CICS Transaction Gateway

IBM

Messages

Version 7.1

CICS Transaction Gateway

IBM

Messages

Version 7.1

Contents

About this book

This book lists the error and warning messages for the CICS® Transaction Gateway and CICS Universal Client products, Version 7.1.

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

CCL0510E   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.

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.

1

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'.

CTG0801E   **CTGBATCH Error** *errno* **occurred writing to log destination** *dest-number* **. errno2=**errno2 **.**

Explanation: CTGBATCH has encountered an error *errno* while attempting to write to logging destination *dest-number* (1=STDOUT 2=STDERR).

System action: CTGBATCH will attempt to re-direct subsequent logging data to the default JES log destination.

User response: Investigate why the desired logging destination has become unavailable and correct the problem. It is possible that there is no space left on the chosen file system.

CTG0803E   **CTGBATCH Create pipes failed with errno=**errno **errno2=**errno2 **. Pipes opened=**number of open pipes **.**

Explanation: CTGBATCH has encountered error *errno* while creating pipes to communicate with the target program. Without the ability to create and use inter-process pipes, logging will be unavailable.

System action: CTGBATCH will terminate with JES return code 16, after writing diagnostic information to the STDERR log destination.

User response: The z/OS C/C++ Run-Time Library Reference (SC28-1663) documents the possible errno values and likely reason for the failure. In this case, the failing function call is pipe(). If problems persist, contact

your service organization and supply the diagnostic log data together with the JCL being used to start the Gateway daemon.

CTG0804E   **CTGBATCH Error** *errno* **occurred starting child process for** *HFS_executable* **. errno2=**errno2 **.**

Explanation: CTGBATCH has encountered error *errno* while starting the target program *HFS_executable* . Possible reasons could be that the user does have the correct authority to execute the target program, the program is not an USS executable or valid shell script.

System action: CTGBATCH will terminate with JES return code 8, after writing diagnostic information to the STDERR log destination.

User response: The z/OS C/C++ Run-Time Library Reference (SC28-1663) documents the possible errno values and likely reason for the failure. In this case, the failing function call is spawnp(). If problems persist, contact your service organization and supply the diagnostic log data together with the JCL being used to start the Gateway daemon.

CTG0805E   **CTGBATCH ReadPipeWriteFile bad pipe=**=*pipe_number* **.**

Explanation: CTGBATCH has encountered an unexpected internal error when processing log data.

System action: CTGBATCH attempts to continue.

User response: Contact your service organization and supply the diagnostic log data together with the JCL being used to start the Gateway daemon.

CTG0806E   **CTGBATCH Error** *errno* **occurred reading output of pipe** *pipe_number* **. errno2=**errno2 **.**

Explanation: CTGBATCH has encountered an error while attempting to read log data from the target program. It is possible that the target program has ended abnormally, before CTGBATCH had a chance to process the latest log data.

System action: CTGBATCH will close the resources associated with this log before continuing.

User response: If the problem is believed to be in the target program, then check for any diagnostic information specific to the target program that may have logged before it ended abnormally. If this is not the case, contact your service organization and supply the diagnostic log data together with the JCL being used to start the Gateway daemon.

CTG0807E CTGBATCH Dump of internal state will follow: *state-information*

Explanation: CTGBATCH has encountered an unexpected condition.

System action: CTGBATCH will write internal state data to the error log.

User response: Check the diagnostic messages immediately prior to this state information dump for an indication of the specific problem.

CTG0808W CTGBATCH STDOUT DD statement missing or invalid. JES will dynamically allocate a JES LOG destination.

Explanation: CTGBATCH was unable to use the supplied STDOUT DD statement, or the STDOUT DD statement was not defined.

System action: CTGBATCH continues. A JES log will be dynamically defined for stdout log messages. The log name will likely be 'SYS00001'. When CTGBATCH completes, the JES return code will be at least 4.

User response: If the JCL for CTGBATCH did contain a STDOUT DD statement, verify that the definition is correct.

CTG0809W CTGBATCH STDERR DD statement missing or invalid. JES will dynamically allocate a JES LOG destination.

Explanation: CTGBATCH was unable to use the supplied STDERR DD statement, or the STDERR DD statement was not defined.

System action: CTGBATCH continues. A JES log will be dynamically defined for stdout log messages. The log name will likely be 'SYSOUT', although this can be overridden by the LE runtime option ″MSGFILE″. When CTGBATCH completes, the JES return code will be at least 4.

User response: If the JCL for CTGBATCH did contain a STDERR DD statement, verify that the definition is correct.

CTG0810W CTGBATCH STDENV DD statement missing or invalid. CTGBATCH will attempt to continue.

Explanation: CTGBATCH was unable to use the supplied STDENV DD statement, or the STDENV DD statement was not defined.

System action: CTGBATCH continues. If the target program is a USS executable (e.g. ctgmsgs) then a STDENV may not be required.

User response: If the JCL for CTGBATCH did contain

a STDENV DD statement, verify that the definition is correct.

CTG0812W CTGBATCH Runtime env *env-var* **is not set.**

Explanation: CTGBATCH diagnostic checks have found that a significant environment variable has not been set.

System action: CTGBATCH continues.

User response: Check whether or not the listed variable should be set in the STDENV data. It will be one of the following: - PATH should at least include ″/bin″ for base USS system commands and ″<Java path>/bin″ for the JVM startup script ″java″. - CICSCLI should specify the path to the ″ctg.ini″ file to be used by this Gateway daemon. If omitted, the Gateway daemon will attempt to find ″ctg.ini″ in the product ″bin″ directory. - TMPDIR should typically be set to ″/tmp″; however, if /tmp is prone to becoming full - then it is advisable to specify a writable path which is dedicated to Gateway daemon usage. The ctgstart script utilizes fundamental USS commands which depend on the TMPDIR path having free space and being writable. - _BPX_SHAREAS should be set to YES when CTGBATCH is being used to launch either the Gateway daemon or ctgmaster. This will ensure mean that the target executable and CTGBATCH will run in the same address space if at all possible. - TZ should be set to reflect the local timezone and daylight savings time. The setting should usually reflect the setting of TZ in /etc/profile. The full format is : TZ= standardHH[:MM[:SS]] [daylight[HH[:MM[:SS:]]] [,startdate[/starttime],enddate[/endtime]]] An example for UK could be TZ=GMT0BST,M3.5.0,M10.4.0 or for US Eastern could be TZ=EST5EDT For a further information please refer to z/OS UNIX System Services Command Reference (SA22-7802).

CTG0814W CTGBATCH RLIMIT_AS query failed with *errno*

Explanation: CTGBATCH diagnostics have failed to obtain the Region size data.

System action: CTGBATCH continues.

User response: This is a soft failure within CTGBATCH diagnostics, likely related to the problem for which diagnostics were enabled.

CTG0816W CTGBATCH Call to getcwd() failed.

Explanation: CTGBATCH diagnostics have failed to obtain the current working directory data.

System action: CTGBATCH continues.

User response: This is a soft failure within CTGBATCH diagnostics, likely related to the problem for which diagnostics were enabled.

CTG0822W   CTGBATCH Call to getpwnam() failed.

Explanation: CTGBATCH diagnostics have failed to obtain specific data relating to the current Runtime user.

System action: CTGBATCH continues.

User response: This is a soft failure within CTGBATCH diagnostics, likely related to the problem for which diagnostics were enabled.

CTG0823E   CTGBATCH GetMessage() has been passed a NULL pointer.

Explanation: CTGBATCH has encountered an unexpected internal error when generating a local message text.

System action: CTGBATCH attempts to continue.

User response: Contact your service organization and supply the diagnostic log data together with the JCL being used to start the Gateway daemon.

CTG0824E   CTGBATCH GetMessage() Message number *message-number* **unknown.**

Explanation: CTGBATCH has encountered an unexpected internal error when generating a local message text.

System action: CTGBATCH attempts to continue.

User response: Contact your service organization and supply the diagnostic log data together with the JCL being used to start the Gateway daemon.

CTG0827W   CTGBATCH Child completed with a non-zero return code *rc*

Explanation: The target program completed with a non-zero return code.

System action: CTGBATCH will end normally.

User response: Investigate the target program log data to determine why it may have completed with a non-zero return code.

CTG0828E   CTGBATCH The target executable *target-program* **does not exist.**

Explanation: The target executable *target-program* could not be found.

System action: CTGBATCH will terminate with JES return code 8.

User response: Check that *target-program* is correct. Ensure that the PARM string specified for the EXEC PGM=CTGBATCH JCL statement is correct. Up to and including the first forward slash (/) character of the PARM string is taken to be a LE runtime option. Therefore, ensure that the HFS path component of the PARM string includes its own leading / character. See

the Administration book for more details.

CTG0829W   CTGBATCH Call to __getLogin1() failed.

Explanation: CTGBATCH diagnostics have failed to obtain general data relating to the current Runtime user.

System action: CTGBATCH continues.

User response: This is a soft failure within CTGBATCH diagnostics, likely related to the problem for which diagnostics were enabled.

CTG0830W   CTGBATCH Call to setLocale() (query only) failed.

Explanation: CTGBATCH diagnostics have failed to obtain Locale data.

System action: CTGBATCH continues.

User response: This is a soft failure within CTGBATCH diagnostics, likely related to the problem for which diagnostics were enabled.

CTG0831E   CTGBATCH Call to setenv(*env-var***) failed with errno** *errno* **.**

Explanation: CTGBATCH has encountered error *errno* while attempting to set an environment variable from the STDENV data.

System action: CTGBATCH continues.

User response: The z/OS C/C++ Run-Time Library Reference (SC28-1663) documents the possible errno values and likely reason for the failure. In this case, the failing function call is setenv(). If problems persist, contact your service organization and supply the diagnostic log data together with the JCL being used to start the Gateway daemon.

CTG0832E   CTGBATCH More than one language DD statement was defined.

Explanation: CTGBATCH detected that more than one of CTGMSGEN, CTGMSGJA or CTGMSGZH DD statements have been defined.

System action: CTGBATCH will terminate with JES return code 20.

User response: Check the JCL used to start CTGBATCH and ensure that at most one of CTGMSGEN, CTGMSGJA or CTGMSGZH dummy DD statements are defined.

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.

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.

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*, **TermId =** *Termid* **)**

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 TermId.

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.

CCL1122E   **Invalid list of components. Please re-type.**

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.

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.

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.

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.

CCL1214E   A thread ended unexpectedly while writing a trace point (pid = *pid*, tid = *tid*, trace pt = *trace pt*, buffer size = *size*)

Explanation: A thread ended unexpected, 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.

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.

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.

CCL1521E **Exception** *exception* **raised in** *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 CclECI::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.

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.

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*, *index*, *name*)

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.

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.

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 organisation.

CCL2200E   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 message is logged, and processing continues. The requested action does not succeed.

User response: Check the Windows Event Log for further information on the error. If the problem persists, contact your service organization.

CCL2201E   Cannot communicate with the CICS UC service (Error Code=*error code* **)**

Explanation: An error occurred during communications with the IBM CICS Universal Client service. The Windows OpenService function returned the specified error.

System action: The message is logged, and processing continues. The requested action does not succeed.

