LNS and OpenLNS Runtime Errors

(9/06/13)

This document covers the following LNS and OpenLNS errors that you may see when using LonMaker, OpenLNS CT, other LNS applications, LNS Server, OpenLNS Server or the i.LON SmartServer: LNS, NS, NI, CONNERR, LNS, DS, Formatter, and VNI.

Any comments below for LonMaker also apply to OpenLNS CT, and any comments about LNS also apply to OpenLNS.

The LNS/OpenLNS Server runs automatically when using LonMaker/OpenLNS CT and is manually started when using most LNS Applications.

Solutions to some of these errors are provided in the Echelon Knowledge Base and are referenced by KB# (for example KB409). The link to the Knowledge Base is given below.

http://www.echelon.com/support/kb/search.asp

Not all Knowledge Base (KB) solutions are posted so if you don't see the number listed in this document then contact LonSupport for the solution.

Note that NSS and LCA are referring to the LNS Server or OpenLNS Server.

Network Service Errors

A network services error occurs when the LNS Server is unable to successfully complete a service. These errors will appear in the format NS: #<Error Number>.

lcaErrNsCancel   **NS, #1**

The service and the associated transaction were cancelled. This error occurs when an internal error occurs, such as the LNS engine restarting.

lcaErrNsNodeHasNoAppl   **NS, #5**

The device was discovered to be applicationless, or not associated with a program. If you encounter this error, load the application into the device. For host-based nodes, ensure that the host application is running.

lcaErrNsBadManagerId   **NS, #6**

The current manager ID does not correspond to a known manager. This is an internal error and should be reported to customer support.

lcaErrNsResourceProblem   **NS, #7**

A problem occurred in resource allocation. This is an internal error and should be reported to customer support.

See KB528.

lcaErrNsInterfaceFailure   NS, #8

The host was unable to communicate with the NSI. This may be due to problems communicating with the LNS engine, or with the network interface. If you encounter this error, ensure that the LNS engine process is running, and that the network interface is functioning properly.

See KB469.

lcaErrNsBadClient   NS, #9

The specified server (or the current server) is undefined. This is an internal error and should be reported to customer support.

lcaErrNsBadServer   NS, #10

The specified server (or the current server) is undefined.

lcaErrNsNoClient   NS, #11

No client is currently defined. This is an internal error and should be reported to customer support.

lcaErrNsNoTxInProgress   NS, #12

There is no transaction defined for the client. This warning can be expected when cancelling a method that is not part of an explicit transaction.

lcaErrNsLNSNotFound   NS, #13

The NSI has not been configured to communicate with the LNS. This error may occur when opening the database with the remote database collection if the clients NSD is not properly configured. To reconfigure the client's NSD, open the database remotely using the Networks collection.

lcaErrNsClientBlocked   NS, #14

The client is already involved in a service request. You should invoke only one service per client. This includes all requests to start or commit transactions. When a client has already launched a service, wait until the current service completes before you invoke a new service. This is an internal error and should be reported to customer support.

lcaErrNsTxInProgress   NS, #15

The client is already involved in a transaction. End the client's current transaction by calling CancelTransaction method, or wait for the transaction to complete and call the CommitTransaction method, before starting a new transaction for that client.

You should avoid starting multiple transactions on a single network database within a single application.

lcaErrNsNotImplemented   NS, #16

The specified service or option is not implemented by the specified server. This error could indicate a bad service parameter. This is an internal error and should be reported to customer support.

lcaErrNsErrorInfo   NS, #17

The service failed. When you encounter this error, a service-specific error status is available in the LastError property of the System object.

lcaErrLNServiceFailure   NS, #18

The requested service failed. This is an internal error, and should be reported to customer support.

lcaErrNsMsgRejectedByNode   NS, #19

A node returned a negative response to a command from LNS. This could be due to a LonTalk authentication failure. It may also result from an inconsistency between the program interface defined in the database and what the node actually supports.

See KB576.

lcaErrNsPrematureRelease   NS, #20

This is an internal error and should be reported to customer support.

lcaErrNsRootBusy   NS, #21

A transaction is still in progress. This is an internal error and should be reported to customer support.

lcaErrNsRootDoesNotExist   NS, #22

The service depends on a transaction that does not exist. This is an internal error and should be reported to customer support.

lcaErrNsNoFreeRootTransactions   NS, #23

This is an internal error and should be reported to customer support.

lcaErrNsCantFindObject   NS, #24

One of the objects needed to complete the service is not in the database of the LNS.

lcaErrNsDuplicateObject   NS, #25

An attempt was made to add an object that already exists.

lcaErrNsCommError   NS, #26

The LNS cannot communicate with the node. This may be because the Neuron ID referenced by the application is invalid or does not exist, the LNS or the node is not physically attached to the media, the LNS is not configured (because the MgmtMode property was not set to **lcaMgmtModePropagateConfigUpdates** after the database was initially created), problems with the physical media, or the LNS timers (set using the RetryCount and TxTimer properties of the System object) are too low.

See KB 149.

lcaErrNsOperationError   NS, #27

The node failed to go online, offline or to reset when requested to do so. This may be due to a problem in the implementation of the device.

lcaErrNsUnsupportedInfo   NS, #28

The requested information is not available from an application node. For example, transceiver status, SI/SD data, and network variable names are not always stored in the node.

