collision detection (CSMA/CD).

EXCI See *external CICS interface*.

extended binary-coded decimal interchange code (EBCDIC)

A coded character set of 256 8-bit characters developed for the representation of textual data.

extended logical unit of work (extended LUW)

A logical unit of work that is extended across successive ECI requests to the same CICS server.

external call interface (ECI)

A facility that allows a non-CICS program to run a CICS program. Data is exchanged in a COMMAREA or a channel as for usual CICS interprogram communication.

external CICS interface (EXCI)

An MVS application programming interface provided by CICS Transaction Server for z/OS that enables a non-CICS program to call a CICS program and to pass and receive data using a COMMAREA. The CICS application program is started as if linked-to by another CICS application program.

external presentation interface (EPI)

A facility that allows a non-CICS program to appear to CICS as one or more standard 3270 terminals. 3270 data can be presented to the user by emulating a 3270 terminal or by using a graphical user interface.

external security interface (ESI)

A facility that enables client applications to verify and change passwords for user IDs on CICS servers.

F**firewall**

A configuration of software that prevents unauthorized traffic between a trusted network and an untrusted network.

FMH See *function management header*.

fully-qualified APPLID

Used to identify CICS Transaction Gateway connections on the CICS server and tasks in a CICSplex. It is composed of an APPLID with an optional network qualifier. See also *APPLID* and *APPLID qualifier*.

function management header (FMH)

One or more headers, optionally present in the leading request units (RUs) of an RU chain, that allow one LU to (a) select a transaction program or device at the session partner and control the way in which the end-user data it sends is handled at the destination, (b) change the destination or the characteristics of the data during the session, and (c) transmit between session partners status or user

information about the destination (for example, a program or device). Function management headers can be used with LU type 1, 4, and 6.2 protocols.

G

gateway

A device or program used to connect two systems or networks.

gateway classes

The gateway classes provide APIs for ECI, EPI, and ESI that allow communication between Java client applications and the Gateway daemon.

Gateway daemon

A long-running Java process that listens for network requests from remote Client applications. It issues these requests to CICS servers using the CICS connectivity components. The Gateway daemon on z/OS processes ECI requests and on UNIX, Windows, and Linux platforms it process EPI and ESI requests as well. The Gateway daemon uses the GATEWAY section of ctg.ini for its configuration.

Gateway group

A set of Gateway daemons that share an APPLID qualifier, and where each Gateway daemon has a unique APPLID within the Gateway group.

gateway token

A token that represents a specific Gateway daemon, when a connection is established successfully. Gateway tokens are used in the C language statistics and ECI V2 APIs.

global transaction

A recoverable unit of work performed by one or more resource managers in a distributed transaction processing environment and coordinated by an external transaction manager.

H

HA group

See *highly available gateway group*.

highly available gateway group (HA group)

A Gateway group that utilizes TCP/IP load balancing, and can be viewed as a single logical Gateway daemon. A Gateway daemon instance in a HA group can recover indoubt XA transactions on behalf of another Gateway daemon within the HA group

host A computer that is connected to a network (such as the Internet or an

SNA network) and provides an access point to that network. The host can be any system; it does not have to be a mainframe.

host address

An IP address that is used to identify a host on a network.

host ID

In TCP/IP, that part of the IP address that defines the host on the network. The length of the host ID depends on the type of network or network class (A, B, or C).

host name

In the Internet suite of protocols, the name given to a computer. Sometimes, host name is used to mean the fully qualified domain name; other times, it is used to mean the most specific subname of a fully qualified domain name. For example, if mycomputer.city.company.com is the fully qualified domain name, either of the following can be considered the host name: mycomputer.city.company.com, mycomputer.

hover help

Information that can be viewed by holding a mouse over an item such as an icon in the user interface.

HTTP See *Hypertext Transfer Protocol*.

HTTPS

See *Hypertext Transfer Protocol Secure*.

Hypertext Transfer Protocol (HTTP)

In the Internet suite of protocols, the protocol that is used to transfer and display hypertext and XML documents.

Hypertext Transfer Protocol Secure (HTTPS)

A TCP/IP protocol that is used by World Wide Web servers and Web browsers to transfer and display hypermedia documents securely across the Internet.