User response: Check the Windows Event Log for further information on the error. If the problem persists, contact your service organization.

CCL2202E   Cannot start the Client daemon (Error Code=*error code* **)**

Explanation: An error occurred while the IBM CICS Universal Client service was starting. The Windows Service Control Manager returned the specified error.

System action: The message is logged. Neither the Client daemon nor the IBM CICS Universal Client service start.

User response: Check the Windows Event Log for further information on the error. If the problem persists, contact your service organization.

CCL2203E   An error occurred while the CICS UC service was stopping. (Error Code=*error code* **)**

Explanation: An error occurred while the IBM CICS Universal Client service was stopping. The Windows Service Control Manager returned the specified error.

System action: The message is logged. The Client daemon shuts down. The IBM CICS Universal Client service might not be marked as terminated.

User response: Check the Windows Event Log for further information on the error. If the problem persists, contact your service organization.

CCL2204E   Cannot send control message to the CICS UC service (Error Code=*error code* **)**

Explanation: An error occurred while an internal message was being sent to the IBM CICS Universal Client service.

System action: The message is logged, and processing continues. The requested action does not succeed.

User response: Check the Windows Event Log for further information on the error. If the problem persists, contact your service organization.

CCL2205E   Insufficient security permissions to communicate with the CICS UC service.

Explanation: The current user has insufficient authority for the administration of Windows Services.

System action: The message is logged, and processing continues. The requested action does not succeed.

User response: Check the Windows Event Log for further information on the error. Ensure the current user has the required level of authority.

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.

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

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 may terminate.

User response: See following message for details and correct the configuration file and restartthe 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 value too high; the maximum permitted value *max* has been used.**

Explanation: MAXREQUESTS 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 Universal Client is intended for single user use only; the CICS Transaction Gateway 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.

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 will fail to start.

User response: The Information Center for the CICS TG and CICS UC 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 ″APPLID″ parameter**

Explanation: The ″Application ID″ parameter specified on SECTION CLIENT is deprecated and is overidden 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 overidden. 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.

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 signon capable terminals.

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

User response: Use the option to request a non-signon 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.

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* **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 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*

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*

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*

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*

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*

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*

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*

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*

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*

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.

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.

CCL3315E   Unexpected comms error (internal return codes *rc1* **,** *rc2* **)**

Explanation: An network communications error occurred.

System action: The message is written to the error log and the Client continues. The internal return codes provide information for use by IBM service groups.

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* **(***errordesc* **)**

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 *errordesc* 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* **(***errordesc* **)**

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 *errordesc* 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. 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 failed, ErrNo = ***error* **(***errordesc* **)**

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 *errordesc* 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 failed, ErrNo = ***error* **(***errordesc* **)**

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 *errordesc* 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 failed, ErrNo = ***error* **(***errordesc* **)**

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 *errordesc* 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.

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 recv 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 addressof 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 is closed.

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

User response: 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 = *errordesc*)

Explanation: The TCP/IP socket send call did not complete within 5 seconds. The *errordesc* 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 within5 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.

CCL4449E TCP/IP data to send too long, length = *INSERT-0*

Explanation: The data to send over the TCP/IP socket was too long

System action: The data is truncated to the maximum size and sent.

User response: This is an internal error. Contact your service organization.

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.

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.

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.

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 real LU names, correct the 'NETNAME' parameter and retry. If using alias LU names, set the 'PARTNERLUALIAS' parameter to 'Y' 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 WinAPPCStartup 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 not loaded or terminated, APPC return code *primary RC*

Explanation: APPC could not execute the verb because a component of the product providing the SNA

transport could not be loaded or terminated whilst processing the verb.

System action: The Client continues.

User response: Contact your system administrator.

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   Communications subsystem abended, APPC return code *primary RC***,** *secondary RC*

Explanation: Either the APPC provider has abended or has not been configured properly on the Client machine.

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

User response: Check the configuration of the APPC provider on the CICS Client machine. If using NWSAA v2.2, check that the NDS configuration for the NWSAA Client is correct.

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 using real 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 real LU names, the specified mode is not defined. 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.

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.

CCL4678E   Environment variable *name* **is not set**

Explanation: The protocol driver requires the named environment variable to be defined.

System action: The Client SNA protocol driver is not loaded. Error messages are written to the SNA error log.

User response: Refer to the configuration documentation for the SNA transport product. A number of environment variables must be correctly defined for the correct operation of SNA protocol driver. Stop the Client, make sure the named variable is correctly defined in the environment where the Client will run; and then retry.

CCL4679E   Environment variable *name* **=** *value*

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

System action: The Client daemon continues, if possible.

User response: Check that the value of the *name* environment variable is correct.

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.

CTG5000E   There is not enough storage within the Gateway daemon address space to write an SMF record

Explanation: A call to the UNIX System Services API __smf_record failed with ENOMEM, which indicates that it did not have enough memory available.

System action: This message is logged. All data in the SMF record is lost.

User response: See the CICS Transaction Gateway Information Centre for details on dealing with out of memory errors.

CTG5002E   SMF is not active

Explanation: The Gateway daemon attempted to write a record to SMF. The Unix System Services call __smf_record failed with reason code JRSMFNotAccepting, indicating that SMF is not accepting records.

System action: This message is logged. All data in the current SMF record is lost.

User response: Contact the systems administrator responsible for SMF. For further details on the error code see the C/C++ Run-Time Library Reference book, and refer to the section describing the __smf_record() call.

CTG5009E   An error has occured in JNI Function getSysEnvData() with return and reason data = *INSERT-0*

Explanation: A call to retrieve information regarding the storage characteristics of the current address space has failed. The return code and reason code is included in the message text as encoded as a hexadecimal string. The most significant half-word represents the return code (0 = OK, 4 = WARNING, 8 = ERROR). The least significant half-word represents the reason code.

System action: This message is logged. The SE resource group statistics which are specific to the z/OS platform will not be updated.

User response: If the return and reason data is 0x00085008, then there has been a failure to allocate the memory required to store the data. See the CICS Transaction Gateway Information Centre for details on dealing with out of memory errors. If the messagereports any other non-zero return and reason code, contact your service organization.

CTG6010S   Severe Error - Cannot read parameters

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) has an internal error reading the passed parameters.

System action: The utility fails to run.

User response: Contact your service organization.

CTG6011E   Error - Too many parameters passed

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) takes either 1,2 or 3 parameters. See other messages for the command format.

System action: The parameters were incorrect, and the utility fails to run.

User response: Check the format of the command used and retry.

CTG6012E   Error - Empty parameter passed

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) takes either 1,2 or 3 parameters. See other messages for the command format.

System action: The parameters were incorrect, and the utility fails to run.

User response: Check the format of the command used and retry.

CTG6013E   Error - missing parameters

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) takes either 1,2 or 3 parameters. See other messages for the command format.

System action: The parameters were incorrect, and the utility fails to run.

User response: Check the format of the command used and retry.

CTG6014E   Bad command, only ″Register″ and ″Deregister″ supported

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) takes either 'R' or 'D' as its first parameter. See other messages for the command format.

System action: The parameters were incorrect, and the utility fails to run.

User response: Check the format of the command used and retry.

CTG6015E   Ready not accepted by ARM - see reason codes

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) called the 'Ready' function of Automatic Restart Manager (ARM) as one of the last phases of registering with ARM. ARM rejected the call.

System action: The utility will terminate.

User response: An explanation of the reason code returned by the 'Ready' request to ARM can be found in the 'z/OS MVS Programming: Sysplex Services Reference' manual.

CTG6016E   Registration failed - see reason codes

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) called the 'Registration' function of Automatic Restart Manager (ARM) as the first phase of registering with ARM. ARM rejected the call.

System action: The utility will terminate.

User response: An explanation of the reason code returned by the 'Registration' request to ARM can be found in the 'z/OS MVS Programming: Sysplex Services Reference' manual.

CTG6017E   Wait for predecessors failed - see reason codes

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) called the 'Wait' function of Automatic Restart Manager (ARM) as the 'Register' request to ARM indicated that this was a restart. ARM rejected the call.

System action: The utility will terminate.

User response: An explanation of the reason code returned by the 'Wait' request to ARM can be found in the 'z/OS MVS Programming: Sysplex Services Reference' manual.

CTG6018E   Deregistration failed - see reason codes

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) called the 'Deregistration' function of Automatic Restart Manager (ARM) to deregister with ARM. ARM rejected the call. This is a non-fatal error.

System action: The utility will terminate.

User response: An explanation of the reason code returned by the 'Deregister' request to ARM can be found in the 'z/OS MVS Programming: Sysplex Services Reference' manual.

CTG6020E   Deregistration failed - was not registered

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) called the 'Deregistration' function of Automatic Restart Manager (ARM) to deregister with ARM. ARM rejected the call because this job was not registered with ARM.

System action: The utility will terminate.

User response: No further action is required.

CTG6021E   Registration failed - ARM_ID already in use

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) called the 'Registration' function of Automatic Restart Manager (ARM) as the first phase of registering with ARM. ARM rejected the call because the ARM_ID was not unique in the sysplex.

System action: The utility will terminate.

User response: Choose a sysplex unique id and re-run the utility.

CTG6022E   Registration failed - ARM_ID is invalid

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) called the 'Registration' function of Automatic Restart Manager (ARM) as the first phase of registering with ARM. ARM rejected the call because the ARM_ID did not conform to the character restrictions.

System action: The utility will terminate.

User response: Check the value in the ARM_ID parameter against the restrictions given by other messages.

CTG6023E   Registration / Deregistration failed - ARM is not running

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) called the 'Registration' function of Automatic Restart Manager (ARM) as the first phase of registering with ARM. The call to ARM failed because ARM was not active on the sysplex.

System action: The utility will terminate.

User response: Investigate why ARM is not running.

CTG6024E   Registration / Deregistration failed - This is not a started job

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) called the 'Registration' function of Automatic Restart Manager (ARM) as the first phase of registering with ARM. ARM rejected the call because only started jobs can register.

System action: The utility will terminate.

User response: The utility may have been called from the USS command line or as part of a USS script. The utility can run successfully only when called from BPXBATCH with a PGM= parameter. Refer to the CTG Administration book for JCL examples.

CTG6025E   Registration failed - ARM_TYPE is invalid

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) called the 'Registration' function of Automatic Restart Manager (ARM) as the first phase of registering with ARM. ARM rejected the call because the ARM_TYPE did not conform to the character restrictions.

System action: The utility will terminate.

User response: Check the value in the ARM_TYPE parameter against the restrictions given by messages.

CTG6026E   Error - ARM_ID is too long - maximum of 16 characters.

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) Registration command takes a required parameter (ARM_ID) that is up to 16 characters long. See other messages for the command format.

System action: The parameter is incorrect, and the utility fails to run.

User response: Check the format of the command used and retry.

CTG6027E   Error - ARM_TYPE is too long - maximum of 8 characters.

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) Registration command takes an optional parameter (ARM_TYPE) that is up to 8 characters long. See other messages for the command format.

System action: The parameter is incorrect, and the utility fails to run.

User response: Check the format of the command used and retry.