LonMaker Turbo SP1 has some fixes

lcaErrNsOutOfRange   NS, #29

The value assigned to a property or parameter is either out of range, or invalid.

lcaErrNsEventFailure   NS, #30

The generation of an event failed. This is an internal error and should be reported to customer support.

lcaErrNsDeferConfigUpdatesMgmntMode   NS, #31

The requested service cannot be provided in the current management mode. When you encounter this error, set the system management mode to **lcaMgmtModePropagateConfigUpdates (0)**, and try the operation again. You can change the system management mode by writing to the MgmtMode property of the System object.

lcaErrLNSubscriptionDblimit   NS, #32

The LNS's event subscription table is full. This is an internal error, and should be reported to customer support.

lcaErrLNSequence   NS, #33

A record from a binary external interface file or binary application image file was received out of sequence. If you encounter the error, the XFB or APB file may be corrupted.

lcaErrLNSegmentation   NS, #34

Segmentation is not supported, or there is a problem with segment order. This is an internal error, and should be reported to customer support.

lcaErrNsChecksum   NS, #35

A checksum error occurred while transferring the binary external interface or binary application image file to the LNS. If you encounter the error, the XFB or APB file may be corrupted.

lcaErrLNSessionError   NS, #36

Invalid session handle or a session error. Make sure that sessions are begun and ended in pairs, and are always part of an explicit transaction.

See KB 149.

lcaErrNsNoNeuronId   NS, #37

An operation requiring a Neuron ID was attempted on a device with no Neuron ID. For example, if you invoke the Wink method on a device whose Neuron ID has not been set in the LNS database, this exception will be thrown. In some other cases, this exception may be thrown if a device's Neuron ID has been set in the LNS database, but the device has not been commissioned. For example, if you read the SelfDocumentation property of a device before that device has been commissioned, this exception will be thrown, regardless of whether or not the device's Neuron ID is set.

lcaErrNsProgramidMismatch   NS, #38

Program IDs do not match. This may occur if there is an incorrect or out-of-date program version.

lcaErrNsObjectInUse   NS, #39

A requested operation can't complete because the object involved in the operation is in use. For example, you may not be able to delete a node because that node is still included in connections.

lcaErrNsNodeHasNoSiData   NS, #40

The node has no SI (self-information) data, when required.

lcaErrNsDbError   NS, #41

An internal database error occurred. This is an internal error, and should be reported to customer support.

See KB469.

lcaErrNsOutOfJournal   **NS, #42**

No more memory is available for journaling. This is an internal error, and should be reported to customer support.

lcaErrNsOutOfMemory   **NS, #43**

No more memory is available for allocation.

lcaErrNsBadParms   **NS, #44**

Invalid parameter values.

See KB636.

lcaErrNsNetworkHasInstalledNodes   **NS, #46**

Attempt to add the LNS failed because application nodes have already been defined or added. You must install the LNS before the other nodes. This is an internal error, and should be reported to customer support.

lcaErrNsDbImportExport   **NS, #47**

Requested operation can't proceed because a database import or export is currently in progress.

lcaErrNsProgramNotFound   **NS, #48**

Could not find the program definition required to execute the service.

lcaErrNsIntegrityError   **NS, #49**

Internal integrity error. This is an internal error, and should be reported to customer support.

lcaErrNsDblimit   **NS, #50**

An unspecified database limit was exceeded.

See KB190.

lcaErrNsNodeNotFound   **NS, #51**

Cannot find the node.

lcaErrNsDuplicateNode   **NS, #52**

An attempt was made to add a node that is already defined.

lcaErrNsNodeDbLimit   **NS, #53**

The node limit has been exceeded.

lcaErrNsDuplicateProgram   **NS, #54**

The program template is already defined.

lcaErrNsProgramDblimit   **NS, #55**

This is an internal error, and should be reported to customer support.

lcaErrNsNvmtNotFound   **NS, #56**

The network variable or message tag cannot be found.

lcaErrNsNvmtDbLimit   **NS, #57**

This is an internal error, and should be reported to customer support.

lcaErrNsChannelNotFound   **NS, #58**

The channel cannot be found.

lcaErrNsProgramIntfMismatch   **NS, #59**

The new program interface does not match the previously defined program interface; i.e., the number of network variables, message tags, or the structure of self identification data is inconsistent even though the program IDs match. This error may also be thrown if the SelfDocConsistency property of a DeviceTemplate object is set to a value that conflicts with the configurations of the devices using that template.

Check if you are using the correct XIF file for the device. You can do one of the following:

1. Delete the device and the device template, and re-add/commission the device again using the correct XIF file. 2. Delete the device and the device template, and re-add/commission the device again using "Upload from Device" for external interface.

lcaErrNsProgramIntfUnsupported   **NS, #60**

The program interface does not support the requested action.

lcaErrNsMsgError   **NS, #61**

There was a failure in the messaging sub-system. This is an internal error, and should be reported to customer support.

lcaErrNsCancelError   **NS, #62**

An error occurred while canceling a transaction or service. This is an internal error, and should be reported to customer support.

lcaErrNsNothingToCancel   **NS, #63**

There was no service or transaction to cancel

lcaErrNsNodeNotInstalled   **NS, #64**

An operation on a node was requested that requires the node to be configured, but the node is not configured. When you encounter this error, make sure that the device in question has been configured.

lcaErrNsAuthViolation   **NS, #65**

Authentication rules were violated.

lcaErrNsChannelDblimit   **NS, #66**

The network channel limit (currently 1000) has been reached.

lcaErrLNSubnetDblimit   **NS, #67**

The network subnet limit (currently 255) has been reached.

