

CICS[®] Transaction Gateway



Messages

Version 3.1

CICS[®] Transaction Gateway



Messages

Version 3.1

Note!

Before using this information and the product it supports, be sure to read the general information under "Appendix. Notices" on page 15.

First edition (September 1999)

This edition applies to Version 3.1 of IBM CICS Transaction Gateway program number 5648-B43.

© **Copyright International Business Machines Corporation 1994, 1999. All rights reserved.**

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

Contents

About this book	v
Conventions and terminology used in this book	v
Prerequisite and related information	v
How to send your comments	v
Obtaining books from IBM	vi
 Error messages	 1
 Appendix. Notices.	 15
Trademarks	16

About this book

This book lists the error messages for CICS Transaction Gateway Version 3.1.

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

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

How to send your comments

Your feedback is important in helping to provide the most accurate and high-quality information. If you have any comments about this book, or any other CICS documentation:

- Visit our Web site at:

<http://www.ibm.com/software/ts/cics/>

and follow the **library** link to our feedback form.

Here you will find the feedback page where you can enter and submit your comments.

- Send your comments by e-mail to idrctf@hursley.ibm.com
- Fax your comments to:

+44-1962-870229 (if you are outside the UK)
01962-870229 (if you are in the UK)

- Mail your comments to:

Information Development
Mail Point 095
IBM United Kingdom Laboratories
Hursley Park
Winchester
Hampshire
SO21 2JN
United Kingdom

Whichever method you use, ensure that you include:

- The name of the book
- The form number of the book
- If applicable, the version of the product
- The specific location of the text you are commenting on, for example, a page number or table number.

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Obtaining books from IBM

For information on books you can download, visit our Web site at:

<http://www.ibm.com/software/ts/cics/>

and follow the **library** link.

You can order hardcopy books:

- Through your IBM representative or the IBM branch office serving your locality.
- From the Web site at:

<http://www.elink.ibm.link.ibm.com/pb1/pb1>

Error messages

CCL65XXI Unable to find message. [{0}]

Explanation: The message appropriate for this error code cannot be found.

System Action: Display the number of the message.

User Response: Find the appropriate message by searching for its number in this message file.

CCL6500I Starting the CICS Transaction Gateway with default values.

Explanation: The Gateway is starting using either the default configuration settings, or those set with the configuration tool. Default settings are used when none have been set with the configuration tool.

System Action: A Gateway console session is started.

User Response: For help on the startup options, enter *JGate?*

CCL6501I Starting the CICS Transaction Gateway with user specified values.

Explanation: The Gateway is starting using options specified from the command prompt.

System Action: Options specified on the command line will override those specified using the configuration tool.

User Response: For help on the startup options, enter *JGate?*

CCL6502I [Initial ConnectionManagers = {0}, Maximum ConnectionManagers = {1},

Explanation: {0} ConnectionManager threads will be created initially. A maximum of {1} ConnectionManager threads will be allowed. One ConnectionManager thread is needed for each connected client program.

System Action: The Gateway will start using these values for the pool of ConnectionManager threads.

User Response: Not required.

CCL6502I Initial Workers=value1, Maximum Workers=value2, tcp: Port= value3.

Explanation: {0} Worker threads will be created initially. A maximum of {1} Worker threads will be allowed. One Worker thread is needed for each in progress request. The tcp: protocol handler will accept connections on TCP/IP port {2}

System Action: The Gateway will start using these values for the pool of Worker threads.

User Response: Not required.

CCL6503I Starting with tracing messages on.

Explanation: Extra tracing messages have been enabled.

System Action: Information passing between the Gateway and the browser will be recorded.

User Response: For help on the startup options, enter *JGate?*

CCL6504I Starting with debugging messages on.

Explanation: Debugging messages have been enabled.

System Action: Debugging information passing between the Gateway and Java™ application will be recorded.

User Response: Not required.

CCL6505I Successfully created the initial ConnectionManager and Worker threads.