CTG6051S   System Error with CSVQUERY, RC=*value*

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) has an internal error calling the CSVQUERY macro.

System action: The utility terminates.

User response: Contact your service organization.

CTG6052E   Error - ctgarm is not APF Authorized.

Explanation: CICS Transaction Gateway Automatic Restart Manager Batch Utility (ctgarm) must be APF authorized in order to register with ARM. This should happen during installation.

System action: The utility terminates.

User response: See the installation instructions in the CTG Administration book for details of how to APF authorize ctgarm.

CTG6103E   Command *command name* cannot be found in your PATH.

Explanation: The install requires the use of the command: *command name*, but this is not in the current PATH.

System action: The install will terminate.

User response: Identify where this command is on your system. Add its location to your PATH and retry the

install. See the Problem Determination section in the Administration book for your platform for further information.

CTG6106E   Language *language code* **is not recognized. Use command 'ctgmsgs' to list the recognized languages.**

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   Code set *code set* **is not recognized. Use command 'ctgmsgs' to list the recognized code sets.**

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 application will not start.

User response: See the Administration book for details of the versions of Java supported. Install the correct level of Java and add access to it from the PATH.

CTG6110E   CTGENVVAR file *HFS path* **not found. Check the value of environment variable 'CTGENVVAR'.**

Explanation: CICS Transaction Gateway on z/OS uses the 'ctgenvvar' 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 place. Note that the value of CTGENVVAR is a HFS path, and therefore case-sensitive.

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   Severe error: 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   Severe error: 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   Severe error: 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 **Error: 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 **Severe error: Unable to recognize the platform from the 'uname' string. 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 ctgconvenv cannot find the specified file, or the file does not have execute permission for this user. The syntax of the command is: ctgconvenv ['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 ctgconvenv 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.

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.

CTG6129E **Unable to find a running copy.**

Explanation: A running instance of the CICS Transaction Gateway could not be found by ctgd.

System action: The Gateway daemon, if running, does not stop.

User response: Ensure the administration port specified in ctgd.conf matches that of the running Gateway daemon, or manually kill the processes if the Gateway daemon is still running.

CTG6131W **Environment variable 'CICSCLI' is not set. Default location for CTG.INI is assumed.**

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 **Environment variable '_BPX_SHAREAS' is not set. Address space sharing will depend upon system defaults.**

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'. 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'.

CTG6205E **ctgasi - input was corrupted**

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) has an internal error and can not read the parameters passed to it.

System action: The utility fails to run.

User response: Contact your service organization.

CTG6206E **ctgasi - ASCRE failed for CTGRRMS, RC =** *return code*, **RSN =** *reason code*

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) has an internal error calling the ASCRE macro.

System action: The utility fails to run.

User response: Contact your service organization.

CTG6207E **ctgasi must be APF Authorized**

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) has not been installed correctly as it must be APF Authorized to execute.

System action: The utility fails to run.

User response: Refer to the SMP/E Program Directory.

CTG6208S **ctgasi - system error with CSVQUERY, RC =** *value*

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) has an internal error calling CSVQUERY macro.

System action: The utility fails to run.

User response: Contact your service organization.

CTG6209E **ctgasi - user ID is not authorized, SAFRC =** *SAF return code ACF return code ACF reason code*

Explanation: The user ID used to run the CICS Transaction Gateway Address Space Initiator Utility (ctgasi) has insufficient authority to the 'CTG.RRM.SERVICES' entity of the 'FACILITY' class.

System action: The utility fails to run.

User response: Check that the user ID has been granted the correct level of access. See the CICS TG Administration book for details.

CTG6210E **ctgasi - shutdown requested when address space not yet set up**

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) cannot shutdown the CTGRRMS services because they are not running.

System action: The utility ends.

User response: To start the CTGRRMS services, use the ctgasi command without the refresh or shutdown options.

CTG6211E   ctgasi - could not create the required name token pair

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) has an internal error creating a name token pair used by the CTGRRMS services.

System action: The utility fails to run.

User response: Contact your service organization.

CTG6212E   ctgasi - could not delete the required name token pair

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) has an internal error deleting a name token pair used by the CTGRRMS services.

System action: The utility fails to run.

User response: Contact your service organization.

CTG6213E   ctgasi - could not access the ECBs to control address space init

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) has an internal error accessing the event control blocks (ECBs) used to communicate between address spaces as part of the initialization of the CTGRRMS services.

System action: The utility fails to run.

User response: Contact your service organization.

CTG6214E   ctgasi - bad parameter passed: *parameter*

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) was called with an incorrect parameter *parameter* .

System action: The utility fails to run.

User response: See messages (CTG6221I onwards) for the command format.

CTG6215E   ctgasi - cannot refresh as users are still registered, count =*number*

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) has been called to do a refresh but there are still *number* users registered as using the CTGRRMS services. This can be caused by failing programs not deregistering from the CTGRRMS services.

System action: The utility ends.

User response: Do not refresh the CTGRRMS services while there are active users. If the count of users is incorrect, a refresh can be forced by using the '-f' option to override this check.

CTG6216E   ctgasi - ASCRE failed for CTGINIT, Post value = *return code*

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) has an internal error after calling ASCRE to initialize the CTGRRMS services. CTGINIT has posted the return code *return code* .

System action: The utility fails to run.

User response: Contact your service organization.

CTG6217E   ctgasi - dump called when address space not yet set up

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) was called to dump the internal trace table for CTGINIT. There is no trace table to dump out, because the CTGRRMS services have not been started.

System action: The utility fails to run.

User response: To start the CTGRRMS services, use the ctgasi command without the dump option.

CTG6218E   ctgasi - PCs not defined so can not check count of current users

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) has been called to do a refresh but there is an internal error with the CTGRRMS services. The CTGRRMS services cannot be used to determine the number of active users.

System action: The utility ends.

User response: You should not refresh the CTGRRMS services while there are active users, but because the services seem to be severely corrupted, a refresh can be forced by using the '-f' option to override this check.

CTG6219E   ctgasi - PC failed so can not check count of current users

Explanation: The CICS Transaction Gateway Address Space Initiator Utility (ctgasi) has been called to do a refresh but there is an internal error with the CTGRRMS services. The CTGRRMS services cannot be used to determine the number of active users.

System action: The utility ends.

User response: You should not refresh the CTGRRMS services while there are active users, but because the services seem to be severely corrupted, a refresh can be forced by using the '-f' option to override this check.

CTG6246E **ETCRE failed in CTGINIT, rc =** *return code*

Explanation: The CICS Transaction Gateway Resource Recovery Management Service services (CTGRRMS) have failed due to an internal error.

System action: The services have failed.

User response: Contact your service organization.

CTG6247E **CTGINIT failed as ASPARM is too small, size =** *actual size*, **req =** *minimum size*

Explanation: The CICS Transaction Gateway Resource Recovery Management Service services (CTGRRMS) have failed due to an internal error.

System action: The services have failed.

User response: Contact your service organization.

CTG6248E **ASEXT failed in CTGINIT, rc =** *return code*, **rsn =** *reason code*

Explanation: The CICS Transaction Gateway Resource Recovery Management Service services (CTGRRMS) have failed due to an internal error.

System action: The services have failed.

User response: Contact your service organization.

CTG6249E **CTGINIT failing as version** *Services version* **is less than requested** *Gateway version*

Explanation: The CICS Transaction Gateway Resource Recovery Management Service services (CTGRRMS) have failed because the version of CICS Transaction Gateway trying to initialize the services (ctgasi) is more recent than can be supported by the back level code (CTGINIT) in the LNKLST.

System action: The services have failed.

User response: Ensure that the most recent version of CTGRRMS services (CTGINIT) is in the LNKLST. See the CTG Administration book for details.

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 ouptut 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 valuewhich 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 limitor, increase the EXCI logon limit to greater than, or equal to, the number of Worker threads.

CTG6407E   Unable to parse statistics interval length (statint) value *statint value* **. Statistics interval length is set to the default**

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 (000001) 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   Unable to parse statistics end of day (stateod) value *stateod value* **. Statistics end of day is set to the default**

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

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.

CTG6421W   Health value has been set to 0.

Explanation: Communications problems between the Gateway daemon and CICS have been reported to the workload manager. No new work will be sent to the Gateway daemon until the problems are resolved and the health value is reset.

System action: The Gateway daemon continues.

User response: 1. Check the logs to determine the cause of the communications problems and take corrective action. 2. Reset the health value; see the information center for details.

**CTG6423W   Health reporting cannot be enabled. rc = ** *INSERT-0*

Explanation: The configuration file contains the setting healthreporting=ON, indicating that the state of communications between the Gateway daemon and CICS should be reported to the TCP/IP workload manager. An error with return code $1 prevents health reporting from being enabled.

System action: The Gateway daemon continues.

User response: Investigate the cause of the problem. If health reporting cannot be enabled on your system, for example because the version of z/OS does not support it, remove the entry from the configuration file.

CTG6484W   keyringpw supplied without keyring. Parameter ignored

Explanation: The keyringpw parameter has been supplied without a user keyring also being specified.

System action: CICS Transaction Gateway ignores the keyringpw parameter and continues the startup process.

User response: In order to use the keyringpw parameter, a keyring must be specified.

CTG6488E   Request rejected. Gateway daemon is shutting down

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.

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.

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. [*ConnectionManager*] [*count*]

Explanation: A client has disconnected from ConnectionManager *ConnectionManager*, but *count* work requests are still in progress.

System action: A ConnectionManager 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 ConnectionManager *name*, but *count* requests which require completion are still outstanding.

System action: The ConnectionManager will attempt to clean up the allocated resources.

User response: For information only.

CTG6525E   Unable to start handler for the *protocol* : protocol. [*RC*]

Explanation: Where *protocol* specifies the protocol type, 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.

CTG6528W   Outstanding work still in progress. [*name*] [*count*]

Explanation: A client has disconnected from ConnectionManager *name*, with work requests in progress. *count* requests have failed to finish within a reasonable time.

System action: The ConnectionManager 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: ConnectionManager 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.

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 occuring 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.

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 NONAMES=ON, to avoid this. Refer to the CICS Transaction Gateway Administration book for further details.

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 ConnectionManager 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 ConnectionManager and Worker threads initially created.

CTG6552E Error whilst accepting a connection from *server* **. Connection closed. [***error***]**

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. [*ConnectionManager***]**

Explanation: Where *ConnectionManager* specifies the ConnectionManager 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 protocol handlers started successfully.

Explanation: Gateway cannot access any protocol handlers.

System action: If no protocol handlers can be started, the Gateway will exit.

User response: Check the CTG initialization file for protocol handlers; uncomment the protocol handle you want.

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: ConnectionManager *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 ConnectionManagers.**

Explanation: A connection was accepted from *name*, but a ConnectionManager did not become available in time.

System action: The connection was rejected.

User response: Increase the maximum number of ConnectionManagers.

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   *protocol* **protocol handler has been stopped because of persistent errors. Restart will be attempted.**

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.

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.

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.

CTG6589W   **The CICS Transaction Gateway for z/OS does not support the adminport parameter. It has been ignored**

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 occuring in future, remove the adminport 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, providing you run the Configuration tool on z/OS, or specify the -plat z/OS or -plat 390 option if running it on a different platform.