See KB190.

lcaErrLNSubnetNotFound   **NS, #68**

The specified subnet ID was not found. Subnet IDs must be created automatically by the LNS or defined manually using the Add method of the Subnets collection object.

lcaErrNsRouterDblimit   **NS, #69**

The network router limit (currently 32766) has been reached.

lcaErrNsRouterNotFound   **NS, #70**

The specified router handle does not exist.

lcaErrNsTimerRange   **NS, #71**

The timer value computed for a connection or for the LNS Object Server to communicate with a node exceeds the range supported by LONWORKS devices. This is an internal error, and should be reported to customer support.

lcaErrNsWrongChannel   **NS, #72**

An attempt was made to add, commission, move, or replace a device that is on the wrong channel. A device is considered to be on the wrong channel when a configured, learning, or non-permanent bridge class router exists between the device and the channel on which it is to be placed. When you encounter this error, place the device or router on the correct channel, or define the device or router without a channel and have LNS determine the channel automatically

lcaErrNsInsufficientRouters   **NS, #73**

An attempt was made to create a connection between nodes on different channels, but no routers were available to complete a logical path between the channels. This error can occur when attempting to remove or move a router which has connections across it, or a router which connects an NSI to either the LNS or another NSI

lcaErrNsNoSessionInProgress   **NS, #74**

The EndSession method of the System object was called before a corresponding BeginSession was called. Make sure that a session is open before you call EndSession.

lcaErrNsTopologyPhysicalLoop   **NS, #75**

The attempted router operation would have created a loop in the physical network topology.

lcaErrNsTopologyLogicalLoop   **NS, #76**

The attempted router operation would have created a loop in the logical network topology. Normally, loops are initially detected as physical loops. However, it is possible to add a permanent repeater or bridge, so that a logical loop exists even though a physical loop does not.

lcaErrNsTopologySubnetViolation   **NS, #77**

An attempt was made to define a device with a subnet that is incompatible with the devices channel, due to router constraints. To resolve this error, either leave the subnet undefined (so LNS will define it) when you define the device, or make sure that the selected subnet is not in use on other logical channels.

lcaErrNsRouterMustBePermanent   **NS, #78**

An attempt was made to illegally add a non-permanent router, or to change the class of a permanent router to non-permanent. Once a permanent router has been added to the topology, it cannot be changed to a non-permanent router, even by removing and then re-adding it.

lcaErrNsCantDetermineChannel   **NS, #79**

An attempt was made to register or add a device without a specified channel handle, and the system was not able to automatically determine the channel on which the device resides. This error will occur if the channel the device resides on is connected to any other channel by repeaters or permanent bridges. Also, this error will occur if the channel is not currently configured in the system.

lcaErrNsRedundantRoutersMustBeConfigured   NS, #80

An attempt was made to add a redundant router that was not of the configured class, to change the class of a redundant router to the non-configured class, or to add a redundant router to an existing non-redundant, non-configured class router.

lcaErrNsCantDetermineXcvrId   NS, #81

A channel was defined with a wildcard transceiver ID, but the transceiver ID could not be automatically determined by the system.

lcaErrNsRouterCantBePermanent   NS, #82

An attempt was made to change a non-permanent router to a permanent router by writing to the ClassId property of the Router object. A router can be changed in this way only by being moved.

lcaErrNsObjectLocked   NS, #83

An attempt was made to change a locked object. This is an internal error, and should be reported to customer support.

lcaErrNsInvalidContext   NS, #84

An attempt was made to invoke a service from an event handler that was called directly from a background task. This is an internal error, and should be reported to customer support.

lcaErrLNServerNotFound   NS, #85

The specified server could not be found. The server ID may be invalid. This is an internal error, and should be reported to customer support.

lcaErrNsNodeStateError   NS, #86

The node was in the wrong state for the attempted operation. To resolve this error, try setting the State property of the node's AppDevice object to lcaStateCnfgOnline.

lcaErrNsLmobjNotFound   NS, #87

The specified LonMark object could not be found.

lcaErrNsLmobjDbLimit   NS, #88

The LonMark object limit has been exceeded.

lcaErrNsInvalidSidata   NS, #89

The self-documentation information was invalid (e.g., improper LonMark definitions).

lcaErrLNServiceCommError   NS, #90

A communications error occurred between the client and the server.

lcaErrNsRequestNotAllowed   NS, #91

The requested service has been disabled. This error is usually the result of a remote full client performing an operation that has been disabled by the server.

lcaErrNsCpNotFound   NS, #92

The specified configuration property does not exist.

lcaErrNsCpDbLimit   NS, #93

The configuration property database limit has been exceeded.

lcaErrNsCpRangeNotFound   NS, #94

The specified configuration property range does not exist.

lcaErrNsCpRangeDblimit   NS, #95

The configuration property range database limit has been exceeded.

