

# Message Class 1005 - RPC Server

If you are communicating with an EntireX RPC server, this class is returned if an error occurred in the RPC server.

If you are communicating with a Natural RPC Server and you receive errors in the range 10050001 - 10059999 on the client side, make a note of the last 4 digits of the error number and refer to the Natural error code documentation to determine the cause of the error instead here. For example, if you receive error number 10050082, Natural error 0082 has occurred on the server side.

Messages of this class have the format:

1005*nnnn*

where 1005 is the message class, and

*nnnn* is the message number in the range 0000 - 9999.

## Overview of Messages

10050000	10050001	10050002	10050003	10050004	10050005
10050006	10050007	10050008	10050009	10050010	10050011
10050012	10050013	10050014	10050015	10050016	10050017
10050018	10050019	10050021	10050022	10050023	10050024
10050025	10050026	10050027	10050028	10050029	10050030
10050031	10050032	10050033	10050034	10050035	10050036
10050037	10050038	10050039	10050040	10050041	10050042
10050043	10050044	10050045	10050046	10050047	10050048
10050049	10050050	10050051	10050052	10050053	10050054
10050055	10050056	10050057	10050058	10050059	10050060
10050061	10050062	10050063	10050064	10050065	10050066
10050067	10050068	10050069	10050074	10050075	10050076
10050077	10050082	10050083	10050084	10050085	10050086
10050087	10050088	10050089	10050090	10050091	10050092
10050093	10050094	10050095	10050096	10050098	10050099
10050100	10050101	10050102	10050103	10050104	10050110
10050111	10050112	10050113	10050114	10050116	10050117
10050118	10050119	10050120	10050122	10050123	10050124
10050125	10050126	10050127	10050129	10050130	10050131
10050132	10050134	10050135	10050136	10050137	10050138
10050139	10050140	10050142	10056971	10056974	10056977
10056978					

---

**10050000**

**Reserved**

**10050001      Cannot open or read File=<file name>**

**Explanation**    An invalid configuration file name or a wrong file type was used to start the RPC server.

**Action**            Restart the server with correct configuration file.

**10050002                      Cannot open or read File=<file name>**

**Explanation**                      Configuration problem.

**Action**                      Contact Software AG support

**10050003      Bad configuration: endworkers set to timeout**

**Explanation**    RPC server was started with an incomplete configuration file: Parameter `endworkers` has been set to timeout.

**Action**            Restart server with desired definition.

**10050004      Bad configuration: MinWorkers set to <number>**

**Explanation**    RPC server was started with an inconsistent configuration file: Parameter `minworkers` was either less than 1 or greater than the available thread entries.

**Action**            Restart server with desired definition.

**10050005      Bad configuration: MaxWorkers set to <number>**

**Explanation**    RPC server was started with an inconsistent configuration file: Parameter `maxworkers` was either greater than the available thread entries or (if `endworkers=never` was not specified) was less than 2.

**Action**            Restart server with desired definition.

**10050006      MaxWorkers must not be less than MinWorkers**

**Explanation**    RPC server was started with an inconsistent configuration file: Parameter `minworkers` was greater than `maxworkers`.

**Action**            Restart server with desired definition.

**10050007      ERXInit() failed**

**Explanation** Initialization of ERX runtime failed. This message is accompanied by an ERX runtime error message.

**Action** Check ERX runtime error message for further action.

**10050008      Create Instance failed**

**Explanation** A new thread could not be established. This message is accompanied by message 10050009.

**Action** See message 10050009 for further action.

**10050009      Search for free entry in Thread table failed**

**Explanation** The predefined thread table is full. This table normally is 256 entries long, under CICS there are 38 entries located in the ERX main control block. With ENDW=USER, each user thread requires an additional entry.

**Action** Check reason of the overflow: if a thread abnormally terminates (abends), the thread entry might be marked as UNUSABLE. If there are too many abnormal terminations during the lifetime of the RPC server, the table might be shortened. In other cases, contact Software AG support.

**10050010      Get Server status failed**

**Explanation** Internal function failed. This message follows a genuine error message.

**Action** Check original ERX error message for further action.

**10050011      Set Server status failed**

**Explanation** Internal function failed. This message follows a genuine error message.

**Action** Check original ERX error message for further action.

**10050012      Follow Worker creation failed**

**Explanation** Internal function failed. This message follows a genuine error message.

**Action** Check original ERX error message for further action.

**10050013 ERXRegister() failed**

**Explanation** Register call to ERX runtime failed. This message is accompanied by a genuine error message.

**Action** Check original ERX error message for further action.

**10050014 ERXLogon() failed**

**Explanation** Logon call to ERX runtime (logon to Broker) failed. This message is accompanied by a genuine error message.