Explanation: ConnectionManager and Worker threads handle network connections and requests from or replies to Java clients.

System Action: ConnectionManager will handle network connections from a particular Java client, the Worker thread will execute requests from it.

User Response: Not required.

CCL6506I Client connected. [{0}] - {1}

Explanation: A client program connected. The connection is being handled by ConnectionManager {0}. The connection was made with protocol characteristics {1}.

System Action: If an existing client program is in execute mode, put the connected client program in a queue. If there is no client program in execute mode, execute the connected client program.

User Response: Not required.

CCL6507I Client disconnected. [{0}] - {1}

Explanation: A client program has disconnected. The connection was being handled by ConnectionManager {0}. The connection was made with protocol characteristics {1}.

System Action: If there is still work in progress, the ConnectionManager waits for the length of the closetimeout period for in-progress requests to complete.

Error messages

User Response: Not required.

CCL6508I Type Q or - to stop the CICS Transaction Gateway.

Explanation: Typing either character, followed by the *Enter* key, will stop the Gateway.

System Action: Not required.

User Response: Enter option to stop gateway.

CCL6509I The CICS Transaction Gateway is stopping.

Explanation: The Gateway is executing a controlled stop.

System Action: Release threads and ports.

User Response: Not required.

CCL6510I TCP/IP hostnames will not be shown.

Explanation: The -nonames command line option was specified. This is useful in environments where a TCP/IP DNS is not available.

System Action: Any displayed TCP/IP addresses will not be resolved to TCP/IP hostnames.

User Response: Not required.

CCL6511I ThreadManager created {0} {1}(s).

Explanation: The ThreadManager created {0} thread(s) of type {1}

System Action: Create {0} threads of type {1}

User Response: Use the configuration tool to specify the number of threads that can be created.

CCL6512I Waiting to receive a request. [{0}]

Explanation: ConnectionManager {0} is waiting for an inbound request.

System Action: When a request is received, ConnectionManager will allocate a worker thread.

User Response: Not required.

CCL6513I About to receive remainder of a request. [{0}]

Explanation: A header has been received for a request of type {0}.

System Action: Retrieve information on request to execute the remainder of request.

User Response: Not required.

CCL6514I Accepting work request. [{0}] [{1}]

Explanation: Worker thread {0} has accepted a request from ConnectionManager {1}.

System Action: Execute CICS request.

User Response: Not required.

CCL6515I Finished work request. [{0}]

Explanation: Worker thread {0} has successfully completed a request.

System Action: Send CICS reply to client.

User Response: Check CICS reply from Gateway.

CCL6516W Outstanding work in progress. [{0}] [{1}]

Explanation: A client has disconnected from ConnectionManager {0}, but {1} 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.

CCL6517I All outstanding work has been finished. [{0}]

Explanation: Where {0} specifies the ConnectionManager.

System Action: Wait for next client program connection or stop gateway.

User Response: Either run a new client program or quit the gateway.

CCL6518W Unfinished extended requests were in progress. [{0}] [{1}]

Explanation: A client has disconnected from ConnectionManager {0}, but {1} extended requests were unfinished.

System Action: A ConnectionManager will not become available to handle another connection until the extended requests have been cleaned up.

User Response: Not required.

CCL6519I Attempts have been made to clean up all unfinished extended requests.

Explanation: Gateway is cleaning up all unfinished extended requests.

System Action: Cleaning up all unfinished extended requests.

User Response: Not required.

CCL6520I This is the first request of an extended request sequence. [{0}]

Explanation: Where {0} specifies the ConnectionManager thread.

System Action: Allocate a worker thread to the request.

User Response: Not required.

CCL6521I This request is part of an extended request sequence. [{0}]

Explanation: Where {0} specifies the ConnectionManager thread.

System Action: Allocate worker thread to the request.

User Response: Not required.

CCL6522I This is the final request of an extended request sequence. [{0}]

Explanation: Where {0} specifies the ConnectionManager thread.

System Action: Allocate worker thread to the request.