lcaErrNsConnectionError   NS, #96

Connection rebinding failed for a set of moved nodes. Possible causes are: no more address table slots, no more groups, or broadcast violation.

lcaErrNsFileLookupError   NS, #97

An error occurred during a file operation, probably due to an out of range file index.

lcaErrNsFileIoError   NS, #98

The file operation target failed to read/write a file.

lcaErrNsFileTimeoutError   NS, #99

The target timed out during a file transfer operation.

lcaErrNsFileWindowError   NS, #100

The target received a packet out of order during a file transfer operation.

lcaErrNsFileAuthError   NS, #101

File transfer failed due to incorrect authentication.

lcaErrNsFileAccessUnavail   NS, #102

Random file access is not implemented on the target.

lcaErrNsFileOpenFailure   NS, #103

The target failed an open file operation.

lcaErrNsFileSeekInvalid   NS, #104

The target failed a file seek operation.

lcaErrNsCantModifyConstCp   NS, #105

An attempt was made to modify a constant configuration property.

lcaErrNsCpDefaultsNotFound   NS, #106

Default configuration parameters have not been uploaded to the database. To correct this, you should set default values from the current values in the device using the UploadConfigProperties method of the AppDevice object. Use the lcaConfigPropOptSetDefaults and lcaConfigPropOptLoadUnknown upload options when you call the method. To avoid this error in other databases, import a program template with a XIF containing CP default values.

lcaErrNsCantFindConnection   NS, #107

The requested connection was not found.

lcaErrNsLmobjMemNotFound   NS, #108

The requested LonMark object member was not found. You will encounter this error if you try to use the UnassignNetworkVariable method to remove a network variable from a LonMarkObject, but the network variable was not previously assigned to the LonMarkObject. You can use the AssignNetworkVariable method to assign a network variable to a LonMarkObject.

lcaErrLNStaleFileHandle   NS, #109

An attempt was made to use a stale file transfer handle. This is an internal error, and should be reported to customer support.

lcaErrNsFileLimitReached   NS, #110

The file transfer handle limit has been reached.

lcaErrNsFileContention   **NS, #111**

A file could not be accessed due to contention with the initiator. You can avoid this error by not performing file transfers with devices that are already engaged in file transfers with other devices.

lcaErrNsAccessExpired   **NS, #112**

The LNS demonstration software has expired. A new copy must be obtained.

lcaErrNsCpValueNotFound   **NS, #113**

A value could not be found for the specified configuration parameter. This exception will be thrown if you use the GetDataPoint method to create a data point with the **lcaDataSourceOptionsDatabaseOnly (2)** option set, and then attempt to read the value of the data point, but the value does not exist in the LNS database.

lcaErrNsNodeReset   **NS, #114**

An operation failure occurred due to an unexpected node reset.

lcaErrNsManagerNotAllowed   **NS, #115**

The specified services are not allowed by the manager. This is an internal error, and should be reported to customer support.

lcaErrNsUnsupportedService   **NS, #116**

Service attempted to reboot a node that is not a 3150.

lcaErrNsLabelNotFound   **NS, #117**

The requested label could not be found.

lcaErrNsDuplicateLabel   **NS, #118**

An object with this label already exists.

lcaErrNsNoRecoveryInProgress   **NS, #119**

There is no database recovery in progress for the recovery status service.

lcaErrNsCantModifyProgramIntf   **NS, #120**

The service attempted to modify a read-only program interface.

lcaErrNsCantModifyNvType   **NS, #121**

The service attempted to modify a read-only network variable type.

lcaErrNsNvNotLmobjMember   **NS, #122**

The network variable is not a member of a LonMark object.

lcaErrNsHostResourceProblem   **NS, #123**

There is a resource problem in the API or application.

lcaErrNsTxAlreadyCanceled   **NS, #124**

The specified transaction was already canceled.

lcaErrNsFirmwareVersionMismatch   **NS, #125**

The device contains an incompatible firmware version. This error will be generated if you attempt to load a new application image into a device with the Load method, and the system image (firmware version) used by the device is incompatible with the new application image. Some devices support the LoadEx method, which will upgrade the system image to a compatible version when this error occurs.

lcaErrNsUnimplementedCategory   **NS, #126**

The service encountered an unimplemented category or category operation. This is an internal error, and should be reported to customer support.

lcaErrNsUnimplementedProperty   **NS, #127**

The service encountered an unimplemented property or property operation. This is an internal error, and should be reported to customer support.

lcaErrNsDisallowedInMipMode   **NS, #128**

The specified service or operation is disallowed because the interface is not an NSI. This is an internal error, and should be reported to customer support.

lcaErrNsDisallowedInTxHandler   **NS, #129**

The specified service or operation is disallowed in the transaction notification handler. This is an internal error, and should be reported to customer support.

lcaErrNsLNSNotInitialized   **NS, #130**

The NsInit() function either was not called or failed. This is an internal error, and should be reported to customer support.

lcaErrNsLNSEngineNotFound   **NS, #131**

The LNS Win32 engine was not found. If you encounter this error, you should verify that LNS is installed properly on your machine.

lcaErrNsInvalidNsi   **NS, #132**

The network interface is not a valid NSI mip.

lcaErrNsFileHeaderError   **NS, #133**

The file which was accessed had an invalid header

lcaErrNsDbVersionError   **NS, #134**

Returned when the LNS encounters a database (or non-volatile data) with the wrong version number.