**Action** Check original ERX error message for further action.

**10050015 ERXLogoff() failed**

**Explanation** Logoff call to ERX runtime (logoff from Broker) failed. This message is accompanied by a genuine error message.

**Action** Check original ERX error message for further action.

**10050016 ERXUnregister() failed**

**Explanation** Unregister call to ERX runtime failed. This message is accompanied by a genuine error message.

**Action** Check original ERX error message for further action.

**10050017 Server Statistics:**

**Explanation** Informational. This message is the header of the statistic messages 10050018 - 10050023 which will be printed when RPC server terminates.

**Action** None.

**10050018 Conversations successful ..... <number>**

**Explanation** Informational. Prints the number of conversational and non-conversational calls.

**Action** None.

**10050019          Workers successful ..... <number>**

**Explanation**      Informational. Prints the number of worker threads with normal end.

**Action**              None.

**10050021          Workers failed in executing ..... <number>**

**Explanation**      Informational. Prints the number of worker threads with abnormal end.

**Action**              None.

**10050022          Workers maximum parallel active ..... <number>**

**Explanation**      Informational. Prints the maximum number of concurrently active worker threads.

**Action**              None.

**10050023          Workers failed in starting ..... <number>**

**Explanation**      Informational. Prints the number of failing worker threads.

**Action**              None.

**10050024          ERXEndServing() failed**

**Explanation**      EndServing call to ERX runtime ( Broker DEREGISTER) failed. This message is accompanied by a genuine error message.

**Action**              Check original error message for further action.

**10050025          Check Worker creation failed**

**Explanation**      Internal function failed. This message is accompanied by a genuine error message.

**Action**              Check original error message for further action.

**10050026          Create Worker failed**

**Explanation**      Internal function failed. This message is accompanied by a genuine error message.

**Action**              Check original error message for further action.

**10050027 Set Worker status failed**

**Explanation** Internal function failed. This message is accompanied by a genuine error message.

**Action** Check original error message for further action.

**10050028 Check for Worker stop failed**

**Explanation** Internal function failed. This message is accompanied by a genuine error message.

**Action** Check original error message for further action.

**10050029 Check for new Worker to create failed**

**Explanation** Internal function failed. This message is accompanied by a genuine error message.

**Action** Check original error message for further action.

**10050030 ERXDynamicParmGeneration() failed**

**Explanation** Internal function to prepare the data conversation failed. This message is accompanied by a genuine error message.

**Action** Check original error message for further action.

**10050031 ERXUnmarshal() failed for <library> <program>**

**Explanation** Internal function for data conversation failed, *<library>=requested Library, <program>=requested Program*. This message is accompanied by a genuine error message.

**Action** Verify that the data received from client is correct. It may be necessary to reproduce the error while tracing is on.

**10050032 ERXMarshal() failed for <library> <program>**

**Explanation** Internal function for data conversation failed, *<library>=requested Library, <program>=requested Program*. This message is accompanied by a genuine error message.

**Action** Verify that the data sent by a server subroutine is correct. It may be necessary to reproduce the error while tracing is on.

**10050033      Could not resolve RPC call for <library> <program>**

**Explanation**    A client request could *not* be answered successfully. The client is notified with the appropriate protocol error, depending on the genuine error. This message is accompanied by a genuine error message.

**Action**            Check original messages for possible errors.

**10050034      Exception Type=<type> received!**

**Explanation**    SignalHandler indicates critical error. Depending on the operating system, dump information will be provided. Message 10050060 provides information about the program that is involved in this abend.

**Action**            Check message 10050060 for more information.

**10050035      ERXBeginServing() failed**

**Explanation**    BeginServing call to ERX runtime (Broker REGISTER) failed. This message is accompanied by a genuine error message.

**Action**            Check original error message for further action.

**10050036      SAG EntireX RPC server <Runtime platform>**

**Explanation**    Informational. This message is the header of the information messages 10050037 - 10050042 and 10050051 - 10050052 which are printed when RPC server is started.