User Response: Not required.

CCL6523I About to execute work request. [{0}]

Explanation: Where {0} specifies the Worker thread.

System Action: Executing request.

User Response: Not required.

CCL6524I Successfully started handler for the {0}: protocol.

Explanation: Where {0} specifies the protocol type.

System Action: Gateway started with {0} protocol handler.

User Response: Not required.

CCL6525W Unable to start handler for the {0}: protocol. [{1}]

Explanation: Where {0} specifies the protocol type, and {1} 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: Gateway forced to stop because of {0} protocol handler error.

User Response: Check if protocol or port is available and supported by running gateway.

CCL6526I Parameters passed to {0}: protocol handler = {1}.

Explanation: The parameters, {1}, defined in the Gateway.properties file have been passed to the {0} protocol handler.

System Action: Gateway started with {0} protocol handler = {1}.

User Response: Not required.

CCL6527I Using {0} class to provide security to {1}.

Explanation: The {0} ServerSecurity class is being used on connection {1}.

System Action: Gateway is using secured connection.

User Response: Not required.

CCL6528W Outstanding work still in progress. [{0}] [{1}]

Explanation: A client has disconnected from ConnectionManager {0}, with work requests in progress. {1} requests have failed to finish within a reasonable time.

System Action: The ConnectionManager marks itself as available again.

User Response: Not required.

CCL6529W Unable to send reply to client. [{0}]

Explanation: Where {0} specifies the worker thread. This could be because the client has already disconnected.

System Action: Not required.

User Response: Check if client is still running.

CCL6530I Disconnecting idle client. [{0}]-{1}

Explanation: ConnectionManager disconnected the client at {1}, which had been idle for longer than the idletimeout value specified in the configuration file.

System Action: System action is not required.

User Response: Not required.

CCL6531W Unable to disconnect idle client - work still in progress. [{0}]-{1}

Explanation: The client at {1} 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: Not required.

Error messages

CCL6532I Disconnecting non-responsive client. [{0}]-{1}

Explanation: The client at {1} did not respond to a PING before the next was due to be sent.

System Action: The client will be disconnected.

User Response: Not required.

CCL6533W Unable to read the ctg.ini file. [{0}]

Explanation: Where {0} specifies the reason for the failure.

System Action: Not required.

User Response: Check if the ctg.ini file is available in the ctg/bin directory.

CCL6534I Using a worker timeout of {0} milliseconds.

Explanation: The worker thread timeout is set to {0} ms.

System Action: Not required.

User Response: Not required.

CCL6535I Using a close timeout of {0} milliseconds.

Explanation: The closedown timeout is set to {0} ms.

System Action: Not required.

User Response: Not required.

CCL6536I Generic ECI reply calls will not be allowed.

Explanation: Generic ECI reply calls will not be allowed.

System Action: Not required.

User Response: Not required.

CCL6537I Security classes will be required for all connections on all protocols.

Explanation: Security is set on all protocols for all connections.

System Action: Not required.

User Response: Not required.

CCL6550E Unable to listen on requested port.

Explanation: Gateway is unable to recognise port number.

System Action: Not required.

User Response: Check valid port number, protocol and server name.

CCL6551E Unable to create requested ConnectionManager and worker threads.

Explanation: Usually this means that the system is running low on resources.

System Action: Not required.

User Response: Reduce the number of ConnectionManager and worker threads initially created.

CCL6552E Error whilst accepting a connection from {0}. Connection closed. [{1}]

Explanation: A connection was accepted from {0}, but a problem of type {1} occurred.

System Action: Unable to connect.

User Response: Check error code.

CCL6553E Error reading request. [{0}]

Explanation: Where {0} specifies the ConnectionManager thread.

System Action: Unable to execute request.

User Response: Check if connection is still available.

CCL6554E Error in native method. [{0}]

Explanation: Where {0} specifies the Worker thread.

System Action: Not required.

User Response: Not required.