I

ID data

An ID data structure holds an individual result from a statistical API function.

identity propagation

The concept of preserving a user's security identity information (the distributed identity) independent of where the identity information has been created, for use during authorization and for auditing purposes. The distributed identity is carried with a request from the distributed client application to the CICS server, and is incorporated in the access control of the server as part of the authorization process,

for example, using RACF®. CICS Transaction Gateway flows the distributed identity to CICS. See also *distributed identity*.

identity propagation login module

A code component that provides support for identity propagation. The identity propagation login module is included with the CICS Transaction Gateway ECI resource adapters (cicseci.rar and cicseciXA.rar), conforms to the JAAS specification and is contained in a single Java class within the resource adapter. See also *identity propagation*.

iKeyman

A tool for maintaining digital certificates for JSSE.

in doubt

The state of a transaction that has completed the prepare phase of the two-phase commit process and is waiting to be completed.

in flight

The state of a transaction that has not yet completed the prepare phase of the two-phase commit process.

independent logical unit

A logical unit (LU) that can both send and receive a BIND, and which supports single, parallel, and multiple sessions. See *BIND*.

<install_path>

This term is used in file paths to represent the directory where you installed the product.

Internet Architecture Board

The technical body that oversees the development of the internet suite of protocols known as TCP/IP.

Internet Protocol (IP)

In TCP/IP, a protocol that routes data from its source to its destination in an Internet environment.

interoperability

The capability to communicate, run programs, or transfer data among various functional units in a way that requires the user to have little or no knowledge of the unique characteristics of those units.

IP Internet Protocol.

IPIC See *IP interconnectivity*.

IP address

A unique address for a device or logical unit on a network that uses the IP standard.

IP interconnectivity (IPIC)

The IPIC protocol enables Distributed Program Link (DPL) access from a non-CICS program to a CICS program over TCP/IP, using the External Call Interface (ECI). IPIC passes and receives data using COMMAREAs, or containers.

J

J2EE See *Java 2 Platform Enterprise Edition*

J2EE Connector architecture (JCA)

A standard architecture for connecting the J2EE platform to heterogeneous enterprise information systems (EIS).

Java An object-oriented programming language for portable interpretive code that supports interaction among remote objects.

Java 2 Platform Enterprise Edition (J2EE)

An environment for developing and deploying enterprise applications, defined by Sun Microsystems Inc. The J2EE platform consists of a set of services, application programming interfaces (APIs), and protocols that allow multi-tiered, Web-based applications to be developed.

JavaBeans

As defined for Java by Sun Microsystems, a portable, platform-independent, reusable component model.

Java Client application

The Java client application is a user application written in Java, including servlets and enterprise beans, that uses the Gateway classes.

Java Development Kit (JDK)

The name of the software development kit that Sun Microsystems provided for the Java platform, up to and including v 1.1.x. Sometimes used erroneously to mean the Java platform or as a generic term for any software developer kits for Java.

JavaGateway

The URL of the CICS Transaction Gateway with which the Java Client application will communicate. The JavaGateway takes the form `protocol://address:port`. These protocols are supported: `tcp://`, `ssl://`, and `local:`. The CICS Transaction Gateway runs with the default port value of 2006. This parameter is not relevant if you are using the protocol `local:`. For example, you might specify a JavaGateway of `tcp://ctg.business.com:2006`. If you specify the protocol as `local:` you will connect directly to the CICS server, bypassing any CICS Transaction Gateway servers.

Java Native Interface (JNI)

A programming interface that allows Java code running in a Java virtual machine to work with functions that are written in other programming languages.

Java Runtime Environment (JRE)

A subset of the Java Software Development Kit (SDK) that supports the execution, but not the development, of Java applications. The JRE comprises the Java Virtual Machine (JVM), the core classes, and supporting files.

Java Secure Socket Extension (JSSE)

A Java package that enables secure Internet communications. It implements a Java version of the Secure Sockets Layer (SSL) and Transport Layer Security (TSL) protocols and supports data encryption, server authentication, message integrity, and optionally client authentication.

Java virtual machine (JVM)

A software implementation of a processor that runs compiled Java code (applets and applications).

JDK See *Java development kit*.

JCA See *J2EE Connector Architecture* .

JNI See *Java Native Interface*.

JRE See *Java Runtime Environment*

JSSE See *Java Secure Socket Extension*.

JVM See *Java Virtual Machine*.

K**keyboard mapping**

A list that establishes a correspondence between keys on the keyboard and characters displayed on a display screen, or action taken by a program, when that key is pressed.

key ring

In the JSSE protocol, a file that contains public keys, private keys, trusted roots, and certificates.