**Action**            None.

**10050037            Minworkers (free for new conversation) <number>**

**Explanation**            Informational. Prints `minworkers` startup parameters.

**Action**            None.

**10050038            Maxworkers (max parallel active) ..... <number>**

**Explanation**            Informational. Prints `maxworkers` startup parameters.

**Action**            None.

**10050039            Endworkers (criteria when to stop) ... <number>**

**Explanation**            Informational. Prints `endworkers` startup parameters.

**Action**                    None.

**10050040            Timeout (in seconds for Broker) ... <seconds>**

**Explanation**            Informational. Prints `timeout` startup parameters.

**Action**                    None.

**10050041            API used (for Broker) ..... <api-version>**

**Explanation**            Informational. Prints `ETB_APIVERS` startup parameters.

**Action**                    None.

**10050042            Codepage (for data conversion) ..... <CodePage>**

**Explanation**            Informational. Prints `CODEPAGE` startup parameters.

**Action**                    None.

**10050043    Invalid Thread Status**

**Explanation**    Internal check has detected an error in thread status logic. The reason might be system shortages. The thread might be flagged unusable.

**Action**            Check if the server can still continue working. If not, restart server. If problem persists, contact Software AG support.

**10050044    ERXDynDispatch() failed**

**Explanation**    Dispatcher call to ERX runtime (Broker `RECEIVE`) failed. This message is accompanied by a genuine error message.

**Action**            Check original error message for further action.

**10050045    SERVER**

**Explanation**    Informational. This token is used to identify the originator thread of a message.

**Action**            None.



**10050046 Worker**

**Explanation** Informational. This token is used to identify the original thread of a message.

**Action** None.

**10050047 Transaction Control Callback failed**

**Explanation** Internal function failed. This message is accompanied by a genuine error message.

**Action** Check original error message for further action.

**10050048 Server Control Callback failed**

**Explanation** Internal callback function failed. This message is accompanied by a genuine error message.

**Action** Check original error message for further action.

**10050049 Client Identification Callback failed**

**Explanation** Internal callback function failed. This message is accompanied by a genuine error message.

**Action** Check original error message for further action.

**10050050 Init Globals failed**

**Explanation** Internal function failed. This message is accompanied by a genuine error message.  
CICS: This message will also be used to dynamically initialize a shared memory block within the ERX Main control block.

**Action** Check original error message for further action. CICS: Recompile and/or reload ERX Main control block.

**10050051 Broker Id (default) ... <BrokerID>**

**Explanation** Informational. Prints BROKERID startup parameters (entered in the command line or configuration parameter `BrokerID`).

**Action** None.

**10050052      Service Name (default) ..... <ClassName><ServerName><Service>**

**Explanation**    Informational. Prints CLASS SERVER SERVICE startup parameters (entered in the command line or configuration parameter Class, ServerName and Server).

**Action**            None.

**10050053      Load of Memory <MapName> failed**

**Explanation**    CICS only: ERX Main control block could not be loaded.

**Action**            Check CICS resource definitions and if module exist in DFHRPL load library.

**10050054      Invalid Memory Version=<version> found in <MapName>**

**Explanation**    CICS only: Invalid version detected in ERX Main control block. Either the version is out of support or (most likely) the control is corrupt.

**Action**            Recompile ERX Main control block with updated ERXMAIN macro.

**10050055      Read of Memory <MapName> failed**

**Explanation**    The message indicates an error with ERX Main control block. This message is accompanied by a genuine error message.

**Action**            Check original messages for possible errors.

**10050056      Usage : <SeverName> Broker@Server/Class/Service [-Option]  
                  or :  
                  <SeverName> Broker Class Server Service [-Option]  
                  or :  
                  <SeverName> CFG=ConfigFile.cfg [-Option]**

**Explanation**    Usage error. The correct syntax will be displayed. The server will terminate.

**Action**            Start server again with correct syntax.

**10050057      or : <ServerName> MEM=memory**

**Explanation**    Operating system-dependent USAGE message, part of message 10050056.

**Action**            Start server again with correct syntax.

**10050058**      **Option : -s[ilent] suppresses input from STDIN Default: <ServerName> ETB001  
RPC SRV1 CALLNAT**

**Explanation**    Operating system-dependent USAGE message, part of message 10050056.

**Action**            Start server again with correct syntax.

**10050059**            **Press Return key to continue.**

**Explanation**    Part of message 10050056.

**Action**            Confirm usage messages and start server again with correct syntax.

**10050060**      **Program program-name abended with exception= exception-code**

**Explanation**    (Customer) server subroutine <program-name> has abended. Depending on the operating system, dump information will be provided. The server will continue running. The thread in which the abend occurred will be terminated or (if not possible) will be marked as UNUSABLE.