CCL6555E Error writing reply [{0}]

Explanation: Where {0} specifies the Worker thread.

System Action: Unable to write reply.

User Response: Check if connection is still available.

CCL6556E Error copying request on local Gateway.

Explanation: A problem has occurred in the local Java Gateway support.

System Action: Unable to copy request on local Gateway.

User Response: Check if Gateway is configured to use local protocol.

CCL6557E: Error whilst executing request. [{0}]

Explanation: An error occurred while processing request with name [{0}]

System Action: Not required.

User Response: Check that request is valid.

CCL6558E 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 Gateway.properties for protocol handlers; uncomment the protocol handle you want.

CCL6559E Unexpected exception occurred [{0}]

Explanation: An exception of type [{0}] occurred.

System Action: Not required.

User Response: Not required.

CCL6560E Unable to accept request of type {0}. [{1}]

Explanation: ConnectionManager {1} has received a request {0} 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 {0} class is in the CLASSPATH of the Gateway process.

CCL6561E Unable to use {0} class to provide security to {1}.

Explanation: The {0} class could not be successfully loaded to provide security to the connection from {1}.

System Action: The connection will be closed.

User Response: Make sure that the {0} class is in the CLASSPATH of the Gateway process.

CCL6562E Connection to {0} rejected due to insufficient ConnectionManagers.

Explanation: A connection was accepted from {0}, but a ConnectionManager did not become available in time.

System Action: The connection was rejected.

User Response: Increase the maximum number of ConnectionManagers.

CCL6563E {0} protocol handler exited unexpectedly. [{1}]

Explanation: The {0} protocol handler exited due to problem {1}.

System Action: Gateway exits and displays error message.

User Response: Check error code.

CCL6569E: Unable to open file {0}

Explanation: File with name {0} could not be opened.

System Action: Not required.

User Response: Check that file is not in use or read-only.

CCL6570E: Processing file {0}

Explanation: Processing file with name {0}.

System Action: Not required.

User Response: Not required.

CCL6571E: Creating file {0}

Explanation: File with name {0} is being created.

System Action: Not required.

User Response: User response not required.

CCL6572E: Renaming file {0} to {1}

Explanation: File name {0} is being renamed to {1}

System Action: Not required.

User Response: User response not required.

CCL6573E: Processing complete

Explanation: System has finished processing.

System Action: Not required.

User Response: User response not required.

CCL66XXI Unable to find message. {0}

Explanation: This message is displayed when the appropriate message could not be found for your locale. {0} contains the message number that should have been displayed.

System Action: Display the number of the message.

User Response: Find the appropriate message by searching for its number in this message file.

Error messages

CCL6601I Request specific trace {0}.

Explanation: A historical message.

System Action: Not required.

User Response: Not required.

CCL6602I GatewayRequest type = {0}, flow version = {1}, flow type = {2}, Gateway return code = {3}, length of data following the header = {4}.

Explanation: Information from current request.

System Action: Execute request.

User Response: Not required.

CCL6650E Unable to connect to the Gateway.

Explanation: Client application unable to connect to gateway.

System Action: Client request unable to start.

User Response: Check if Gateway is running; if client application has specified the correct parameters to connect to the gateway.

CCL6651E Unable to connect to the Gateway [address={0}, port={1}] [{2}]

Explanation: Where {0} specifies the Gateway TCP/IP address, {1} specifies the TCP/IP port and {2} specifies the cause of the error.

System Action: Client request unable to start.

User Response: Check error code.

CCL6652E Unable to start the Gateway listener.

Explanation: Client application could not start the Gateway listener.

System Action: Close this JavaGateway instance.

User Response: Check that Gateway is correctly configured.

CCL6653E Unable to flow request to the Gateway; this JavaGateway instance has been closed.

Explanation: The Gateway is unreachable.

System Action: Close this JavaGateway instance.

User Response: Check if the connections to the Gateway still exist.

CCL6654E An error occurred while the Gateway was closing. [{0}]

Explanation: Where {0} specifies the cause of the error.