lcaErrNsNoNetworkInterface   **NS, #135**

Returned when an attempt is made set the MgmtMode property to **lcaMgmtModePropagateConfigUpdates (0)** without selecting a network interface.

lcaErrNsNsiInuse   **NS, #136**

Returned when an attempt is made to remove an NSI that is being used by other processes.

lcaErrNsDialingFailure   **NS, #137**

Failed to call host. Probably due to busy, no answer, or recent failure.

lcaErrNsNetworkInterfaceState   **NS, #138**

Local network interface is in the wrong state (e.g. unconfigured). When you encounter this error, set the network management mode to **lcaMgmtModePropagateConfigUpdates (0)** by writing to the MgmtMode property of the System object. Then, re-commission the network interface, if necessary.

See KB190.

lcaErrNsNetworkInterfaceConfig   **NS, #139**

Local interface is not properly configured or updated. When you encounter this error, set the MgmtMode property of the System object to **lcaMgmtModePropagateConfigUpdates (0)**. and recommission the network interface, if necessary.

lcaErrNsNetworkInterfaceInUse   **NS, #140**

Network interface is currently in use.

lcaErrNsInvalidNeuronId   **NS, #141**

The specified neuron ID is illegal. This error is reported if a Neuron ID containing zeroes in the middle 4 bytes is specified. Neuron Chips that use this format are defective and should be returned.

lcaErrNsDialupDataUninitialized   **NS, #142**

Device not configured to dialup to host upon LNS communication.

lcaErrNsCannotStartAppl   **NS, #143**

Dialup NSI was unable to start host application or LNS after connection.

lcaErrNsCapacityLimit   **NS, #144**

License capacity or credit limit reached. This error will occur when all the credits in the license, including deficit credits, have been used. For more information on licensing, see Chapter 13 of the *LNS Programmer's Guide.*

lcaErrNsCpNvLengthUnknown   **NS, #145**

Length of configuration NV implementing a CP is unknown.

lcaErrNsInvalidErrorContext   **NS, #146**

An update error had been reported prior to updating the nodes. This is an internal error and should be reported to customer support.

lcaErrNsLicenseViolation   **NS, #147**

LNS license access failure. A variety of conditions may cause this error. Some relate directly to internal licensing components. For example, the license DLLs may not have installed properly, the files that identifies the license (LNSeng.exe for the LNS Server, lonmaker.DSL for LonMaker) do not exist, or the Crypkey licensing components (crypserv.exe and ckldrv.sys) are not running properly. You can resolve these problems by deleting the files that may be causing problems, and re-installing LNS. The license DLLs, as well as the crypserv.exe and ckldrv.sys files, can be found in the Windows "LNS Licenses" folder, and the Crypkey files can be found in the Windows "system32" folder. Note that you must stop the "CrypkeyLicense" service and "NetworkX" driver before deleting the Crypkey files.

Certain conditions on the PC operating the license may also cause this error. For example, if the PC's file system is corrupted or low on space, or if the license files have been manually deleted, tampered with in any way, or moved by some disk-defragmentation utilities, this error will occur. You can resolve these situations by ordering a replacement license key. If the PC clock is set back to a time before the creation of the license or before the last time credits were purchased, this error will occur. In some cases, this can be resolved by rebooting the PC. Otherwise, it may be necessary to order a replacement key.

The error may occur if the license has been transferred out of the PC, manually terminated, or if there are zero maximum credits. In this case, you need to purchase additional credits for the license.

For more information on licensing, see Chapter 13 of the *LNS Programmer's Guide.*

For solution see KB540.

lcaErrNsLicenseExpired   **NS, #148**

LNS license time limit has expired. This error will occur if you are using a trial license and the number of days allocated to the license have expired, or if you are using a trial license and set the PC clock back.

For more information on licensing, see Chapter 13 of the *LNS Programmer's Guide.*

lcaErrNsConflictWithCurrentNetwork   **NS, #149**

This error will be generated if you open a network that is already opened using a different network interface, LNS type, or database directory.

lcaErrNsConflictWithAnotherNetwork   **NS, #150**

This error will be generated if you open a network that specifies the same database directory as another currently opened network.

lcaErrNsBatchOptionNotImplemented   **NS, #151**

The batch service option combination specified is not implemented. This is an internal error, and should be reported to customer support.

lcaErrNsBatchNoResult   **NS, #152**

Access to specified batch result is not possible, since it does not exist. This is an internal error, and should be reported to customer support.

lcaErrNsLicenseFeatureDisabled   **NS, #153**

Feature has not been enabled in the license.

lcaErrNsDemoLicenseDisallowed   **NS, #154**

Full license already exists. Demo license not allowed.

lcaErrNsVnodeIndexMismatch   **NS, #155**

Virtual node index mismatch

lcaErrNsNiVniOpenFailure   **NS, #156**

Unable to open NI layer or VNI

lcaErrNsVniMsgError   **NS, #157**

VNI messaging error

lcaErrNsMaxDbOpened   **NS, #158**

Exceeded the maximum number of databases

lcaErrNsEventsNotInitialized   **NS, #159**

Event sub-system not initialized.

lcaErrNsMcsNotFound   **NS, #160**

Monitor set cannot be found.