**Action**            Correct the malfunctioning subroutine.

**10050061**                      **Not yet implemented**

**Explanation**                      Informational message.

**Action**                              None.

**10050062**      **ERXTerminateServer() failed**

**Explanation**    The complex function `ERXTerminateServer` will communicate with the Broker to stop all existing replicas for this server. This message is accompanied by a genuine error message.

**Action**            Check original error message for further action.

**10050063      Conversations aborted ..... number**

**Explanation** Informational. Prints the number of "unsuccessful" calls:

- all calls that return with error because of server subroutine problem and/or data conversations,
- Broker (timeout) terminated the call
- client terminates during conversation
- in conversational mode: the conversation did not finish with DISCONNECT-COMMIT.

**Action** None.

**10050064      Thread stopped because of major problems**

**Explanation** Internal error in server. This message is accompanied by a genuine error message.

**Action** Check original error message for further action.

**10050065      ERXFreeVData() failed**

**Explanation** Service function failed. This message is accompanied by a genuine error message.

**Action** Check original error message for further action.

**10050066      The max. thread entries available ..... number**

**Explanation** Informational. This message is printed to a trace file. The maximum number of thread entries is 256.  
Exception: under CICS it is automatically calculated from free memory in the ERX Main control block.

**Action** None.

**10050067      Server is starting.....**

**Explanation** Informational. This message is accompanied by a timestamp. The message will be generated after the Broker availability has been successfully tested. See also message no. 10050069.

**Action** none

**10050068      Server is stopping.....**

**Explanation**      Informational. This message is accompanied by a timestamp.

**Action**              none.

**10050069      Broker not available, has tried to connect x times**

**Explanation**      The EntireX RPC server detected that the Broker is not available (either during startup or at runtime). Depending on the restartcycles option, the server will try to logon again after a one-minute timeout. If the Broker becomes available during these restart cycles, the server will continue with message 10050067. Otherwise, it will terminate.

**Action**              Restart Broker, if desirable.

**10050074      Broker ID (in use) <BrokerID>**

**Explanation**      Informational. Prints the Broker ID with which a session is requested. This Broker ID can be different from the default Broker ID.

**Action**              None

**10050075      Service (in use) <ClassName>, <ServerName>, <Service>**

**Explanation**      Informational. Prints the class name, server name and service with which a connection is established. The values can be different from the default or the logical service

**Action**              None

**10050076      ERX runtime version and RPC server not compatible**

**Explanation**      Your EntireX Installation is destroyed.

**Action**              Reinstall EntireX.

**10050077      Response number from RPC server**

**Explanation**      The EntireX RPC server returned an error.

**Action**              See under error *number* for further explanation and action to take.

**10050082 Callee not found**

**Explanation** The EntireX/Natural RPC Server did not find the server program.

**Action** Depending on the type of your RPC server, one of the following situations may apply:

1. If the server is an EntireX RPC server on Windows/UNIX:

- The server stub with the name *D<library>.dll* on Windows or *D<library>.so/sl* on UNIX must be accessible to the EntireX RPC server. The program called must be available within that library as an export named *D<program>*.
- The same applies to the server program. A server with the name *<library>.dll* on Windows or *<library>.so/sl* on UNIX must be accessible to the EntireX RPC server. The program called must be available within that library as an export named *<program>*.

Check read/execution access, library names and exports.

2. If the server is an EntireX RPC server on CICS:

- A program named *<program>* must exist in the DFHRPL steplib concatenation.
- The program must be CSD-defined and (depending on CICS version) must be installed.

3. If the server is a Natural RPC Server:

- If Natural Logon is switched on for the request or conversation, the program must be found within the target library (or its steplib concatenation).
- If Natural Logon is switched off for the request or conversation, the program must be found in the initial library (or its steplib concatenation) of the Natural RPC Server.

See also the appropriate Natural documentation.

4. If the server is an EntireX XML/SOAP RPC Server:

- The server classes must be accessible from the EntireX XML/SOAP RPC Server.
- Make sure that the path of the server classes is in the classpath.