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.

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 running the Gateway daemon, check whether the Java Client application has specified the correct parameters to connect to the Gateway daemon.

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   Flow did not contain the correct eyecatcher. [Received *error* **] Fatal Error**

Explanation: Network connection did not contain the correct identifier.

System action: Connection closed.

User response: For information only.

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 request 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 Cannot open JavaGateway. 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.

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.

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.

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.

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 **CICS Request: Error in initialization. [***error***]**

Explanation: Where *error* specifies the Java error.

System action: CICS Request revolted.

User response: Check the Java error.

CTG6717E **CICS Request: An attempt to convert from a C byte array to a Java String failed. The codepage specified for the conversion was *code page* .**

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 **CICS Request: Error. [***error***]**

Explanation: Where *error* specifies the error.

System action: CICS Request revolted.

User response: Check error.

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   ECIRequest: 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.

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.

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.

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.

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 classpath 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 classpath 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: System action is not required.

User response: For information only.

CTG6791E   Internal error - terminal state invalid.

Explanation: An internal error has occurred. The terminal state is incorrect.

System action: System action is not required.

User response: For information only.

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 (1} 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.

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.

CTG6806E   Register with RRS failed. Return code = ** *rc* **. Name = ** *name* **.

Explanation: The return code from the RRMS resource manager registration call (crggrm). The return code is in hexadecimal.

System action: CICS Transaction Gateway processing continues.

User response: If the return code is non-zero, contact your service organization for more information and diagnosis, or see the z/OS MVS Programming: Resource Recovery book, for details of the return code and any corrective action which should be taken. Common failing return codes are: 700 - CRG_RM_NAME_REGISTERED - The resource manager is already registered. 300 - CRG_RM_NAME_INVALID - The resource manager name is invalid. If the CICS TG registers as an unauthorized resource manager a .IBM.UA suffix is required. See the administration guide for details on which configurations the CICS TG registers as unauthorized and which characters are valid in an RM name.

CTG6807E   Set exitcode with RRS. Return code = ** *rc* **. Name = ** *name* **.

Explanation: This displays the return code from the RRMS set exit information call (crgseif). The return code is in hexadecimal.

System action: If the return code is non-zero the Gateway daemon or ctgmaster process which encountered the problem fails to start.

User response: If the return code is non-zero, contact your service organization for more information and diagnosis, or see the z/OS MVS Programming: Resource Recovery book, for details of the return code and any corrective action which should be taken.

CTG6808E   Authorize userid/password with RACF. User ID = ** *userid* **, return code = ** *rc* **, errno = ** *error number* **, errno2 = ** *reason code* **.

Explanation: If a zero length password is given then the return code is -2, otherwise this displays the return code from the __passwd call used to authenticate the username and password. If the return code is not zero then errno and errno2 were set by the call. Refer to the problem determination section of the z/OS Administration Guide for further details of security errors. Here is a brief list of possible return codes: 111 - EACCES - The password was not authorized. 121 -

EINVAL - The userid or password was invalid. 143 - ESRCH - The userid was unknown or not defined to the kernel. 157 - EMVSERR - The __passwd is not supported in an address space where a load was done from an uncontrolled library. 163 - EMVSSAFEXTRERR - The userid access has been revoked. 164 - EMVSSAF2ERR - Internal processing error. 168 - EMVSEXPIRE - The password has expired. 169 - EMVSPASSWORD - The new password was not valid, or did not meet the installation-exit requirements.

System action: If the return code is non-zero, a return code of ECI_SECURITY_ERROR will be returned to the application.

User response: If the return code is non-zero, see the C/C++ Run-Time Library Reference book, and refer to the section describing the __passwd() call. More detail on the reason code can be found in the UNIX System Services Messages and Codes book.

CTG6815E   Retrieve Side Information Fast failed. RRM Return code = ** *rc* **.

Explanation: The call to Retrieve Side Information Fast (atrrusf) failed with the specified return code. The return code is in hexadecimal.

System action: A system error is returned on the associated ECI request.

User response: Refer to the z/OS MVS Programming: Resource Recovery book, for details of the return code and any corrective action which should be taken. If the problem persists, contact your service organization.

CTG6816E   Retrieve Current Context Token failed. RRM Return code = ** *rc* **.

Explanation: The call to Retrieve Current Context Token (ctxrcc) failed with the specified return code. The return code is in hexadecimal.

System action: A system error is returned on the associated ECI request.

User response: Refer to the z/OS MVS Programming: Resource Recovery book, for details of the return code and any corrective action which should be taken. If the problem persists, contact your service organization.

CTG6817E   Switch Context failed. RRM Return code = ** *rc* **.

Explanation: The call to Switch Context (ctxswch) failed with the specified return code. The return code is in hexadecimal.

System action: A system error is returned on the associated ECI request.

User response: Refer to the z/OS MVS Programming: Resource Recovery book, for details of the return code and any corrective action which should be taken. If the

problem persists, contact your service organization.

CTG6818E   Begin Context failed. RRM Return code = *rc* .

Explanation: The call to Begin Context (ctxbegc) failed with the specified return code. The return code is in hexadecimal.

System action: A system error is returned on the associated ECI request.

User response: If the return code is non-zero, contact your service organization for more information and diagnosis, or refer to the z/OS MVS Programming: Resource Recovery book, for details of the return code and any corrective action which should be taken. A common failing return code is: 756 - CTX_AUTH_FAILURE - The resource manager is PKM 8-15 problem state and specified a resource manager token that does not belong to a PKM 8-15 problem state resource manager registered in the home address space. This may be due to CTG6806E with return code 700 - CRG_RM_NAME_REGISTERED - The resource manager is already registered.

CTG6819E   End Context failed. RRM Return code = *rc* .

Explanation: The call to End Context (ctxendc) failed with the specified return code.

System action: A system error is returned on the associated ECI request.

User response: Refer to the z/OS MVS Programming: Resource Recovery book, for details of the return code and any corrective action which should be taken. If the problem persists, contact your service organization.

CTG6820E   Error in GetStringPlatform function. Return code = *rc*, data area = *data area*, length = *length* .

Explanation: The call to GetStringPlatform has failed with the specified return code and information. This is an internal error.

System action: CICS Transaction Gateway processing continues.

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

CTG6821E   Error in GetStringPlatformLength function. Return code = *rc* .

Explanation: The call to GetStringPlatformlength has failed with the specified return code and information. This is an internal error.

System action: CICS Transaction Gateway processing continues.

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

CTG6822E   EXCI function error. Function Call = *function call*, Response = *response code*, Reason = *reason code*, Subreason field-1 = *subreason field 1*, subreason field-2 = *subreason field 2*, Cics_Rc = *rc* .

Explanation: The EXCI call failed with the error codes displayed.

System action: An error is returned to the Java Client application.

User response: Refer to the CICS External Interfaces Guide for further information on the error. The EXCI function being performed is specified by the Function Call data in the message, which maps to the EXCI calls as follows; 1=Initialize_User, 2=Allocate_Pipe, 3=Open_Pipe, 4=Close_Pipe, 5=Deallocate_Pipe, 6=DPL_Request.

CTG6823E   EXCI DPL_REQUEST specific error. RESP value = *response code*, RESP2 value = *response code 2*, Abend Code = *abend code*, Cics_Rc = *rc* .

Explanation: A request failed with a DPL specific error. The error codes returned by the DPL_REQUEST are displayed.

System action: An error is returned to the Java Client application.

User response: Refer to the CICS External Interfaces Guide for further information on the error.

CTG6824E   Attempt to change JNI trace filename while trace is running.

Explanation: An attempt was made to change the JNI trace filename whilst trace was running.

System action: This message is logged. Tracing continues to the previous filename.

User response: Deactivate JNI trace before changing the filename. If you want to enable JNI tracing when the CICS Transaction Gateway starts, ensure that you set the trace parameters only once. For example, do not specify both an environment variable setting and a Java directive setting.

CTG6825E   Unsupported call type. Call_Type = *call type*, Cics_Rc = *cics rc* .

Explanation: This message is issued when an invalid EXCI call is being attempted. The only valid call types for use with EXCI are ECI_SYNC, ECI_STATE_SYNC and ECI_SYNC_TPN.

System action: An error is returned to the Java Client application.

User response: Use a call type that is supported on the z/OS platform.

CTG6826E **Invalid commarea length. Commarea_Length =** *commarea length*, **Size of Commarea =** *commarea size*, **Cics_Rc =** *rc* .

Explanation: The commarea length defined, or the actual length of the commarea is too large, and is therefore invalid. The maximum length supported is 32659 bytes.

System action: An error is returned to the Java Client application.

User response: Ensure that the maximum length of the commarea used does not exceed 32659 bytes.

CTG6827E **Error message from CICS:** *cics_error*

Explanation: An error message has been returned directly from the CICS region, and is displayed.

System action: CICS Transaction Gateway processing continues.

User response: Refer to the CICS Messages and Codes book, for details of the error.

CTG6869E **Change trace file name failed. filename =** *file name* .

Explanation: An attempt to change the JNI Trace file name to *file name* has failed.

System action: A TFileException is thrown. CICS Transaction Gateway processing continues. JNI tracing continues using the existing file name.

User response: User code should handle the exception. Check that the file name is valid and can be written to.

CTG6873E **Already initialized.**

Explanation: EXCI connection reuse has already been initialized.

System action: The CICS Transaction Gateway will return an ECI_ERR_SYSTEM_ERROR error.

User response: Contact your service organization for more information and diagnosis.

CTG6874E **Memory allocation failure for TCB table for** *number* **TCB entries.**

Explanation: The CICS Transaction Gateway was unable to allocate memory for the number of TCB entries specified.

System action: The CICS Transaction Gateway returns an ECI_ERR_RESOURCE_SHORTAGE error.

User response: Ensure that the CICS Transaction Gateway has enough memory available to it to operate correctly.

CTG6875W **EXCI warning. Function Call =** *function*, **Response =** *response*, **EXCI Reason =** *return code*, **Subreason field-1 =** *return code*, **subreason field-2 =** *return code* .

Explanation: An EXCI call has returned a warning.

System action: CICS Transaction Gateway processing continues.

User response: Diagnose the warning by referring to the CICS External Interfaces Guide.

CTG6876E **EXCI error. Function Call =** *function*, **Response =** *response*, **EXCI Reason =** *return code*, **Subreason field-1 =** *return code*, **subreason field-2 =** *return code*, **ctg_rc=***error* .

Explanation: An EXCI call has returned an error.

System action: CICS Transaction Gateway processing continues.

User response: Diagnose the error by referring to the CICS External Interfaces Guide.

CTG6877E **Unknown tracepoint! num =** *number* .

Explanation: JNI Tracing has sent an unknown trace point to the formatter.

System action: Cannot be sure what will happen as we do not know this trace point.

User response: Contact your service organization.

CTG6878E **Surrogate check failed. SURROGCHK=yes set in DFHXCOPT. EXCI Reason =** *return code*, **Subreason field-1 =** *return code*, **subreason field-2 =** *return code* .

Explanation: The surrogate user check has failed. The MVS External Security Manager's return code and reason code are subreason codes 1 and 2.