L**local mode**

Local mode describes the use of the CICS Transaction Gateway *local* protocol. The Gateway daemon is not used in local mode.

local transaction

A recoverable unit of work managed by a resource manager and not coordinated by an external transaction manager.

logical CICS server

A logical CICS server name is an alias that can be passed on an ECI request when running in remote mode to CICS Transaction Gateway for z/OS. The alias name is mapped to an actual CICS server name by a LOGICALSERVER definition in the configuration file (ctg.ini).

logical end of day

The local time of day on the 24-hour clock to which a Gateway daemon aligns statistics intervals. If the statistics interval is 24 hours, this is the local time at which interval statistics will be reset and, on z/OS, optionally recorded to SMF. This time is set using the **stateod** parameter in the configuration file (ctg.ini).

logical unit (LU)

In SNA, a port through which an end user accesses the SNA network to communicate with another end user and through which the end user accesses the functions provided by system services control points (SSCP). An LU can support at least two sessions, one with an SSCP and one with another LU, and might be capable of supporting many sessions with other logical units. See also *network addressable unit*, *primary logical unit*, *secondary logical unit*.

logical unit 6.2 (LU 6.2)

A type of logical unit that supports general communications between programs in a distributed processing environment.

The LU type that supports sessions between two applications using APPC.

logical unit of work (LUW)

The processing that a program performs between synchronization points

LU See *logical unit*.

LU 6.2 See *logical unit 6.2*.

LU-LU session

In SNA, a session between two logical units (LUs) in an SNA network. It provides communication between two end users, or between an end user and an LU services component.

LU-LU session type 6.2

In SNA, a type of session for communication between peer systems. Synonymous with APPC protocol.

LUW See *logical unit of work*.

M

managed mode

Describes an environment in which connections are obtained from connection factories that the J2EE server has set up. Such connections are owned by the J2EE server.

media access control (MAC) sublayer

One of two sublayers of the ISO Open Systems Interconnection data link layer proposed for local area networks by the IEEE Project 802 Committee on Local Area Networks and the European Computer Manufacturers Association (ECMA). It provides functions that depend on the topology of the network and uses services of the physical layer to provide services to the logical link control (LLC) sublayer. The OSI data link layer corresponds to the SNA data link control layer.

method

In object-oriented programming, an operation that an object can perform. An object can have many methods.

mode In SNA, a set of parameters that defines the characteristics of a session between two LUs.

N

name server

In TCP/IP, synonym for Domain Name Server. In Internet communications, a host that translates symbolic names assigned to networks and hosts into IP addresses.

NAU See *network addressable unit*.

network address

In SNA, an address, consisting of subarea and element fields, that identifies a link, link station, or network addressable unit (NAU). Subarea nodes use network addresses; peripheral nodes use local addresses. The boundary function in the subarea node to which a peripheral node is attached transforms local addresses to network addresses and vice versa. See also *network name*.

network addressable unit (NAU)

In SNA, a logical unit, a physical unit, or a system services control point. The NAU is the origin or the destination of information transmitted by the path control network. See also *logical unit*, *network address*, *network name*.

network name

In SNA, the symbolic identifier by which end users refer to a network addressable unit (NAU), link station, or link. See also *network address*.

node type

In SNA, a designation of a node according to the protocols it supports and the network addressable units (NAUs) it can contain. Four types are defined: 1, 2, 4, and 5. Type 1 and type 2 nodes are peripheral nodes; type 4 and type 5 nodes are subarea nodes.

nonextended logical unit of work

See *SYNCONRETURN*.

nonmanaged mode

An environment in which the application is responsible for generating and configuring connection factories. The J2EE server does not own or know about these connection factories and therefore provides no Quality of Service facilities.

O

object In object-oriented programming, a concrete realization of a class that consists of data and the operations associated with that data.

object-oriented (OO)

Describing a computer system or programming language that supports objects.

one-phase commit

A protocol with a single commit phase, that is used for the coordination of changes to recoverable resources when a single resource manager is involved.

OO See *object-oriented*.

P**pacing**

A technique by which a receiving station controls the rate of transmission of a sending station to prevent overrun.

parallel session

In SNA, two or more concurrently active sessions between the same two LUs using different pairs of network addresses. Each session can have independent session parameters.

PING In Internet communications, a program used in TCP/IP networks to test the ability to reach destinations by sending the destinations an Internet Control Message Protocol (ICMP) echo request and waiting for a reply.

partner logical unit (PLU)

In SNA, the remote participant in a session.

partner transaction program

The transaction program engaged in an APPC conversation with a local transaction program.