**10050083 SMH is listening on port <port>**

**Explanation** Informational. Message will be displayed if SMH service was successfully established. The local listener port will be exposed.

**Action** None.

**10050084 Shared library EXXRPCSTP not found or in error**

**Explanation** Some platforms require the shared library EXXRPCTCP to establish SMH services.

**Action** Check your installation: On Windows, the library *exxrpctcp.dll* should be contained in <drive>:\SoftwareAG\EntireX\bin; on UNIX, the library *libexxrpctcp.so* should be contained in folder */opt/softwareag/EntireX/lib*.

**10050085 Major problems with TCP**

**Explanation** The EntireX RPC server could not use TCP to support SMH services. This message is issued if TCP requests fail completely.

**Action** Contact your system administrator.

**10050086 Invalid job control - WORKER-PARMS is empty**

**Explanation** The SDF variable *WORKER-PARMS* in S-procedure *START-RPC-SERVER* is not set up correctly. This message will be sent by RPC servers under BS2000/OSD only.

**Action** Correct the job control and rerun the job.

**10050087 Shared memory could not be established**

**Explanation** An error occurred while establishing the main or a worker task. This message will be sent by RPC servers under BS2000/OSD only.

**Action** Contact Software AG support.

**10050088 Abnormal termination during program execution**

**Explanation** The server program called by the EntireX/Natural RPC Server caused an abnormal end.

**Action** Check the program and correct the error.

**10050089 ERXSetSecurity() failed**

**Explanation** RPC server was started with option KERNELSECURITY, but failed.  
 This message is accompanied by a genuine error message.

**Action** Correct installation error and restart.

**10050090 Broker Kernel-Security..... <level>**

**Explanation** Informational.  
 RPC server runs with broker KERNELSECURITY=<level>.

**Action** None.

**10050091 Authentication failed: user=<userid>, reason=<reason>**

**Explanation** The host security refused an authentication request.

**Action** Inspect reason code. Retry if password mistyped, otherwise contact your security administrator.

**10050092 Security could not be initialized, reason=<reason>**

**Explanation** Internal PC routines could not be installed.

**Action** Inspect reason code and contact your system administrator.

**10050093 Reserved**

**Explanation**

**Action**

**10050094 Cannot reset authentication, reason=<reason>**

**Explanation** The host security refused to reset a previously performed authentication request.

**Action** Inspect reason code and contact your security administrator.



**10050095 ERX Runtime not found**

**Explanation** EntireX RPC server requires the ERX shared library to be available. On Windows this is usually located in directory <drive>:\SoftwareAG\EntireX\bin.

**Action** Check EntireX installation. Verify that the *erx.dll* is in standard installation path.

**10050096 PL/I stack level exceeded**

**Explanation** RPC server received request for target PL/I application but application interface has higher indent level than PL/I supports (32).

**Action** Correct RPC client application.

**10050098 RPC server does not support RPC protocol version <rpc-version> found in server mapping**

**Explanation** The server mapping file accessed with IDL library and IDL program for the RPC request contained an RPC protocol version not supported by the RPC server. See *Supported RPC Protocols*. A server mapping file is an EntireX Workbench file with extension *.svm* or *.cvm*. See *Server Mapping Files for COBOL*.

**Action** Contact Software AG support for RPC server upgrade.

**10050099 Server mapping format is invalid**

**Explanation** The server mapping file accessed with IDL library and IDL program for the RPC request contained an invalid format. A server mapping file is an EntireX Workbench file with extension .svm or .cvm. See *Server Mapping Files for COBOL*.

**Action** Check that you have the latest updates for the EntireX Workbench installed. Try to produce the server mapping again, see *COBOL Wrapper* or *IDL Extractor for COBOL* and retry.

- If you are using *server-side* mapping files (.svm), deploy the correct server mapping to the RPC server. See *Deploying Server-side Mapping Files to the RPC Server* in the RPC server documentation for z/OS (CICS, Batch, IMS) | Micro Focus | CICS ECI | IMS Connect | BS2000/OSD | z/VSE (CICS | Batch).

**Note:**

For IMS Connect and CICS ECI connections with the webMethods EntireX Adapter, server-side mapping files are not deployed. They are wrapped into the Integration Server connection - the same as client-side mapping files. You need to update your Adapter connection. See *Step 3: Select the Connection Type* in the Integration Server Wrapper documentation. For RPC connections, deployment to the target RPC server is mandatory.