lcaErrNsMcsDblimit   **NS, #161**

Max number of monitor points exceeded.

lcaErrNsMcpNotFound   **NS, #162**

Monitor point cannot be found.

lcaErrNsMcpDblimit   **NS, #163**

Max number of monitor points exceeded.

lcaErrNsNvmtInUse   **NS, #164**

The network variable or monitor point is in use and cannot be used for current service.

lcaErrNsDbConversionInProgress   **NS, #165**

The engine is in the process of converting a database. Internal

lcaErrNsIllegalMcpTarget   **NS, #166**

Monitoring of specified target is not allowed.

lcaErrNsCantModifyNvName   **NS, #167**

Can't modify network variable name.

lcaErrNsDbconvAccessFailure   **NS, #168**

Unable to read/write old or new record during conversion

lcaErrNsUpgradeDisable   **NS, #169**

Program interface upgrade has been disabled.

lcaErrNsUpgradeInfoNotFound   **NS, #170**

No upgrade change info available for node.

lcaErrNsLNSUpgradeRequired   **NS, #171**

LNS program ID changed. Requires upgrade or revert back to old interface.

lcaErrNsEngineNotInitialized   **NS, #172**

LNS/NSI engine initialization has not completed. This is an internal error, and should be reported to customer support.

lcaErrNsNiNoWinsockDll   **NS, #173**

LNS attempted to use IP without the winsock.dll file installed.

lcaErrNsNiCantOpenIpLink   **NS, #174**

An LNS network interface failed to open an IP connection.

lcaErrNsNiInvalidIpAddress   **NS, #175**

An LNS network interface uses an IP address not defined on this PC.

lcaErrNsNsiNotConfigured   **NS, #176**

NSI needs to be added or re-commissioned.

lcaErrNsNiDeviceOpenFailure   **NS, #177**

NI device driver open failure.

lcaErrNsBadLength   **NS, #178**

Invalid length of parameter, data, service, or message.

lcaErrNsBadName   **NS, #179**

Invalid name length, characters, or usage.

lcaErrNsBadDomain   **NS, #180**

Invalid domain length, index, or usage.

lcaErrNsInsufficientRtrsForMnc   **NS, #181**

No routers exist to complete logical path needed for monitoring and control.

lcaErrNsInsufficientRtrsForNsi   **NS, #182**

No routers exist to complete logical path needed for NSI and control. This exception will be thrown if you attempt to remove a router and the operation fails because it would break communication between the LNS Server and a remote Full client.

lcaErrNs96BitAuthNotSupported   **NS, #183**

This error will be generated if you attempt to use 96-bit authentication on a device that does not support it.

lcaErrNsNeuronModelMismatch   **NS, #184**

Neuron model version mismatch. This error will be generated if you attempt to use the Load method to load an application image into a device that is incompatible with the device's system image. In this case, you need to upgrade the device's system image. Some devices support the LoadEx method, which automatically upgrades the system image before loading the application image if there are compatibility problems.

lcaErrLNSysimageFileFormatError   **NS, #185**

System image or upgrade image file format is invalid. Verify that LNS has been installed correctly.

lcaErrLNSysimageCannotBeWritten   **NS, #186**

This error will be generated if you invoke the LoadEx method on a device, but the system image cannot be written, probably because it is not stored in flash memory. The old system image will remain intact if this error is thrown. However, the device will remain applicationless. You should load a new application image into the device with the Load or LoadEx methods in this case.

lcaErrLNSysimageUpgradeMemoryFailur   **NS, #187**

This error will be generated if there is a failure to write the new system image the device after the LoadEx method has been invoked. These failures usually occur because the device does not

have flash memory in the required location to hold the new image during the download process. The old system image will remain intact if this error is thrown. However, the device will remain applicationless. You should load a new application image into the device with the Load or LoadEx methods in this case.

lcaErrLNSysimageUpgradeFailed   **NS, #188**

This error will be generated when you invoke the LoadEx method on a device, and the new system image is successfully transferred to the device, but the switch from an old system image to a new system image fails. If you encounter this error, try invoking the LoadEx method again, or loading the previous system image back into the device.

lcaErrNs16BitUserErrorCode   **NS, #189**

The user defined error code is too big. This is an internal error, and should be reported to customer support.

lcaErrNs16BitWarningCode   **NS, #190**

The LNS warning code is too big. This is an internal error, and should be reported to customer support.

lcaErrNs16BitErrorCode   **NS, #191**

The LNS error code is too big. This is an internal error, and should be reported to customer support.

lcaErrNsFlexDomainAuthNotSupported   **NS, #256**

An attempt was made to leave a device authenticated but without a domain, and that device does not support flex domain authentication.

lcaErrNsCantModifyBoundNvType   **NS, #257**

Cannot modify the network variable type of a network variable when it is bound.

lcaErrNsCantModifyMonitoredNvType   **NS, #258**

Cannot modify network variable type of a network variable when it is being monitored.

lcaErrNsRegCannotOpenReg   **NS, #259**

Cannot open the LNS registry.

lcaErrNsRegUnknown   **NS, #260**

Unknown LNS registry error.

lcaErrNsRegConfigFileNotFound   **NS, #261**

The LNS registry backup file cannot be found.