System Action: Close the Gateway.

User Response: Check error code.

CCL6655E There was an error reading the reply. [{0}]

Explanation: Where {0} specifies the cause of the error.

System Action: Unable to read the reply.

User Response: Check error code.

CCL6656E Flow did not contain the correct eyecatcher. Fatal Error.

Explanation: Network connection did not contain the correct identifier.

System Action: Connection closed.

User Response: Not required.

CCL6657E Invalid CICS Transaction Gateway address specified.

Explanation: The Gateway URL was incorrectly specified.

System Action: Unable to connect to Gateway at this address.

User Response: Check Gateway is running and address is right.

CCL6658E Local Gateway support has been terminated.

Explanation: Attempts have been made to use the local Gateway support after the LocalJavaGateway.terminate method has been called.

System Action: Unable to make further requests to this gateway.

User Response: Not required.

CCL6659E In use Local Gateways currently exist.

Explanation: The LocalJavaGateway.terminate method has been called while local Gateways are still in use.

System Action: Wait for requests to finish.

User Response: Not required.

CCL6660E Error copying reply. [{0}]

Explanation: Where {0} specifies the cause of the error.

System Action: Unable to copy reply.

User Response: Check error code.

CCL6661E 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: Not required.

User Response: Close JavaGateway instance.

CCL6662E This JavaGateway instance is still open.

Explanation: This JavaGateway instance is still open in the client application.

System Action: Not required.

User Response: Close this JavaGateway instance.

CCL6663E Cannot open a JavaGateway instance when no protocol has been specified.

Explanation: No protocol classes have been requested by the client application.

System Action: Unable to make further requests to this instance.

User Response: Check if client application specifies a valid protocol parameter.

CCL6664E Unable to load relevant class to support the {0} protocol.

Explanation: The relevant class could not be found.

System Action: Unable to create JavaGateway instance.

User Response: If the protocol {0} is xyz, then the required class is called xyzJavaGateway. Ensure that this can be found by your client program.

CCL6665E This JavaGateway instance has been closed.

Explanation: The JavaGateway instance was closed by the client application.

System Action: Unable to make a further request to this instance.

User Response: Start new JavaGateway instance.

CCL6666E Unable to flow request to the Gateway. [{0}]

Explanation: Where {0} specifies the cause of the error.

System Action: Unable to make further request to this instance.

User Response: Check error code.

CCL6667E Error writing request. [{0}]

Explanation: Where {0} specifies the cause of the error.

System Action: Unable to write further requests to this instance.

User Response: Check error code.

CCL6668E Initial handshake flow failed. [{0}]

Explanation: Caused by {0}

System Action: Unable to make initial handshake; hence unable to make requests.

User Response: Check error code.

CCL6669E Cannot open JavaGateway. Must specify both client-side and server-side security classes.

Explanation: Server application needs security.

System Action: Unable to make requests because of lack of security.

User Response: Specify both client and server sides security classes.

CCL6670E Exception occurred in the Gateway. [{0}]

Explanation: Where {0} specifies the cause of the error.

System Action: Not required.

User Response: Check error code.

CCL6671E This JavaGateway instance has yet to be opened.

Explanation: The client application has not opened a new JavaGateway instance yet.

System Action: Not required.

User Response: Open JavaGateway instance before making requests.

Error messages

CCL67nnl Default message.

Explanation: Default Trace message.

System Action: Not required.

User Response: Not required.

CCL6702I CICS Request: Successfully loaded native library.

Explanation: Native library for environment loaded successfully.

System Action: Not required.

User Response: Not required.

CCL6703I CICS Request: Load of native library failed {0}.

Explanation: Where {0} specifies the Java error.

System Action: Unable to load native library.

User Response: Check if native library is available.

CCL6704I BMS Map Converter

Explanation: The BMS Map Converter is a utility that generates Java Map classes from BMS definition files.

System Action: Not required.

User Response: Not required.

CCL6705I {0} 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.