- If you are using *client-side* mapping files (.cvm), you need to rebuild all RPC clients communicating with this RPC server program. Select the appropriate wrapper (see *EntireX Wrappers*) and re-generate the client interface objects. For RPC connections with the webMethods EntireX Adapter, you need to update your Adapter connection. See *Step 3: Select the Connection Type* in the Integration Server Wrapper documentation.

If this fails, contact Software AG support.

**10050100**

**Reserved**

**10050101 Server mapping file in use..... Yes**

**Explanation** If the RPC server uses a server-side mapping container, this message will be printed, otherwise no message. See *Server-side Mapping Files in the RPC Server* in the RPC server documentation for z/OS (CICS, Batch, IMS) | BS2000/OSD | z/VSE (CICS, Batch).

**Action** None.

**10050102      EXTRACTOR service name reserved for internal use**

**Explanation** The RPC server did not come up. In the RPC server configuration, EXTRACTOR is specified as the broker service name, but EXTRACTOR is used for internal purposes of the RPC server and is a Software AG reserved service name.

**Action** Do not specify EXTRACTOR as the broker service name in the RPC server configuration, and restart the RPC server.

**10050103      DEPLOYMENT service name reserved for internal use**

**Explanation** The RPC server did not come up. In the RPC server configuration DEPLOYMENT is specified as the broker service name, but DEPLOYMENT is used for internal purposes of the RPC server and is a Software AG reserved service name.

**Action** Do not specify DEPLOYMENT as the broker service name in the RPC server configuration, and restart the RPC server.

**10050104      Deployment service defined but could not open SVM file**

**Explanation** The RPC server did not come up. In the RPC server configuration, the deployment service is specified (see Deployment Service for z/OS (CICS, Batch, IMS) | BS2000/OSD | z/VSE (CICS, Batch)), but the server-side mapping container could not be opened.

**Action** Check your installation. Possible reasons:

- The server-side mapping container is missing. For configuration see *Enabling the Deployment Service* for z/OS (CICS, Batch, IMS) | BS2000/OSD | z/VSE (CICS, Batch).
- The server-side mapping container has mismatched DCB parameter.
- ....

**10050110 Call for CICS DFHCOMMAREA exceeds 32K limit**

**Explanation** The CICS RPC Server tried to call an RPC server with an interface of type *CICS with DFHCOMMAREA Calling Convention* (COBOL Wrapper | Extractor), but the length of the data exceeded the CICS DFHCOMMAREA maximum of 32 KB.

**Action** Check the interface type of the target RPC server program called:

- If it is *not* a DFHCOMMAREA interface but a Large Buffer or Channel Container interface, the server mapping file may be missing or is wrong. A server mapping file is an EntireX Workbench file with extension .svm or .cvm. See *Server Mapping Files for COBOL*.
  - If you are using *server-side* mapping files (.svm), deploy the correct server mapping to the RPC server. See *Deploying Server-side Mapping Files to the RPC Server* in the RPC server documentation for z/OS (CICS, Batch, IMS) | Micro Focus | CICS ECI | IMS Connect | BS2000/OSD | z/VSE (CICS | Batch).

**Note:**

For IMS Connect and CICS ECI connections with the webMethods EntireX Adapter, server-side mapping files are not deployed. They are wrapped into the Integration Server connection - the same as client-side mapping files. You need to update your Adapter connection. See *Step 3: Select the Connection Type* in the Integration Server Wrapper documentation. For RPC connections, deployment to the target RPC server is mandatory.

- If you are using *client-side* mapping files (.cvm), you need to rebuild all RPC clients communicating with this RPC server program. Select the appropriate wrapper (see *EntireX Wrappers*) and re-generate the client interface objects. For RPC connections with the webMethods EntireX Adapter, you need to update your Adapter connection. See *Step 3: Select the Connection Type* in the Integration Server Wrapper documentation.
- If it is a DFHCOMMAREA interface, it is not possible to call such an RPC server program. Check if you can migrate the RPC server program to a Large Buffer or Channel Container interface. See *Supported COBOL Interface Types* under *Introduction to the IDL Extractor for COBOL* or *COBOL Server Interface Types* under *Introduction to the COBOL Wrapper*.

**10050111      Cannot load interface library EXX2MF**

**Explanation**    The EntireX server was started as Micro Focus server, but the interface library *EXX2MF* was not found or could not properly be initialized.