lcaErrNsRegCannotUpdateConfigFile   **NS, #262**

Cannot update the LNS registry backup file.

lcaErrNsUnsupportedFileDirectoryVer   **NS, #263**

The device contains a version of the file directory that is unsupported.

lcaErrNsBadsiHdrSize   **NS, #264**

Invalid self-identification data header size.

lcaErrNsBadsiCapacityExceedsLimits   **NS, #265**

Capacity defined by self-identification data exceeds supported limits.

lcaErrNsBadsiCountExceedsCapacity   **NS, #266**

Resource count exceeds capacity limits in self-identification data.

lcaErrNsLmsdObjectRange   **NS, #267**

Invalid object range in the LonMark portion of a network variable self-documentation string. When specifying a range of objects, the lowest object index must appear before the highest object index.

lcaErrNsLmsdObjectMemberNumber   **NS, #268**

Invalid object member in the LonMark portion of a network variable self-documentation string. The member number must be between 1 and 32767.

lcaErrNsLmsdObjectArray   **NS, #269**

The LonMark portion of a network variable self-documentation string indicates that elements of the network variable are to be used as members of an object array. However, the network variable array has fewer members than the object array.

lcaErrNsLmsdVersion   **NS, #270**

Unsupported LonMark version number in the node self-documentation string.

lcaErrNsLmsdExpectedObjHdr   **NS, #271**

Expected LonMark object header in node self-documentation string.

lcaErrNsLmsdObjectNameTooLong   **NS, #272**

The LonMark Object name in the node self-documentation string exceeds 16 characters.

lcaErrNsCpsdOwnerType   **NS, #273**

Invalid CP owner type in header portion of the CP self-documentation string. The header must indicate whether the CP is owned by the device, one or more objects, or one or more network variables.

lcaErrNsCpsdMismatchedRangeTypes   **NS, #274**

Mismatched CP range types in the CP self-documentation string. Both the low and high range values must have the same type.

lcaErrNsCpsdDisableWithoutNodeobj   **NS, #275**

The CP Self-documentation of one or more CPs indicate that the object must be disabled in order to set the CP, but no node object is defined. Without a node object, it is not possible to disable an object.

lcaErrNsCpsdFileCpWithoutFtp   **NS, #276**

A CP template file is defined in the external interface file for a device, but the device does not support FTP or direct memory access.

lcaErrNsCpsdCpRange   **NS, #277**

The CP Self-documentation includes a range of objects or network variables, with the first index in the range being greater than the second. CP ranges must be specified from lowest to highest.

lcaErrNsCpsdDistributedArraySize   **NS, #278**

The CP Self-documentation indicates that a CP array should be distributed among range of network variables or objects, but the CP array size does not match the owner's range.

lcaErrNsCpsdTemplateVersion   **NS, #279**

The version of the CP template is unsupported.

lcaErrNsGroupDblimit   **NS, #280**

The number of LonTalk groups has been exceeded. There is a limit of 255 LONtalk groups per domain. In some case you may form connections using subnet or domain broadcast, to limit the use of groups. You may also limit the use of groups by using network variable aliases. These options are specified in the connection description template used when forming connections.

lcaErrNsNodeidDblimit   **NS, #281**

Number of LonTalk node IDs exceeded. There are 127 LONtalk node Ids per subnet. LNS will allocate a new subnet, as needed, if the application does not specify the subnet to be used.

lcaErrNsLNSEngineInitTimeout   **NS, #282**

The application timed out starting the LNS engine process (LNSEng.Exe). This could be an indication that LNS is not installed properly. If you encounter this error, you should try rebooting your PC, or re-installing LNS.

lcaErrNsDmtNotFound   **NS, #283**

The dynamic message tag cannot be found.

lcaErrNsDmtDblimit   **NS, #284**

The maximum number of dynamic message tags has been exceeded. A single device can support up to 65,534 dynamic message tags, including internally defined message tags created for use with message points. You can add and remove message tags from a device with the Add and Remove methods of the MessageTags collection.

lcaErrNsLmobjNotDynamic   **NS, #285**

The specified LonMarkObject is not dynamic. This exception may be thrown if you attempt to assign a network variable to a static LonMarkObject with the AssignNetworkVariable method, unassign a network variable from a static LonMarkObject with the UnassignNetworkVariable method, or delete a static LonMarkObject with the Remove method.

You can determine if a LonMarkObject is dynamic or not by reading its IsDynamic property.

lcaErrNsLmobjNvNotDynamic   **NS, #286**

The specified network variable is not dynamic. This exception may be thrown if you attempt to assign a static network variable to a LonMarkObject with the AssignNetworkVariable method, unassign a static network variable from a LonMarkObject with the UnassignNetworkVariable method, or delete a static network variable with the Remove method.

You can determine if a network variable is dynamic or not by reading its IsDynamic property.

lcaErrNsLmobjInUse   **NS, #287**

The specified LonMarkObject is in use, and at least one of its member network variables is bound. This error may be thrown if you use the Remove or RemoveByIndex methods to delete a LonMarkObject with the **lcaLonMarkObjectRemoveNVs (1)** option set as the *removalFlags* element, and that LonMarkObject contains bound network variables.