System action: The ECI request is rejected.

User response: Check whether SURROGCHK has been set to YES in DFHXCOPT, and that the credentials are accurate.

CTG6882E **No pipe available. An attempt was made to open a pipe to server** *server name***, but no free receive sessions were available. CTG currently holds** *total* **pipes.**

Explanation: An attempt has been made to open a pipe. The target CICS system associated with the pipe has no free receive sessions. The server name field *server name* identifies the target CICS server specified on the ECI call.

System action: The call fails, and DFHXCURM (replaceable) is invoked. If the target CICS system is changed by DFHXCURM the change will not be reflected by the text of this message.

User response: Retry the request to CICS when the system is under less load. Review the Receivecount setting of the CICS session definition.

CTG6883E **Call to make address space non-swappable failed with return code =** *rc***, errno =** *error number***, errno2 =** *reason code* **.**

Explanation: The error number and error reason will indicate the cause of the failure. Detailed descriptions of these errors can be found in the z/OS C/C++ Run-Time Library Reference under the __mlockall() library function. If the user ID running the CICS Transaction Gateway does not have READ access to BPX.STOR.SWAP, errno is set to 139 - EPERM - and the reason code 0X9300000.

System action: The CICS Transaction Gateway has been unable to make the address space non-swappable and continues to run in a swappable mode.

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

CTG6888W **There are** *remaining* **pipes remaining out of a total of** *total* **available EXCI pipes.**

Explanation: This message is output when approximately 90% of available pipes are in use.

System action: The message is logged and the CICS Transaction Gateway continues.

User response: If you have not planned your system for the pipe usage to be as high at this point, consider increasing the number of pipes available to the CICS Transaction Gateway. Alternatively the CICS Transaction Gateway offers a choice of pipe caching models, which can reduce pipe usage in some circumstances. See the Administration book for more details.

CTG6889W **All of the available EXCI pipes are in use.**

Explanation: This message is output when all the available pipes are in use.

System action: The message is logged and the CICS Transaction Gateway continues.

User response: If the CICS Transaction Gateway attempts to allocate any more pipes, requests will start to fail with resource shortage errors. If you are not expecting all pipes to be in use at this point, consider increasing the total number of pipes available to the CICS Transaction Gateway. Alternatively, the CICS Transaction Gateway offers a choice of pipe caching models, which can reduce pipe usage in some circumstances. See the Administration book for more details.

CTG6891E **EXCI pipe limit value** *num pipes* **is out of range. The default value will be used instead.**

Explanation: This message is logged if the value specified to CICS for the maximum number of EXCI pipes is outside the allowed range (100 to 250).

System action: This message is logged, the maximum number of EXCI pipes available is assumed to be the default of 100 and the CICS Transaction Gateway continues.

User response: Reconfigure the maximum number of pipes under CICS to a value within the allowed range.

CTG6896E **Unexpected return code** *return code* **for IEANTRT call. Unable to determine EXCI logon limit.**

Explanation: This message is logged if a call to the z/OS name token retrieval service fails with an unexpected error. The name of the assembler service call which failed is IEANTRT and *return code* is the value this call returned. See the z/OS Assembler Services reference for details on IEANTRT return codes.

System action: This message is logged and the maximum number of EXCI pipes available is assumed to be the default of 100. The CICS Transaction Gateway continues.

User response: If this message persists, or a value other than the default has been specified for the maximum number of concurrent pipes, contact your service organization.

CTG6899E **Unable to start JNI trace, filename** *filename* **errno** *errno* **.**

Explanation: The CICS Transaction Gateway could not open trace file *filename* for writing. *errno* is the operating system return code for the failure.

System action: This message is logged and tracing is deactivated. No trace points will be written.

User response: Investigate why the CICS Transaction Gateway cannot to write to *filename* . For example, check that the user ID under which the CICS Transaction Gateway is running has write access to the file and directory specified.

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 allowserver 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 organisation.

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. 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 **Intialization failed. 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.

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: CICSTERM or CICSPRNT did not start because the Client daemon was not, or could not be, started.

System action: The emulator terminates.

User response: Try to start the Client daemon using the CICSCLI /S command.

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/400 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.

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.

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.

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.

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.

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: client will continue but exits will not be used

User response: Define the exits as defined in writing your own exits (in info centre)

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.

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.

CCL8024E   Unable to communicate with the Client daemon

Explanation: CICSCLI could not communicate with the Client daemon process. This probably means that the Client daemon was unable to start.

System action: CICSCLI performs no action.

User response: Try starting the Client daemon using the CICSCLI /S command.

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.

CCL8036E   Unable to perform start request, see Client error log

Explanation: The Client daemon was unable to complete the request to start the connection to a 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.

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.

CCL8051E   Invalid request to change security information

Explanation: A CICSCLI command was issued with the /U or /P options but without the /C 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.

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.

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.

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.

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.

CTG8233E   **Cannot specify** *action* **and its short form** *abreviation* **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.

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.

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 that 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.

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.

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.

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.

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.

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 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* **. 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.

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.

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 isonly 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 ÔÇÿNAMEÔÇÖ 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.

CTG8431E   Handshake failure for IPIC connection to CICS server *CICS Server name as defined in the INI file.* **response code=***Response code from CICS***, reason=***Reason for response from CICS.* **[***Numeric reason code from CICS.* **].**

Explanation: The physical socket connection was established, but CICS rejected the connection.

System action: Any requests that were attempting to use the CICS server *CICS Server name as defined in the INI file.* 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 *CICS Server name as defined in the INI file.* is defined correctly in the INI file (remote mode) or Server URL(local mode) and check the reason text *Response code from CICS* . Check the console log for the CICS server. Check topic ″DFHIS1011″ in the CICS infocenter

for the meaning of the return code.

CTG8432W   CICS has purged IPIC conversation *IPIC Conversation ID used by CICS to identify work.* **to CICS server** *CICS Server name as defined in the INI file.* **with Mirror task=***Mirror task in CICS that was used for this transaction.***, Transaction ID:** *XID or IPIC LUW token (if relevant) for the transaction.* **.**

Explanation: The CICS connection defined as *CICS Server name as defined in the INI file.* 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) for the transaction.* and mirror transaction *Mirror task in CICS that was used for this transaction.* .

System action: CICS Transaction Gateway will end the work and free each purged conversation to CICS.

User response: >

CTG8433E   Connection failure for IPIC connection to CICS server *CICS Server name as defined in the INI file.* **reason=***Failure 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 *CICS Server name as defined in the INI file.* 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 INI file (remote mode) or Server URL(local mode) and check the reason text *Failure reason text.* .

CTG8434W   Error reading IPIC Server Definition *IPIC Server name as defined in the INI file.* **from ctg.ini. Reason=***Failure reason text.* **.**

Explanation: An error occurred while reading the IPIC server definition.

System action: The Gateway will ignore the named server definition.

User response: Check that the IPIC server is defined correctly in the INI file.

CTG8435W   Duplicate IPIC Server Definition *IPIC Server name as defined in the INI file.* **in ctg.ini.**

Explanation: The named IPIC server definition is a

duplicate of another definition within the ini file.

System action: The Gateway will ignore the duplicate server definition.

User response: Check that the IPIC server is defined correctly in the 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.

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.

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.

CTG8603E **Lookup failure for XID entry (xmt_rc=***xmt_rc* **rc=***rc* **).**

Explanation: Ths CICS Transaction Gateway was unable to map an XA standard XID to a RRMS unit of recovery.

System action: This message is logged, or traced, and the CICS Transaction Gateway continues.

User response: If this message is logged and the problem perisists contact your service organization.

CTG8605E **Error when deleting XID entry (xmt_rc=***reason code* **xa_rc=***reason code* **).**

Explanation: The CICS Transaction Gateway was unable to remove a reference to the XID from the XID mapping table. If the table is shared, a common error would be that the ctgmaster process has been ended prematurely.

System action: The entry will remain in the mapping table.

User response: If the problem persists, contact your service organisation. Recycle the CICS TG or if IP Spraying, recycle the ctgmaster and then the CICS TG which will cause the entry to be freed up.

CTG8607E **Authorized RRS Express_UR_Interest call failed. RRMS_RC=***rrms_rc* **.**

Explanation: A call to the RRMS API (ATREINT1) failed. RRMS returned the return code *rrms_rc* . The CICS Transaction Gateway was unable to express interest in a transactional unit of recovery.

System action: The CICS Transaction Gateway resource adapter returns a resource manager failure error (XAER_RMERR) on the transaction. The transaction fails to start and no work can be performed against it.

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

CTG8609E   Authorization error when executing *RRMS function* **(**auth return code **).**

Explanation: In order to execute *RRMS function*, the CICS Transaction Gateway uses CTGRRMS services. Attempting to call the service failed with error *auth return code* . Return code values are: * 0x04 - Version number check failed (RRMSInit call) * 0x08 - Parameter length checks failed * 0x0C - Security check failed * 0x10 - Internal error (ASID table) * 0x14 - Function not supported

System action: The RRMS function is not executed and a resource manager failure error will be returned to the calling application.

User response: The CICS Transaction Gateway administration guide contains details on the mandatory steps required to be able to use the CTGRRMS services.

CTG8610E   RRS ATRAPRP Prepare_Agent_UR call failed (RRMS_RC=rrms return code **).**

Explanation: A call to the RRMS API to prepare a transaction failed with return code *rrms return code* .

System action: If *rrms return code* is non zero, this error is logged and the CICS Transaction Gateway resource adapter returns a failure return code to the transaction manager. The transaction manager may attempt to re-issue the prepare or may back out the transaction.

User response: Check the Gateway daemon log for further log messages that may more specific information on the cause of the problem. If the problem persists, contact your service organization.

CTG8611E   One-phase-commit RRS Commit call ATRCMIT failed (RRMS_RC=return code **).**

Explanation: A request to RRMS to commit a transaction failed. The transaction is a one-phase-commit optimized XA transaction.

System action: The resource adapter returns a serious error to the transaction manager for the affected transaction, which might not be completed. The CICS Transaction Gateway continues to process transactions.

User response: The RRMS system administrator should check the status of the transaction, and manually complete it if necessary. Check the Gateway daemon log for further messages related to this transaction.

CTG8613E   Two-phase-commit Commit_Agent_UR RRS call failed. (RRMS_RC=return code **).**

Explanation: A request to RRMS to commit a two-phase-commit XA-based transaction failed. The

return code indicates that a repeated request is not likely to succeed.

System action: The resource adapter returns a serious error to the transaction manager for the affected transaction, which might not be completed. The CICS Transaction Gateway continues to process transactions.

User response: The RRMS system administrator should check the status of the transaction, and manually complete it if necessary. Check the Gateway daemon log for further messages related to this transaction.

CTG8620E   Error in GetByteArrayElements function.

Explanation: Unable to transfer XID data between Java and C. This would normally denote an environment issue.

System action: The XA request is rejected.

User response: Contact your service organisation to resolve this problem.

CTG8621E   Error calling JNI (rc = reason code **).**

Explanation: CICS TG has been unable to determine the XA options used on the transaction request.

System action: The request is rejected and no state-change is made on the transaction.