**Action**            ● Check if the library *EXX2MF* can be found in the standard EntireX installation directory and if it is valid.

                     ● Check for the Micro Focus installation if the library directory is located in the OS-library path.

**Note:**

Software AG might not support Micro Focus on your platform.

**10050112      EXX2MF initialization error**

**Explanation**    The EntireX interface library *EXX2MF* could not properly perform (thread) initialization.

**Action**            Check the Micro Focus installation.

**10050113      Invalid usage of SAG service**

**Explanation**    EntireX services EXTRACTOR or DEPLOYMENT were used other than by broker services EXTRACTOR or DEPLOYMENT.

**Action**            ● For the EntireX RPC client: use broker service EXTRACTOR and broker service DEPLOYMENT respectively.

                     ● For the EntireX RPC server: configure the RPC server with EXTRACTOR=YES and DEPOLOYMENT=YES.

                     ● Allow services EXTRACTOR and DEPLOYMENT in the Broker attribute file.

**10050114      No matching user ID**

**Explanation**    The EntireX RPC server runs with option IMPERSONATION= AUTO , SAME\_USER. This requires the client to send both, the EntireX user ID as well as the RPC user ID and the IDs must be equal.

**Action**            Correct your EntireX client.

**10050116      Deployment service not supported in that configuration**

**Explanation**    Deployment service is feasible only when using the programming language COBOL for the server programs (see also configuration parameter MARSHALLING in the respective RPC documentation section).

**Action**            Check the RPC server configuration options for consistency.

**10050117      Server mapping file (SVM) parameter required**

**Explanation**    By default, COBOL support under UNIX or Windows requires a server-side mapping container, but no container was specified.

**Action**            Correct the RPC server configuration and provide a server-side mapping container; see *Server-side Mapping Files in the RPC Server*.

**10050118      No SVM file/record found but required**

**Explanation**    The RPC server runs with COBOL support but no server mapping file was found for this particular RPC request, where one is required. A server mapping file is an EntireX Workbench file with extension .svm or .cvm. See *Server Mapping Files for COBOL*.

**Action**

- If you are using *server-side* mapping files (.svm), deploy the correct server mapping to the RPC server. See *Deploying Server-side Mapping Files to the RPC Server* in the RPC server documentation for z/OS (CICS, Batch, IMS) | Micro Focus | CICS ECI | IMS Connect | BS2000/OSD | z/VSE (CICS | Batch).

**Note:**

For IMS Connect and CICS ECI connections with the webMethods EntireX Adapter, server-side mapping files are not deployed. They are wrapped into the Integration Server connection - the same as client-side mapping files. You need to update your Adapter connection. See *Step 3: Select the Connection Type* in the Integration Server Wrapper documentation. For RPC connections, deployment to the target RPC server is mandatory.

- If you are using *client-side* mapping files (.cvm), you need to rebuild all RPC clients communicating with this RPC server program. Select the appropriate wrapper (see *EntireX Wrappers*) and re-generate the client interface objects. For RPC connections with the webMethods EntireX Adapter, you need to update your Adapter connection. See *Step 3: Select the Connection Type* in the Integration Server Wrapper documentation.

**10050119 Wrong SVM record type found**

**Explanation** The server mapping file accessed with IDL library and IDL program is inconsistent and does not match the RPC request. A server mapping file is an EntireX Workbench file with extension .svm or .cvm. See *Server Mapping Files for COBOL*.

**Action** Check you have the latest updates for the EntireX Workbench installed. Try to produce the server mapping again and retry. See *COBOL Wrapper* or *IDL Extractor for COBOL*.

- If you are using *server-side* mapping files (.svm), deploy the correct server mapping to the RPC server. See *Deploying Server-side Mapping Files to the RPC Server* in the RPC server documentation for z/OS (CICS, Batch, IMS) | Micro Focus | CICS ECI | IMS Connect | BS2000/OSD | z/VSE (CICS | Batch).

**Note:**

For IMS Connect and CICS ECI connections with the webMethods EntireX Adapter, server-side mapping files are not deployed. They are wrapped into the Integration Server connection - the same as client-side mapping files. You need to update your Adapter connection. See *Step 3: Select the Connection Type* in the Integration Server Wrapper documentation. For RPC connections, deployment to the target RPC server is mandatory.