PLU See *primary logical unit* and *partner logical unit*.

port An endpoint for communication between devices, generally referring to a logical connection. A 16-bit number identifying a particular Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) resource within a given TCP/IP node.

port sharing

A way of load balancing TCP/IP connections across a group of servers running in the same z/OS image.

prepare phase

The first phase of a XA process in which all participants are requested to confirm readiness to commit.

presentation logic

The part of a distributed application that is concerned with the user interface of the application. Compare with *business logic*.

primary logical unit (PLU)

In SNA, the logical unit that contains the primary half-session for a particular logical unit-to-logical unit (LU-to-LU) session. See also *secondary logical unit*.

protocol boundary

The signals and rules governing interactions between two components within a node.

Q**Query strings**

Query strings are used in the statistical data API. A query string is an input parameter, specifying the statistical data to be retrieved.

R

RACF See *Resource Access Control Facility*.

realm A named collection of users and groups that can be used in a specific security context. See also *distinguished name* and *identity propagation*.

Recoverable resource management services (RRMS)

The registration services, context services, and resource recovery services provided by the z/OS sync point manager that enable consistent changes to be made to multiple protected resources.

Resource Access Control Facility (RACF)

An IBM licensed program that provides access control by identifying

users to the system; verifying users of the system; authorizing access to protected resources; logging detected unauthorized attempts to enter the system; and logging detected accesses to protected resources.

region In workload management on CICS Transaction Gateway for Windows, an instance of a CICS server.

remote mode

Remote mode describes the use of one of the supported CICS Transaction Gateway network protocols to connect to the Gateway daemon.

remote procedure call (RPC)

A protocol that allows a program on a client computer to run a program on a server.

Request monitoring exits

Exits that provide information about individual requests as they are processed by the CICS Transaction Gateway.

request unit (RU)

In SNA, a message unit that contains control information such as a request code, or function management (FM) headers, end-user data, or both.

request/response unit

A generic term for a request unit or a response unit. See also *request unit* and *response unit*.

response file

A file that contains predefined values that is used instead of someone having to enter those values one at a time. See also *CID methodology*.

response unit (RU)

A message unit that acknowledges a request unit; it can contain prefix information received in a request unit.

Resource adapter

A system-level software driver that is used by an EJB container or an application client to connect to an enterprise information system (EIS). A resource adapter plugs in to a container; the application components deployed on the container then use the client API (exposed by adapter) or tool-generated, high-level abstractions to access the underlying EIS.

resource group ID

A resource group ID is a logical grouping of resources, grouped for statistical purposes. A resource group ID is associated with a number of resource group statistics, each identified by a statistic ID.

resource ID

A resource ID refers to a specific resource. Information about the resource is included in resource-specific statistics. Each statistic is identified by a statistic ID.

resource manager

The participant in a transaction responsible for controlling access to recoverable resources. In terms of the CICS resource adapters this is represented by an instance of a ConnectionFactory.

Resource Recovery Services (RRS)

A z/OS facility that provides two-phase sync point support across participating resource managers.

Result set

A result set is a set of data calculated or recorded by a statistical API function.

Result set token

A result set token is a reference to the set of results returned by a statistical API function.

rollback

An operation in a transaction that reverses all the changes made during the unit of work. After the operation is complete, the unit of work is finished. Also known as a backout.

RU See *Request unit* and *Response unit*.

RPC See *remote procedure call*.

RRMS

See *Recoverable resource management services*.

RRS See *Resource Recovery Services*.

S

SBCS See *single-byte character set*.

secondary logical unit (SLU)

In SNA, the logical unit (LU) that contains the secondary half-session for a particular LU-LU session. Contrast with primary logical unit. See also *logical unit*.

Secure Sockets Layer (SSL)

A security protocol that provides communication privacy. SSL enables client/server applications to communicate in a way that is designed to prevent eavesdropping, tampering, and message forgery. SSL applies only to internet protocols, and is not applicable to SNA.

server name remapping

See *dynamic server selection*.

servlet

A Java program that runs on a Web server and extends the server's functionality by generating dynamic content in response to Web client requests. Servlets are commonly used to connect databases to the Web.

session limit

In SNA, the maximum number of concurrently active logical unit to logical unit (LU-to-LU) sessions that a particular logical unit (LU) can support.

silent installation

Installation that does not display messages or windows during its progress. Silent installation is not a synonym of "unattended installation", although it is often improperly used as such.

single-byte character set (SBCS)

A character set in which each character is represented by 1 byte. Contrast with double-byte character set.

sign-on capable terminal

A sign-on capable terminal allows sign-on transactions, either CICS-supplied (CESN) or user-written, to be run. Contrast with sign-on incapable terminal.