User response: Contact your service organisation.

CTG8624E   Error during Set_Syncpoint_Controls call (RRMS_RC=rrms_rc **).**

Explanation: A request to the RRMS API set syncpoint controls failed. The Gateway daemon was unable to set itself as the correct type of syncpoint manager with RRMS in order to peform actions on a transaction.

System action: It might be not be possible to prepare, commit or backout the transaction. In this case the resource adapter will return an error to the transaction manager. The CICS Transaction Gateway continues to process requests for other transactions.

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

CTG8642E   Error storing XID (xmt_rc = reason code **).**

Explanation: The CICS TG has received a request to start a transaction, but could not store temporary information needed to identify the transaction.

System action: The CICS TG cannot start the transaction.

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

CTG8643E   The Gateway daemon has become unregistered as a RRMS resource manager.

Explanation: Either RRMS has been restarted, or the master process (ctgmaster) terminated whilst this instance of the Gateway daemon was running.

System action: All transactions through this instance of the Gateway daemon will fail.

User response: If not using load balancing restart this instance of the Gateway daemon. If load balancing restart every instance of the Gateway daemon in the load balancing group and restar the master process. See the administration guide for details on the correct shutdown and restart sequence.

CTG8644E   RRMS encountered an unexpeced error whilst processing a unit of recovery (RRMS_RC=*rrms_rc*).

Explanation: An RRMS API return code indicated that an unexpected error occurred whilst processing a unit of recovery, and that the transaction may have been damaged.

System action: The resource adapter will return a serious error to the transaction manager for the affected transaction. The CICS Transaction Gateway continues to process other transactions.

User response: The RRMS administrator should view the transaction and make any changes necessary.

CTG8645E   Transaction Failure requires that the UR is resolved manually. (RC=*rrms_rc*).

Explanation: Due to an internal processing error, it is not possible for the Gateway daemon to commit, or backout a transaction.

System action: The resource adapter will return a serious error to the transaction manager for the transaction in question. The CICS Transaction Gateway continues to process other transactions.

User response: The RRMS administrator should resolve the transaction manually.

CTG8651W   RRM name exceeds maximum length of 32 characters.

Explanation: CTG_RRMNAME parameter exceeded 32 characters in length.

System action: The name is truncated to 32 characters before registering with RRMS. This message is output and the CICS Transaction Gateway continues.

User response: To avoid this message on startup,

specifiy a name less than or equal to 32 characters.

CTG8659E   Unable to initialize CTGRRMS services (return code=*return code*)

Explanation: To support XA based requests, the CICS Transaction Gateway issues authorized RRS calls, through CTGRRMS services. It was not possible to enable these services. Return code values are: * 0x04 - Version number check failed (RRMSInit call) * 0x08 - Parameter length checks failed * 0x0C - Security check failed * 0x10 - Internal error (ASID table) * 0x14 - Function not supported

System action: This message is logged. The CICS Transaction Gateway does not start up.

User response: If XA support is required, see the information center for information on how to start CTGRRMS services. If XA support is not required, set the xasupport configuration parameter off and restart the CICS Transaction Gateway.

CTG8663W   Deregister from CTG RRMS Services failed with rc = *return code*

Explanation: The CICS Transaction Gateway or CTGMASTER was unable to deregister from the CTGRRMS Services during the shutdown process.

System action: This message is logged and the CICS Transaction Gateway process ends.

User response: If the CTGRRMS services are to be refreshed or stopped, use the -f option, because this Gateway instance or CTGMASTER process will still be shown as registered.

CTG8664E   No CTGMASTER is running with an RRM name of: *name*

Explanation: This CICS Transaction Gateway has been configured as part of a TCP/IP load balancing group but the CTGMASTER process is not running. The CTGMASTER process should use the name *name* for RRS registration.

System action: This message is logged. The CICS Transaction Gateway does not start up.

User response: Configure and start a CTGMASTER instance for the name given in the CTG_MASTER_RRMNAME environment variable and restart the CICS Transaction Gateway.

CTG8665E   Invalid XA commit option (*option*)

Explanation: An internal error has occurred, the XA commit option specified is invalid.

System action: It is not possible for the CICS Transaction Gateway to commit the UR. A serious error is returned to the transaction manager.

User response: The RRMS administrator should resolve the transaction manually. If the problem persists, contact your service organisation.

CTG8668W **A problem occured while writing to the JNI trace file.**

Explanation: The CICS Transaction Gateway encountered and error while writing to the JNI trace file.

System action: This message is logged and tracing is deactivated. No further trace points will be written.

User response: Investigate why the CICS Transaction Gateway can no longer write to the trace file. For example, check that the file system where the file is written has sufficient free space.

CTG8700E **ctgmaster failed to initialize**

Explanation: Ctgmaster could not start.

System action: The ctgmaster process ends.

User response: Check the logs for error messages and correct any reported problems.

CTG8804E **The CICS Client/Gateway is not installed properly**

Explanation: The CICS Transaction Gateway or CICS Universal Client 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 or CICS Universal Client.

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 or HKEY_LOCAL_MACHINE\SOFTWARE\IBM\CICS Universal Client.

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 or HKEY_LOCAL_MACHINE\SOFTWARE\IBM\CICS Universal Client. Then repeat the command line option.

CTG8807E **Unable to find a JVM**

Explanation: When trying to run a Java application, the application launcher tries to find a JVM, it does this by querying the registry.

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.

CTG8815E **File does not exist**

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.

CTG8818E **The JVM specified is unsupported**

Explanation: When trying to point to a Java Virtual Machine using the CTGJAVA command, the specified JVM is tested and determined to be at 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 application launcher tests the currently set JVM. It is determined to be at an unsupported level.

System action: The command line application terminates.

User response: Use the CTGJAVA command to point to a supported Java Virtual Machine to run.

CTG8820E   Detected Operating System is not supported for this application

Explanation: The operating system you are running, as detected by the JVM, is not supported.

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. On Windows platforms this failure can also occur when starting the IBM CICS Universal Client service.

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. Additionally on Windows operating systems check the application event log for messages generated by the IBM CICS Universal Client service.

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 organisation.

CTG8923W   Environment variable AUTH_USERID_PASSWORD is set to the incorrect value *INSERT-0* ; user ID and password authentication is enabled.

Explanation: The AUTH_USERID_PASSWORD environment variable contains an incorrect value. User IDs and passwords will be authenticated against the appropriate security mechanisms.

System action: User ID and password authentication is enabled.

User response: Set the environment variable AUTH_USERID_PASSWORD to either YES or NO. Set this environment variable to YES to enable user ID and password authentication. Set it to NO to disable user ID and password authentication.

CTG8924W   Environment variable CTG_MIXEDCASE_PW is not set; mixed-case password support is not enabled.

Explanation: The CTG_MIXEDCASE_PW environment variable controls whether or not passwords are converted to upper case before authentication. It is required only if AUTH_USERID_PASSWORD is set.

System action: All passwords will be converted to upper case before authentication takes place.

User response: Set the environment variable CTG_MIXEDCASE_PW to either 'YES' or 'NO'. Set this environment variable to YES to enable mixed case passwords. Set it to NO to convert passwords to upper case.

CTG8927W   Environment variable CTG_MIXEDCASE_PW is set to the incorrect value *INSERT-0* ; mixed-case password support is not enabled.

Explanation: The CTG_MIXEDCASE_PW environment variable contains an incorrect value. All passwords will be converted to upper case before authentication takes place.

System action: All passwords will be converted to upper case before authentication takes place.

User response: Set the environment variable CTG_MIXEDCASE_PW to either YES or NO. Set this environment variable to YES to enable mixed case passwords. Set it to NO to convert passwords to upper case.

CTG8929W   Environment variable AUTH_USERID_PASSWORD is not set; user ID and password authentication is not enabled.

Explanation: The environment variable AUTH_USERID_PASSWORD controls whether or not user IDs and passwords supplied on CICS requests are authenticated.

System action: User ID and password authentication is not enabled.

User response: Set the environment variable AUTH_USERID_PASSWORD to either 'YES' or 'NO'. Set this environment variable to YES to enable user ID and password authentication. Set it to NO to disable user ID and password authentication.

CTG8936W   ctgmaster -input switch value (*value*) is not valid and has been ignored

Explanation: The ctgmaster program was invoked with the -input switch, but *value* is not a valid value.

System action: Execution of ctgmaster continues,

using the console type defined in the CTG_MASTER_INPUT environment variable. If the environment variable is undefined or not valid, the default of STDIN is used instead.

User response: Check that the -input switch specifies a string value of STDIN if running in the USS shell, or CONSOLE if running in batch mode.

CTG8937W   ctgmaster -xa_max_tran switch value (*value1* **) is out of range. xa_max_tran is unchanged (***value2* **)**

Explanation: The ctgmaster program was invoked with the xa_max_tran switch, but *value1* is not a valid value.

System action: Execution of ctgmaster continues, using the value defined in the CTG_XA_MAX_TRAN environment variable. If the environment variable is undefined or not valid, the default value of 8192 is used instead.

User response: Check that the -xa_max_tran switch specifies an integer between 1 and 8192.

CTG8938W   ctgmaster -trace switch did not specify a destination path and has been ignored

Explanation: The ctgmaster program was invoked with the -trace switch, but the value specified is not valid.

System action: Execution of ctgmaster continues, using the value defined in the CTG_MASTER_TRACE environment variable. If the environment variable is undefined or not valid, trace output is written to STDERR.

User response: Check that the -trace switch specifies a string value that represents a valid HFS file name, on a path to which the user has write access.

CTG8939W   ctgmaster -natlang switch value (*value* **) is not valid and has been ignored**

Explanation: The ctgmaster program was invoked with the -natlang switch, but *value* is not a valid value.

System action: Execution of ctgmaster continues, using the value defined in the CTG_MASTER_NATLANG environment variable. If the environment variable is undefined or not valid, the default English (EN) NLS message bundle is used instead.

User response: Check that the -natlang switch is one of the following two-character strings: EN (English) JA (Japanese) ZH (Simplified Chinese).

CTG8940W   ctgmaster -rrmname switch value (*value* **) is not valid and has been ignored**

Explanation: The ctgmaster program was invoked with the -rrmname switch, but *value* is not a valid value.

System action: Execution of ctgmaster continues, using the value defined in the CTG_MASTER_RRMNAME environment variable. If the environment variable is undefined or not valid, the default master RRMNAME of CICSTG.DEFAULT.MASTERRRM is used instead.

User response: Check that the -rrmname switch specifies a valid string that is no longer than 32 characters. See the Administration Guide for more details.

CTG8945W   Environment variable CTG_MASTER_TRACE_ON value (*value1* **) is not valid. Using default value=***value2*

Explanation: Environment variable CTG_MASTER_TRACE_ON is set to an incorrect value *value1* .

System action: Execution of ctgmaster continues, using the default value *value2* instead.

User response: Check that the CTG_MASTER_TRACE_ON environment variable is set to either YES or NO.

CTG8946W   Environment variable CTG_MASTER_RRMNAME is undefined. Using default value=*value*

Explanation: Environment variable CTG_MASTER_RRMNAME is not defined. This environment variable defines the RRMNAME that is to be used in this TCP/IP load balancing group.