CCL6706I Classes will be created in package {0}.

Explanation: When the BMS Map Converter generates classes, a package statement will be included, where the name of the package is {0}.

System Action: Not required.

User Response: Not required.

CCL6707I ScreenHandler will be generated.

Explanation: A ScreenHandler bean will be generated for this screen.

System Action: Generating ScreenHandler for screen.

User Response: Not required.

CCL6708I Use this command to generate Map classes from BMS Map files.

Explanation: This command generates Map classes from BMS Map files.

System Action: Not required.

User Response: Not required.

CCL6709I Arguments: <-p packagename > <-b> <-k exitAID> bmsfile1 bmsfile2 ..

Explanation: This help will be displayed if the BMS Map Converter is invoked without arguments.

System Action: Exit BMS Map Converter and display arguments accepted.

User Response: Enter valid arguments.

CCL6710I Processing file {0} ...

Explanation: The BMS Map Converter has begun processing the file {0}.

System Action: BMS Map Converter is processing this BMS Map.

User Response: Not required.

CCL6714I CICS Request: Cics_Rc {0} not recognised.

Explanation: CICS Server return code {0} is not valid.

System Action: Not required.

User Response: Check if Cics_Rc is valid.

CCL6716E CICS Request: Error in initialization. [{0}]

Explanation: Where {0} specifies the Java error.

System Action: CICS Request revolted.

User Response: Check the Java error.

CCL6717E CICS Request: An attempt to convert from a C byte array to a Java String failed.

Explanation: The codepage specified for the conversion was {0}.

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 {0}.

User Response: Check user codepage.

CCL6718E CICS Request: Error. [{0}]

Explanation: Where {0} specifies the error.

System Action: CICS Request revolted.

User Response: Check error.

CCL6719E CICS Request: Java default character encoding not obtained. [{0}]

Explanation: An attempt was made to obtain the Java default character encoding. {0} specifies the Java error.

System Action: Not required.

User Response: Check the return java error.

CCL6726I ECIRequest: ECI_BACKOUT request, Cics_Rc = {0}, Luw_Token = {1}.

Explanation: CICS Transaction back out.

System Action: CICS Server returns to the status it was before the last request.

User Response: Check Cics_Rc, if it is not 0 then check CICS Server and CICS Client.

CCL6730I ECIRequest: Commarea_Length limited to length of Commarea.

Explanation: Commarea_Length set to the length of Commarea specified; usually this is caused by user-defined Commarea being larger than the Commarea specified.

System Action: Set Commarea_Length to length of specified Commarea.

User Response: Reduce User-defined Commarea.

CCL6731W 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: Not required.

User Response: Use ECI_GET_REPLY and ECI_GET_REPLY_WAIT call types with caution.

CCL6732I ECIRequest: The array containing the status bytes is null. The ConnectionType, CicsServerStatus and CicsClientStatus will be set to their defaults.

Explanation: No status information has been returned, so the status fields will be set to their defaults.

System Action: Not required.

User Response: Not required.

CCL6733W 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: Not required.

User Response: Specify Larger Commarea.

CCL6734W 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: Not required.

User Response: Specify larger Commarea.

CCL6735I ECIRequest: Call_Type = {0}, Server = {1}, Extend_Mode = ECI_STATE_IMMEDIATE.

Explanation: Trace message for ECIRequest.

System Action: Not required.

User Response: Not required.

CCL6749I ESIRRequest: Userid, Password or Server Name too long

Explanation: Either the Userid, Password or Server Name entered is longer than required.

System Action: Not required.

User Response: Re-enter the correct Userid, Password or Server name.

CCL6755I EPIRequest: size field limited to data.length.

Explanation: The size field may not exceed the length of the data array.

System Action: Not required.

User Response: Specify larger data array.

CCL6762W 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.

Error messages

System Action: Not required.

User Response: Specify a termIndex.

CCL6770E ECIRequest: Invalid Call_Type.

Explanation: Invalid call type passed to ECIRequest

System Action: Not required.