SIT See *system initialization table*.

SLU See *secondary logical unit*.

SMIT See *System Management Interface Tool*.

SNA See *Systems Network Architecture*.

SNA sense data

An SNA-defined encoding of error information. In SNA, the data sent with a negative response, indicating the reason for the response.

SNASVCMG mode name

The SNA service manager mode name. This is the architecturally-defined mode name identifying sessions on which CNOS is exchanged. Most APPC-providing products predefine SNASVCMG sessions.

socket A network communication concept, typically representing a point of connection between a client and a server. A TCP/IP socket will normally combine a host name or IP address, and a port number.

SSL See *Secure Sockets Layer*.

SSLight

An implementation of SSL, written in Java, and no longer supported by CICS Transaction Gateway.

statistic data

A statistic data structure holds individual statistical result returned after calling a statistical API function.

statistic group

A generic term for a collection of statistic IDs.

statistic ID

A label referring to a specific statistic. A statistic ID is used to retrieve specific statistical data, and always has a direct relationship with a statistic group.

standard error

In many workstation-based operating systems, the output stream to which error messages or diagnostic messages are sent.

subnet

An interconnected, but independent segment of a network that is identified by its Internet Protocol (IP) address.

subnet address

In Internet communications, an extension to the basic IP addressing scheme where a portion of the host address is interpreted as the local network address.

sync point

Synchronization point. During transaction processing, a reference point to which protected resources can be restored if a failure occurs.

SYNCONRETURN

A request where the CICS server takes a sync point on successful completion of the server program. Changes to recoverable resources made by the server program are committed or rolled-back independently of changes to recoverable resources made by the client program issuing the ECI request, or changes made by the server in any subsequent ECI request. Also referred to as a *nonextended logical unit of work*.

system initialization table (SIT)

A table containing parameters used to start a CICS control region.

System Management Command

An administrative request received by a Gateway daemon (or Gateway daemon address space on z/OS) from the **ctgadmin** command (on UNIX, Linux, or Windows) or the z/OS console. The request might be made to retrieve information about the Gateway daemon, or to alter some aspect of Gateway daemon behavior.

Typically, a **ctgadmin** command in the form **ctgadmin** <command string> is entered by an operator using the command line interface, or a modify command in the form /F <job name>,APPL=<command string> is entered by an operator on the z/OS console.

System Management Interface Tool (SMIT)

An interface tool of the AIX® operating system for installing, maintaining, configuring, and diagnosing tasks.

Systems Network Architecture (SNA)

An architecture that describes the logical structure, formats, protocols, and operational sequences for transmitting information units through the networks and also the operational sequences for controlling the configuration and operation of networks.

System SSL

An implementation of SSL, no longer supported by CICS Transaction Gateway on z/OS.

T

TCP/IP

See *Transmission Control Protocol/Internet Protocol*.

TCP/IP load balancing

The ability to distribute TCP/IP connections across target servers.

terminal emulation

The capability of a personal computer to operate as if it were a particular type of terminal linked to a processing unit and to access data. See also *emulator*, *emulation program*.

thread A stream of computer instructions that is in control of a process. In some operating systems, a thread is the smallest unit of operation in a process. Several threads can run concurrently, performing different jobs.

timeout

A time interval that is allotted for an event to occur or complete before operation is interrupted.

TLS See *Transport Layer Security*.

token-ring network

A local area network that connects devices in a ring topology and allows unidirectional data transmission between devices by a token-passing procedure. A device must receive a token before it can transmit data.

trace A record of the processing of a computer program. It exhibits the sequences in which the instructions were processed.

transaction manager

A software unit that coordinates the activities of resource managers by managing global transactions and coordinating the decision to commit them or roll them back.

transaction program

A program that uses the Advanced Program-to-Program Communications (APPC) application programming interface (API) to communicate with a partner application program on a remote system.

Transmission Control Protocol/Internet Protocol (TCP/IP)

An industry-standard, nonproprietary set of communications protocols that provide reliable end-to-end connections between applications over interconnected networks of different types.