System action: Execution of ctgmaster continues, using the default value *value* .

User response: Check that the CTG_MASTER_RRMNAME environment variable is defined. The RRMNAME must be no longer than 32 characters. See the Adminstration guide for more information.

CTG8947W   Environment variable CTG_MASTER_INPUT value (*value1* **) is not valid. Using default value=***value2*

Explanation: Environment variable CTG_MASTER_INPUT is set to an incorrect value *value1* .

System action: Execution of ctgmaster continues, using the default value *value2* .

User response: Set the CTG_MASTER_INPUT environment variable to STDIN if running from the USS shell, or CONSOLE if running in batch mode.

CTG8948W Environment variable CTG_XA_MAX_TRAN value (*value1***) is not valid. Using default value=***value2*

Explanation: Environment variable CTG_XA_MAX_TRAN is set to an incorrect value *value1* .

System action: Execution of ctgmaster continues, using the default value *value2* .

User response: Set the CTG_XA_MAX_TRAN environment variable to an integer between 1 and 8192. As a guide, take the sum of the MAXCONNECT values of each Gateway daemon in the IP load-balancing group.

CTG8949W Environment variable CTG_MASTER_NATLANG value (*value1***) is not valid. Using default value=***value2*

Explanation: Environment variable CTG_MASTER_NATLANG is set to an incorrect value *value1* .

System action: Execution of ctgmaster continues, using the default value *value2* .

User response: Set the CTG_MASTER_NATLANG environment variable to one of the following two-character strings: EN (English) JA (Japanese) ZH (Simplified Chinese).

CTG8952E ctgmaster initialization has failed to ENQ on resource name (*value1***) with rc=***value2*

Explanation: The ctgmaster program could not obtain an exclusive ENQ on resource name (*value1*). This is required to ensure the integrity of ctgmaster resources.

System action: Execution of ctgmaster is ended.

User response: Check that the resource name *value1* is valid, and that no other address space is holding an ENQ on the same resource name. The ctgmaster ENQ name should be of the form SYSZCTG.<CTG_MASTER_RRMNAME>, where <CTG_MASTER_RRMNAME> is the value of the CTG_MASTER_RRMNAME environment variable. If problems persist, collect ctgmaster trace output and contact your service organization.

CTG8953E ctgmaster initialization has failed to initialize RRMS PC services with rc=*value1*

Explanation: The ctgmaster program could not

initialize the RRMS PC services, which are provided by the CTGRRMS service address space.

System action: Execution of ctgmaster is ended.

User response: Ensure that the user ID starting the ctgmaster program has at least UPDATE access to the RACF profile CTG.RRMS.SERVICE. If problems persist, collect ctgmaster trace output and contact your service organization.

CTG8954E ctgmaster initialization has failed to register with RRMS with rc=*value1*

Explanation: The ctgmaster program could not register with RRMS.

System action: Execution of ctgmaster is ended.

User response: Ensure that the CTG_MASTER_RRMNAME is valid and is not already being used by another process. The name should be no longer than 32 characters, and should not end with .UA; see the Adminstration Guide for full details. If problems persist, collect ctgmaster trace output and contact your service organization.

CTG8955E ctgmaster initialization has failed to start the console listener thread with rc=*value1*

Explanation: The ctgmaster program could not start the console listener thread, which is needed for user interaction.

System action: Execution of ctgmaster is ended.

User response: Ensure that ctgmaster has been started with the correct input type for the runtime environment. The input type is determine by either the CTG_MASTER_INPUT environment variable, or the command line -input switch. When ctgmaster is running in batch mode (via CTGBATCH, or BPXBATCH from JCL), the input type must be set to CONSOLE. When ctgmaster is running in the USS shell, the input type must be set to STDIN.

CTG8984E Duplicate XID found when adding entry to XID Mapping table

Explanation: Whilst attempting to start an XA transaction, the given XID was found to be a duplicate of the XID of an existing transaction.

System action: The transaction branch is not started.

User response: Resolve the existing transaction in RRMS or contact your service organization.

CTG8985E Maximum XA transaction limit is exceeded.

Explanation: The CICS Transaction Gateway could not complete the transaction because too many XA transactions are active.

System action: The transaction is rolled back.

User response: Stop the Gateway daemon or ctgmaster process. Increase the value of the CTG_XA_MAX_TRAN environment variable, and then restart.

CTG8986E Unable to delete shared storage. xmt_rc = *reason code*

Explanation: The CTGMaster process could not delete the shared memory that it owns.

System action: CTGMaster continues to shut down, and the shared memory remains.

User response: Contact your service organization.

CTG8987E Unable to determine number of Gateway Daemons attached to CTGMaster. xmt_rc = *reason code*

Explanation: Details of which processes are attached to shared memory cannot be found by CTGMaster.

System action: CTGMaster will shutdown regardless, the memory will be removed such that existing users still have access until they terminate. New connections will not be possible until a new CTGMaster is established.

User response: Contact your service organisation.

CTG8988E Unable to determine which directory CTGMaster is running in

Explanation: CTGMaster needs to determine where the install directory is so that it can get to necessary data files.

System action: CTGMaster will be unable to start.

User response: If running under USS, add the <CICS TG install>/bin directory to the system path, otherwise launch ctgmaster with an absolute pathname i.e. '/usr/lpp/cicstg/bin/ctgmaster'.

CTG8990E ctgmaster userid is not authorized to access the CTGRRMS PC services

Explanation: The userid under which the ctgmaster program is running does not have sufficient security priviledges to access the CTGRRMS service address space.

System action: The ctgmaster program will terminate.

User response: In order to successfully initialize, the userid under which the ctgmaster program is executing must have at least UPDATE access to the profile 'CTG.RRMS.SERVICE'.

CTG8991E ctgmaster version different from the active CTGRRMS PC services.

Explanation: The version of ctgmaster being run is different from the version of the CTGRRMS service address space which is running.

System action: The ctgmaster program terminates.

User response: Ensure that the versions of ctgmaster and the CTGRRMS services are the same. Refer to the z/OS Administration Guide for details on how to refresh the CTGRRMS Services.

CTG8995E RRM name of the master process has more than 32 characters.

Explanation: The RRM name for the ctgmaster process, as defined by the CTG_MASTER_RRMNAME environment variable, or the -rrmname switch of the ctgmaster command, is too long. The message is issued when an attempt is made to start either a ctgmaster process, or a Gateway daemon that is part of a TCP/IP port sharing group.

System action: This message is output and the ctgmaster process or Gateway daemon fails to start.

User response: Specify a name less than or equal to 32 characters.

CTG8997E stdin console unavailable, ctgmaster will terminate.

Explanation: ctgmaster was started without a stdin console. This is likely to be caused by running ctgmaster under CTGBATCH without using the override ″-input=console″.

System action: ctgmaster will terminate

User response: Add the ″-input=console″ override to the ctgmaster command in the PARM for CTGBATCH, or set the environment variable ″CTG_MASTER_INPUT=CONSOLE″.

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.

CCL9119E   Attempt to get addressability to an IPC queue element failed with rc = *value1*

Explanation: On a Windows platform an attempt to get addressability to a shared memory element has failed for some operating system reason.

System action: The message is written to the error log and the Client tries to continue, although attempts to reference this shared memory element will result in a trap.

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

CCL9124E   Unable to determine maximum file descriptors, sysconf rc=error**, using default**

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 sysconf system call.

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

CTG9200E   XA transactional calls have been issued out of sequence.

Explanation: The transaction manager has issued an XA call on a transaction, but the resource adapter cannot comply, because the transaction is not in a state that allows this request.

System action: The XA call is not actioned. The resource adapter returns a protocol error to the transaction manager. This message is logged, with the relevant XA and RRMS transaction identifier information.

User response: If the transaction manager cannot resolve the transaction, use the RRMS or transaction manager administration tools to resolve the transaction manually.

CTG9201E   Initialization of CICS TG internal storage failed (return code *return code* **).**

Explanation: An error prevents the master process or Gateway daemon from storing details of XA transactions

System action: The Gateway daemon or master process fails to start.

User response: Take action according to the return code. For return code 1007, increase the amount of memory available to the Gateway daemon process, or reduce the value for CTG_XA_MAX_TRAN. For return code 1014, reduce the number of shared memory segments running on the system. See the Unix System Services documentation for details on the of interprocess communication limits. For return code 1027 ensure that the CICS TG <install_path>/bin directory is on the path, and. Ensure that the ctgmaster executable file is in the <install_path>/bin directory.

CTG9202E   An RRS application is already using the same resource manager name (name **).**

Explanation: An attempt is made to start a ctgmaster process or Gateway daemon, but another application is already using the resource manager name supplied. The application could be another instance of the Gateway daemon or a third party product.

System action: The error is logged and the ctgmaster process or Gateway daemon fails to start.

User response: Correct the configuration and try again. See the Adminstration Guide for information on resource manager names and when to invoke a CICS TG master process.

CTG9203E   A ctgmaster process is already using resource manager name (*name***).**

Explanation: A attempt is made to start a ctgmaster process or Gateway daemon, but an instance of ctgmaster is already using the resource manager name supplied.

System action: The error is logged and the ctgmaster or Gateway daemon fails to start.

User response: Correct the configuration and try again. See the Adminstration Guide for information on resource manager names and when to invoke a CICS TG master process.

CTG9204E   CICS Transaction Gateway daemon is unable to communicate with the CTGMASTER. xmt_rc=*return code*** .**

Explanation: An attempt is made to start a Gateway daemon as part of a TCP/IP load balancing group. A ctgmaster is registered with resource manager name (CTG_MASTER_RRMNAME). However, the Gateway daemon cannot access the shared storage that is needed to share information about XA transactions. This could be because the ctgmaster and Gateway daemon have been started using different HFS installations, or the ctgmaster is in the process of shutting down.

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

User response: Ensure that the ctgmaster and Gateway daemon processes were started using the same HFS installation and the the CTG_MASTER_RRMNAME specified by the Gateway daemon points to an active ctgmaster process and not a ctgmaster in the process of shutting down.

CTG9208E   Error in GetMethodID function.

Explanation: The call to GetMethodID has failed. This is an internal error.

System action: CICS Transaction Gateway processing continues.

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

CTG9209E   Error in NewByteArray function.

Explanation: The call to NewByteArray has failed. This is an internal error.

System action: CICS Transaction Gateway processing continues.

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

CTG9212E   The Gateway daemon userid is not authorized to access the CTGRRMS PC services.

Explanation: The userid under which the Gateway daemon is running does not have sufficient security priviledges to access the CTGRRMS service address space.

System action: The Gateway daemon will terminate.

User response: In order to successfully initialize with xasupport=ON, the userid under which the Gateway daemon is executing must have at least UPDATE access to the profile 'CTG.RRMS.SERVICE'.

CTG9213W   CTG_XA_MAX_TRAN value ignored within a TCP/IP load balancing environment.

Explanation: When TCP/IP load balancing for XA transaction support is enabled, the maximum number of concurrent XA transactions is controlled by the value of environment variable CTG_XA_MAX_TRAN specified in the CTGMASTER address space. CTG_XA_MAX_TRAN is required for a standalone Gateway daemon configured for XA transactional support.