- If you are using *client-side* mapping files (.cvm), you need to rebuild all RPC clients communicating with this RPC server program. Select the appropriate wrapper (see *EntireX Wrappers*) and re-generate the client interface objects. For RPC connections with the webMethods EntireX Adapter, you need to update your Adapter connection. See *Step 3: Select the Connection Type* in the Integration Server Wrapper documentation.

If this fails, contact Software AG support.

**10050120 WARNING! API-area was exceeded! Lib=<Library> Pgm=<program>**

**Explanation** EntireX server has found an application that exceeded the calculated API size. The impact depends on how much the limit is exceeded by.

**Action** Check that the program interface definition (IDL file) matches the server's interface.

**10050122 EntireX RPC server error 122 occurred**

**Explanation** An error occurred within the RPC Server runtime while deallocating storage during worker task shutdown.

**Action** None.

**10050123 Invalid transaction ID**

**Explanation** The (CICS) RPC server was started with transaction ID different to the transaction-id that was specified in ERXMAIN control block.

**Action** The transaction IDs must be unique.

**10050124 Server already active**

**Explanation** An attempt was made to start (CICS) RPC server a second time.

**Action** This message was issued by the RPC server (see also message 10170004). Only one server instance can be started using the same shared memory block (ERXMAIN control block).

**10050125 No free server resources available**

**Explanation** An attempt was made to start an RPC worker or user task, but the allocated thread table has no more free entries.

**Action** Either reduce the MAXW parameter or increase the SIZE parameter. Check also the server log file for errors, because this situation may also be caused by aborting threads.

**10050126 RPC Server is initialized**

**Explanation** Message of RPC server running under z/OS. Informational. Supports process automation.

**Action** None.

**10050127 RPC in closure**

**Explanation** Message of RPC server running under z/OS. Informational. Supports process automation.

**Action** None.

**10050129 Invalid parameter/program search order**

**Explanation** Internal error.

**Action** Contact Software AG support



**10050130      Service not supported. (<service>)**

**Explanation** The server was configured to use a feature that is not supported under this operation system.

**Action** Change server configuration.

**10050131      Setting CICS sync point failed (<resp>, <resp2>)**

**Explanation** RPC server under CICS tried to perform EXEC CICS SYNCPOINT (due to end of conversation) but failed.

**Action** Check CICS response codes for this error.

**10050132      Server option <option> deprecated**

**Explanation** At startup, the server found an option that is deprecated.

**Action** The message is informational. Normally, the option will still be supported for compatibility reasons, but support may be withdrawn in a future release. See *Release Notes* for recommended action.

**10050134                      Reserved for internal use.**

**10050135                      Reserved for internal use.**

**10050136                      Reserved for internal use.**

**10050137                      Reserved for internal use.**

**10050138                      Reserved for internal use.**

**10050139                      Reserved for internal use.**

**10050140                      Reserved for internal use.**

**10050142      RPC Server initialization timeout at worker creation**

**Explanation**    A final step of the server initialization is to launch the configured minimum number of worker tasks. If this does not succeed within 10 seconds, the server aborts the initialization.

**Action**            Verify server output for additional messages indicating problems with the worker initialization. Verify server trace output for additional error messages.

**10056971      EntireX RPC server error 6971 occurred**

**Explanation**    The cause of the error and corrective action depends on the type of your RPC server.

- Action**
- If the server is an EntireX RPC server, contact Software AG support because of internal problems within the server.
  - If the server is a Natural RPC Server, see the appropriate Natural documentation for error 6971.

**10056974      EntireX RPC server error 6974 occurred**

**Explanation**    The cause of the error and corrective action depends on the type of your RPC server.

- Action**
- If the server is an EntireX RPC server, increase your memory resources and retry the program because a memory request could not be fulfilled by the operating system.
  - If the server is a Natural RPC Server, see the appropriate Natural documentation for error 6974.

**10056977      EntireX RPC server error 6977 occurred**

**Explanation**    The EntireX/Natural RPC request could not be executed.

**Action**            Examine the log of the EntireX/Natural RPC Server for the cause of the malfunction.

**10056978      EntireX RPC server error 6978 occurred**

**Explanation**    The cause of the error and corrective action depends on the type of your RPC server.

- Action**
- If the server is an EntireX RPC server, the reason could be an invalid compression value.
  - If the server is a Natural RPC Server, refer to the appropriate Natural documentation for error 6978.