User Response: Check call type.

CCL6771E 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.

CCL6772E 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.

CCL6776E CicsCpRequest: Unable to convert the CICS code page to a Java Encoding. [{0}]

Explanation: An attempt to convert a CICS code page to a Java format, (for example, 850 to Cp850), failed. {0} is the Java error.

System Action: Not required.

User Response: Check the Java error and the CICS Code page.

CCL6780E EPIRequest: Invalid Call_Type.

Explanation: Invalid call type passed to EPIRequest

System Action: Not required.

User Response: Check call type.

CCL6781E EPIRequest: Invalid event.

Explanation: Invalid event passed.

System Action: Not required.

User Response: Check event.

CCL6782E 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: Not required.

User Response: Specify a termIndex.

CCL6783E EPIRequest: termIndex {0} does not exist.

Explanation: Where {0} is the invalid termIndex.

System Action: Not required.

User Response: Check that termIndex is available and unique to this connection.

CCL6784E EPIRequest: termIndex {0} belongs to another connection.

Explanation: Where {0} is the invalid termIndex.

System Action: Not required.

User Response: Check that termIndex is unique to this connection.

CCL6785E EPIRequest: EPI Initialization failed. [{0}]

Explanation: Where {0} is the error.

System Action: Exit application.

User Response: Not required.

CCL6786I There are no free terminals available.

Explanation: The maximum number of terminals for this application has been reached.

System Action: Stop starting new transactions.

User Response: Wait for free terminal.

CCL6787E Exception [{0}] occurred while attempting to load classes from the path: {1}"

Explanation: An exception occurred while screen handler beans were 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: Not required.

User Response: Check if class path exist and classes are available.

CCL6788E Exception [{0}] occurred while attempting to load the class: {1}

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: Not required.

User Response: Check exception and that class is available and has a public constructor.

CCL6789E An exception has occurred: {0}

Explanation: Where {0} is the exception.

System Action: Not required.

User Response: Check where exception has occurred.

CCL6790E Too much data received from CICS.

Explanation: More data was returned by CICS than could be handled.

System Action: Not required.

User Response: Not required.

CCL6791E Internal error - terminal state invalid.

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

System Action: Not required.

User Response: Not required.

CCL6792I 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: Not required.

User Response: Check if a valid transaction ID is given.

CCL6793E 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.

CCL6794E Cursor could not be set to row {0}, column {1} - out of range.

Explanation: Either the row {0} or the column {1} are off the screen.

System Action: Not required.

User Response: Check if either the row or the column are out of range.

CCL6795E Unsupported datastream received from CICS - command is {0}.

Explanation: The datastream received from CICS could not be parsed.

System Action: Not required.

User Response: Not required.

CCL6796E Failed to start transaction {0}.

Explanation: Transaction {0} could not start.

System Action: Not required.

User Response: Check if transaction is valid.

CCL6797E 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.

CCL6798E Unknown attribute type {0}.

Explanation: The attribute type {0} is not supported.

System Action: Not required.

User Response: Check attribute type.

CCL6799I Return code {1} from {0} call.

Explanation: The EPI call {0} returned the return code {1}.

System Action: Not required.

User Response: Check return code; if it is not 0 than check for error.

CCL6901I Exception occurred [{0}]

Explanation: This message is added to the servlet log when an unexpected error occurs.

System Action: Not required.

User Response: Reset TerminalServlet and restart application.

Error messages

CCL6902I Initialization: loading property file {0}

Explanation: This message is added to the servlet log when the servlet loads a properties file.

System Action: Servlet opens properties file.

User Response: Not required.

CCL6903E Initialization failed. Property file {0} not found

Explanation: This message is added to the servlet log if the servlet cannot find the property file {0}.

System Action: The servlet will not start.

User Response: Check if property file exist and the location is correct.

CCL6904E Initialization failed. Error reading property file {0}

Explanation: The servlet could not read the property file {0}.

System Action: The servlet will not start.