Transport Layer Security (TLS)

A security protocol that provides communication privacy. TLS enables client/server applications to communicate in a way that is designed to prevent eavesdropping, tampering, and message forgery. TLS applies only to internet protocols, and is not applicable to SNA. TLS is also known as SSL 3.1.

two-phase commit

A protocol with both a prepare and a commit phase, that is used for the coordination of changes to recoverable resources when more than one resource manager is used by a single transaction.

type 2.0 node

A node that attaches to a subarea network as a peripheral node and provides a range of end-user services but no intermediate routing services.

type 2.1 node

An SNA node that can be configured as an endpoint or intermediate routing node in a network, or as a peripheral node attached to a subarea network.

U**unattended installation**

Unattended installation is installation performed without user interaction during its progress, or, with no user present at all, except for the initial launch of the process. -

Uniform Resource Locator (URL)

A sequence of characters that represent information resources on a computer or in a network such as the Internet. This sequence of characters includes (a) the abbreviated name of the protocol used to access the information resource and (b) the information used by the protocol to locate the information resource.

unit of recovery (UR)

A defined package of work to be performed by the RRS.

unit of work (UOW)

A recoverable sequence of operations performed by an application between two points of consistency. A unit of work begins when a transaction starts or at a user-requested sync point. It ends either at a user-requested sync point or at the end of a transaction.

UOW See *unit of work*.

UR See *unit of recovery*.

URL See *Uniform Resource Locator*.

user registry

The location where the distinguished name of a user is defined and authenticated. See also *distinguished name*.

user session

Any APPC session other than a SNASVCMG session.

V

verb A reserved word that expresses an action to be taken by an application programming interface (API), a compiler, or an object program.

In SNA, the general name for a transaction program's request for communication services.

version string

A character string containing version information about the statistical data API.

W

WAN See *wide area network*.

Web browser

A software program that sends requests to a Web server and displays the information that the server returns.

Web server

A software program that responds to information requests generated by Web browsers.

wide area network (WAN)

A network that provides communication services to a geographic area larger than that served by a local area network or a metropolitan area network, and that can use or provide public communication facilities.

wrapping trace

On Windows, UNIX, and Linux, a configuration in which the **Maximum Client wrap size** setting is greater than 0. The total size of Client daemon binary trace files is limited to the value specified in the **Maximum Client wrap size** setting. With standard I/O tracing, two files, called `cicscli.bin` and `cicscli.wrp`, are used; each can be up to half the size of the **Maximum Client wrap size**.

X**XA requests**

An XA request is any request sent or received by the CICS Transaction Gateway in support of an XA transaction. These requests include the XA commands `commit`, `complete`, `end`, `forget`, `prepare`, `recover`, `rollback`, and `start`.

XA transaction

A global transaction that adheres to the X/Open standard for distributed transaction processing (DTP.)

Index

A

accessibility 129

D

disability 129

documentation 125

G

glossary of terms and abbreviations 131

P

publications 125

T

trademarks 160

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

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

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply in the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the information. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this information at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM United Kingdom Laboratories, MP151, Hursley Park, Winchester, Hampshire, England, SO21 2JN. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Programming License Agreement, or any equivalent agreement between us.

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 and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)[®] are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. 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 IBM trademarks is available on the Web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Other product and service names might be trademarks of IBM or other companies.

Sending your comments to IBM

If you especially like or dislike anything about this book, use one of the methods listed below to send your comments to IBM.

Feel free to comment on what you regard as specific errors or omissions, and on the accuracy, organization, subject matter, or completeness of this book.

Limit your comments to the information in this book and the way in which the information is presented.

To ask questions, make comments about the functions of IBM products or systems, or to request additional publications, contact your IBM representative or your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate, without incurring any obligation to you.

You can send your comments to IBM in any of the following ways:

- By mail, to this address:
User Technologies Department (MP095)
IBM United Kingdom Laboratories
Hursley Park
WINCHESTER,
Hampshire
SO21 2JN
United Kingdom
- By fax:
 - +44 1962 842327 (if you are outside the UK)
 - 01962 842327 (if you are in the UK)
- Electronically, use the appropriate network ID:
 - IBM Mail Exchange: GBIBM2Q9 at IBMMAIL
 - IBMLink: HURSLEY(IDRCF)
 - Internet: idrcf@hursley.ibm.com

Whichever you use, ensure that you include:

- The publication title and order number
- The topic to which your comment applies
- Your name and address/telephone number/fax number/network ID.



Product Number: 5655-W10, 5724-I81, and 5725-B65

SC34-7061-02



Spine information:



CICS Transaction Gateway

CICS Transaction Gateway Messages

Version 8.0