System action: The value of CTG_XA_MAX_TRAN specified for this Gateway daemon is ignored. The value of CTG_XA_MAX_TRAN specified in the CTGMASTER address space is used instead.

User response: Correct the configuration by unsetting CTG_XA_MAX_TRAN for this Gateway daemon.

CTG9215E   CICS TG version different from the active CTGRRMS PC services.

Explanation: The version of the CICS Transaction Gateway being run is different from the version of the CTGRRMS service address space which is running.

System action: The CICS Transaction Gateway terminates.

User response: Ensure that the versions of the CICS Transaction Gateway and the CTGRRMS services are at the same version. Refer to the z/OS Administration Guide for details on how to refresh the CTGRRMS Services.

CTG9217W   Environment variable CTG_XA_MAX_TRAN value (*value1***) is not valid. Using default value=***value2*** .**

Explanation: Environment variable CTG_XA_MAX_TRAN is set to an incorrect value *value1* .

System action: Execution of the gateway daemon continues, using the default value *value2* .

User response: Set the CTG_XA_MAX_TRAN environment variable to an integer between 1 and 8192.

CTG9219E **An Gateway daemon process is already using resource manager name** *name* **.**

Explanation: An attempt was made to start a ctgmaster process or Gateway daemon, but a Gateway daemon is already using the resource manager name supplied.

System action: The error is logged and the ctgmaster process or Gateway daemon fails to start.

User response: Correct the configuration and try again. See the information center for information on resource manager names and when to invoke a CICS TG master process.

CTG9224E **The Gateway daemon cannot verify that the registered process using resource manager name** *name* **is a ctgmaster process.**

Explanation: The Gateway daemon is attempting to start as part of a Gateway group but cannot verify that the owner of the RRMS registration is a ctgmaster process.

System action: The error is logged and the Gateway daemon fails to start.

User response: Contact your service organisation.

CTG9274E **Memory allocation failure.**

Explanation: The CICS Transaction Gateway application is running in 64-bit mode. The CICS TG attempted to call an internal 31-bit API, but was unable to allocate sufficient storage within the 31-bit address range to pass data to the API.

System action: This message is logged and CICS Transaction Gateway processing continues. If the memory allocation failure prevented an EXCI request from completing, Cics_Rc is set to ECI_ERR_RESOURCE_SHORTAGE.

User response: Configure the system to give give the process invoking the CICS TG classes more storage below the 2 gigabyte line. Consult the z/OS documentation for details on how to do this.

CTG9289E **Memory allocation failure.**

Explanation: This message indicates a malloc() failure when attempting to allocate memory. The memory was required for the call that determines the jobname associated with the Gateway daemon.

System action: CICS Transaction Gateway processing continues. Messages written to the system log will contain spaces in place of the jobname.

User response: Refer to the z/OS documentation to determine what corrective action is necessary to provide adequate memory to the CICS Transaction Gateway process.

CTG9290E **The Unix System Services call BPX1GTH failed with return code** *return code* **and reason code** *reason code* **.**

Explanation: This message indicates a Unix System Services call 'BPX1GTH' failed. The Gateway daemon uses this call to determine its jobname.

System action: CICS Transaction Gateway processing continues. Messages written to the system log will contain spaces in place of the jobname.

User response: Refer to the z/OS Assembler Services Reference to determine what the return code and reason code values signify.

CTG9291E **Cannot invoke initialization for CTGRRMS services.**

Explanation: To support XA based requests, the CICS Transaction Gateway issues authorized RRS calls, through CTGRRMS services. It was not possible to invoke the initialization of these services.

System action: This message is logged. The CICS Transaction Gateway does not start.

User response: If XA support is required, see the information center for information on how to start CTGRRMS services. If XA support is not required, set the xasupport configuration parameter off and restart the CICS Transaction Gateway.

CTG9299E **The Gateway daemon does not have sufficient authority to write to SMF**

Explanation: The Gateway daemon attempted to write an SMF record. This failed because the user ID does not have sufficient authority. The user ID that the Gateway daemon runs under is required to have read access to the BPX.SMF facility class.For further details, see the RACF and Unix System Services documentation

System action: This message is logged. All data in the current SMF record is lost.

User response: If writing to SMF is required, update z/OS RACF with the pre-requisite permissions and restart the Gateway daemon.

CTG9300E   Writing a record to SMF failed with return code (Errno=*errno1*, Errno2=*errno2*)

Explanation: The Gateway daemon issued a call to the Unix System Services API call __smf_record. This failed with the following return codes: *errno1* and *errno2*

System action: This message is logged. All data in the current SMF record is lost.

User response: Verify with the z/OS systems administrator that the SMF subsystem is working correctly.See the CICS Transaction Gateway Information Center for further information onCICS Transaction Gateway recording to SMF.For further details on the error codes see the C/C++ Run-Time Library Reference, and refer to the section describing the __smf_record() call. If the problem persists contact your service organisation.

CTG9301E   SMF is not currently accepting records

Explanation: The Gateway daemon attempted to write a record to SMF. The Unix System Services call __smf_record failed with reason code JRSMFNotAccepting, indicating that SMF is not currently accepting records.

System action: This message is logged. All data in the current SMF record is lost.

User response: Contact the systems administrator responsible for SMF. For further details on the error code see the UNIX System Services Messages and Codes book, and refer to the section on reason codes.

CTG9302E   The Gateway daemon JNI failed to allocate storage to process an SMF record.

Explanation: A call to the UNIX System Services API malloc failed, which indicates that it did not have enough memory available.

System action: This message is logged. All data in the SMF record is lost.

User response: See the CICS Transaction Gateway Information Centre for details on dealing with out of memory errors.

CCL9303E   Memory allocation failed. 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. 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.

CCL9344E   Error during CICS_EciListSystems call. Return code: *rc*

Explanation: An error was encountered during execution of the code. 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 CICS Client/Server Programming book 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: Reinstall your Java Virtual Machine

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

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 creat 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

CTG9505E   Cannot find class com.ibm.ctg.server.JGate

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.

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

CTG9528E   Unable to install *service name*, **error:** *error message text*

Explanation: Installation of the CICS TG service failed. An error code describing why failure occurred is supplied.

System action: The command line application terminates

User response: Check that the service isn't already installed and try again.

CTG9532E   Service *service name* **failed to stop**

Explanation: When trying to stop the CICS TG service before uninstallation an error occurred.

System action: The command line application terminates.

User response: Try to stop the service manually and try the uninstall again.

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.

CTG9534E   DeleteService failed: *error message text*

Explanation: During an uninstall of the CICS TG service the service could not be deleted.

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.

CTG9538E   Failed to create service: *error message text*

Explanation: During an install of the CICS TG service the service could not be created.

System action: The command line application terminates.

User response: Check to see if the service is already installed.

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.

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.

CTG9552E   Service *service name* **failed to start. Parameter list** *values* **.**

Explanation: A problem occurred when the named service was being started. *values* are the parameters that were defined for the service.

System action: The service failed to start. This message is written to the application event log of the Windows operating system.

User response: Inspect the application event log for other messages that help to explain the problem. Address the cause of failure and retry the command.

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 application event log of the Windows operating system.

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 application event log of the Windows operating system.

User response: Inspect the 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 application event log of the Windows operating system.

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.

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.

CTG9603E   ConnectionManager has returned an invalid connection

Explanation: The connection returned by the ConnectionManager 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.

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.

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 ECIConnection is no longer allowed as it has been closed. A new ECIConnection 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. 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: System action is not required.

User response: Connection handle supplied is null.

CTG9619E   Cannot associate connection with ECIManagedConnection. 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 ConnectionManager not of type javax.resource.cci.Connection**

Explanation: The ConnectionManager 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: The ConnectionRequestInfo handle is not of the correct type.

CTG9623E **ConnectionManager supplied is null**

Explanation: The ConnectionManager 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 ConnectionManager to the ECIConnectionFactory constructor.

System action: System action is not required.

User response: The ConnectionManager 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 ECIInteraction not of type ECIConnection**

Explanation: The ECIConnection that is being used by your ECIInteraction 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 ECIInteractionSpec**

Explanation: Only InteractionSpec objects of the type ECIInteractionSpec can passed to the ECIInteraction.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 ECIInteraction.execute() a non null ECIInteractionSpec must be supplied at all times.

System action: System action is not required.

User response: Null was passed to the ECIInteractionSpec.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. Error Code=

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 EPIConnection 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 ConnectionManager

Explanation: The ConnectionManager did not return an EPIConnection for use by the application.

System action: System action is not required.

User response: No connection returned by the ConnectionManager.

CTG9641E   Invalid connection was returned by ConnectionManager

Explanation: An unusable Connection object was returned by the ConnectionManager.

System action: System action is not required.

User response: Invalid connection was returned by ConnectionManager.

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.

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   No Connection Manager passed to resource adapter. Adapter unable to function

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.

CTG9663E   A connection which is not the active connection attempted to run on the resource adapter

Explanation: The EPIConnection 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. 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 ECIInteractionSpec 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 ECIInteractionSpec.

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 ECIInteractionSpec. 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 ECIInteractionSpec 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 CTG daemon.

CTG9678E EPI resource adapter failed trying to read streamed EPI data. See Linked exception for details.

Explanation: During a read of the terminal, an exception has occurred. The original exception has been linked.

System action: The terminal read could not continue.

User response: Retrieve the linked exception from the CICSUserInputException and diagnose from the original message.

CTG9679E EPI resource adapter failed trying to write streamed EPI data. See Linked exception for details.

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 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.

CTG9682E   Mapped Input Record is not an ECIChannelRecord.

Explanation: The Input 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 ECIChannelRecord.

System action: The request is rejected.

User response: Check the type of the input record.

CTG9683E   Mapped Output Record is not an ECIChannelRecord.

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 ECIChannelRecord.

System action: The request is rejected.

User response: Check the type of the input record.

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   Attempted to Rollback due to error setting *requested value* **. Error occurred during rollback because of setting** *associated value*

Explanation: When trying to make multiple changes to the Gateway daemon *requested value* failed. Then when trying to undo the changes already made, a further change *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 unwriteable 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 unwriteable 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.

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:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, 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

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

AIX
AnyNet
AS/400
CICS
CICS/400
CICS/ESA
CICS/VSE
DB2
Domino
Hummingbird
IBM
IBM
IBMLink
IMS
iSeries
MQSeries
MVS
MVS/ESA
Notes
OS/2
OS/390
POWER
pSeries
RACF
Redbooks
RETAIN
RMF
RS/6000
SAA
SP2
System/390
Tivoli
TXSeries
VisualAge
VSE/ESA

VTAM
WebSphere
z/OS
zSeries

Microsoft®, Windows®, Windows NT®, 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 Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX® is a registered trademark of The Open Group in the United States and other
countries.

Intel®, Intel Inside® (logos), MMX and Pentium® are trademarks of Intel Corporation
in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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

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.

IBM

Program Number: 5724-I81, 5655-R25 and 5724-J09

Spine information:

CICS Transaction Gateway

CICS Transaction Gateway Messages

Version 7.1

IBM