User Response: Ensure that the servlet property file is in the correct format.

CCL6905W Initialization: property {0}={1} is invalid and has been ignored.

Explanation: This message is added to the servlet log if, during initialization, the servlet finds that an initialization parameter or property file entry is not valid. This could mean either that the property name is incorrect, or that the value specified is not a valid setting for the property.

System Action: The servlet continues running.

User Response: Check if property is valid.

CCL6906I Initialization completed successfully.

Explanation: This message is added to the servlet log when the servlet has started.

System Action: Servlet ready to start.

User Response: Not required.

CCL6907I The servlet is shutting down. In progress sessions will be terminated.

Explanation: This message is added to the servlet log when the servlet begins shutting down.

System Action: The servlet will shut down.

User Response: Not required.

CCL6908I The servlet has now stopped.

Explanation: This message is added to the servlet log when the servlet has been unloaded.

System Action: Unloading servlet.

User Response: Not required.

CCL6910W The requested action: {0} is unrecognized.

Explanation: This message is added to the servlet log if the value of the "request" parameter is unrecognized. The only supported values of "request" are "send" and "disconnect".

System Action: The request is not completed.

User Response: Check that request is send or disconnect.

CCL6911W Failed to allocate a terminal for this request.

Explanation: A terminal could not be allocated for a user. This could be because a terminal could not be connected to CICS (for example, if the server is down), or the maximum number of terminals are in use.

System Action: Unable to allocate a terminal.

User Response: Check server is running and maximum number of terminals has not been exceeded.

CCL6913W File {0} not found

Explanation: The file indicated could not be loaded by the servlet.

System Action: System unable to load file.

User Response: Check if file is available and is in the correct location.

CCL6914I Setting property {0} to the value: {1}

Explanation: If extended logging is on, this message is added to the servlet log when the servlet is requested to set a property value.

System Action: Set property value.

User Response: Not required.

CCL6915I Initialization: reading parameters

Explanation: This message is added to the servlet log when the servlet begins processing its initialization parameters.

System Action: Initialization reading parameters.

User Response: Not required.

CCL6916E Failed to initialize terminal pool. The error message is: {0}

Explanation: This message is added to the servlet log if an error occurs while creating the terminal pool. The servlet will not start.

System Action: Servlet exit.

User Response: Check error message.

CCL6917I Creating screen handler bean {0}

Explanation: If extended logging is on, this message is added to the servlet log when the servlet has loaded a screen handler bean.

System Action: Servlet loading screen handler bean.

User Response: Not required.

CCL6918I Getting the property named {0}

Explanation: If extended logging is on, this message is added to the servlet log when the servlet is requested to display a property value.

System Action: Servlet displays a property value.

User Response: Not required.

CCL6919W The property named {0} does not exist

Explanation: This message is added to the servlet log when the servlet is requested to set or to display a property which it does not recognize.

System Action: Servlet cannot find property.

User Response: Check property name is correct and exists.

CCL6920I Setting screen wrapper class to {0}

Explanation: Screen wrapper is being set to a different class.

System Action: Not required.

User Response: Not required.

CCL6921I Handling a request

Explanation: If extended logging is on, this message is added to the servlet log when a request is processed.

System Action: Execute request.

User Response: Not required.

CCL6923I Redirecting to page: {0}

Explanation: If extended logging is on, this message is added to the servlet log when the next page to display to the user has been determined.

System Action: Redirect to a loaded page.

User Response: Not required.

CCL6924E An error occurred while processing the request.

Explanation: An error occurred while a request was being processed. Further messages should indicate the cause of the problem.

System Action: Terminate request.

User Response: Check further messages for cause of problem.

CCL6925I Request processing has completed.

Explanation: If extended logging is on, this message is added to the servlet log when a request has been processed.

System Action: Request has been completed.

User Response: Not required.

Appendix. 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:

CICS

IBM

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, or other countries, or both.

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



Program Number: 5648-B43



Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC34-5593-00