The bound network variables must be disconnected before the LonMarkObject can be deleted. You can disconnect the network variables by specifying the **lcaLonMarkObjectRemoveAndDisconnectNVs (3)** option as the *removalFlags* element when you call the Remove or RemoveByIndex methods.

lcaErrNsApplReadWriteProtected   **NS, #288**

This exception will be thrown if you call the Load method to load a new application into a device, but the device's current application is read/write protected.

lcaErrNsEngineIsSuspended   **NS, #289**

The engine is currently suspended. This is an internal error, and should be reported to customer support.

lcaErrNsNotSupportedFromRemoteClient   **NS, #290**

The operation is not supported from a remote full client. This is an internal error, and should be reported to customer support.

lcaErrNsUnexpectedLink   **NS, #291**

The link between the parent and the child record was unexpected. This error usually indicates that the LNS database has been corrupted. If you encounter this error, you should use the Validate method to run a database validation, and consider switching to a backup database.

lcaErrNsUnsupportedProgramCapability   **NS, #292**

The program defines capabilities that are not supported by LNS. This error will be thrown when you call the Import method to import a device's XIF file, and the XIF file (or the device's self-documentation) indicates that it supports dynamic function blocks or dynamic function block members, but does not specify the EXTCAP_SUPPRESS_DYN_FB_DEF and EXTACP_SUPPRESS_DYN_FB_MBR_DEF flags. LNS 3.20 does not support downloading function blocks or function block membership to a device, any so any device that supports these operations needs to have an alternate way to define these dynamic function blocks and function block members. They must indicate that via the EXTCAP flags mentioned above.

lcaErrNsInvalidCpAttribute   **NS, #293**

The LONMARK organization does not allow modifiable device-specific configuration properties to be stored in files accessed via FTP on devices that support only sequential access. If you attempt to set the DeviceSpecificAttribute property to True on such a configuration property, this exception will be thrown.

lcaErrNsNotAllowedInASession   **NS, #294**

The operation is not allowed as part of a session. You can use sessions to group together properties and methods that must be performed together to avoid immediate error conditions. You can start a session with the BeginSession method, and close it with the EndSession method. You should note that the only network operations you can perform within sessions are those

related to changes in the physical topology of your network. These operations include moving devices and routers, adding and removing routers, and setting the class of routers. For more information, see the online help for the BeginSession method.

lcaErrNsInvalidCpFileSize   **NS, #295**

The configuration property value file size reported by the device does not agree with the size defined by the definitions in the configuration property template file. Note that if the device template was imported from an external interface file, the configuration property template file comes from the external interface file. Otherwise, the configuration property template file is loaded from the device.

lcaErrNsWarningFirst   **NS, #4030**

A symbol denoting the beginning of the error number range reserved for warning codes.

lcaErrNsUpdateCommError   **NS, #4030**

The database updates were successful, but one or more nodes were not updated because the LNS was unable to communicate with them. The LNS will continue to try to update the nodes in the background if the UpdateInterval property of the System object is set to a non-zero value, and you can force a retry with the RetryUpdates method.

You can keep track of which devices are up to date using commissioning events, and by reading the CommissionStatus property of each AppDevice object. If you are receiving persistent update failures on a device, you should re-commission the device with the Commission method.

lcaErrNsUpdateFuncError   **NS, #4031**

The database updates were successful, but one or more nodes were not updated because the node(s) rejected the update, for example due to an authentication failure. This usually means that there is a configuration mismatch between the node and the LNS's database. The LNS will continue to try to update the nodes in the background if the UpdateInterval property of the System object is set to a non-zero value, and you can force a retry with the RetryUpdates method.

You can keep track of which devices are up to date using commissioning events, and by reading the CommissionStatus property of each AppDevice object. If you are receiving persistent update failures on a device, you should re-commission the device with the Commission method.

lcaErrNsNeuronStateChangeFail   **NS, #4032**

The LNS's database updates were successful, but one or more nodes were not updated because the node(s) failed to make a necessary state transition. The LNS will continue to try to update the nodes in the background if the UpdateInterval property of the System object is set to a non-zero value, and you can force a retry with the RetryUpdates method.

You can keep track of which devices are up to date using commissioning events, and by reading the CommissionStatus property of each AppDevice object. If you are receiving persistent update failures on a device, you should re-commission the device with the Commission method.

lcaErrNsNodeUpdateDisallowed   NS, #4033

The database updates were successful, but one or more nodes could not be updated because they are in the wrong state. The LNS will continue to try to update the nodes in the background if the UpdateInterval property of the System object is set to a non-zero value, and you can force a retry with the RetryUpdates method. You can keep track of which devices are up to date using commissioning events and by reading the CommissionStatus property of each AppDevice object. If a node is applicationless, you must reload the application using the AppDevice.load method.

You can keep track of which devices are up to date using commissioning events, and by reading the CommissionStatus property of each AppDevice object. If you are receiving persistent update failures on a device, you should re-commission the device with the Commission method.

lcaErrNsUpdateFileXferError   NS, #4034

The database updates were successful, but an update error occurred. The LNS will continue to try to update the nodes in the background if the UpdateInterval property of the System object is set to a non-zero value, and you can force a retry with the RetryUpdates method. You can keep track of which devices are up to date using commissioning events and by reading the CommissionStatus property of each AppDevice object.

You can keep track of which devices are up to date using commissioning events, and by reading the CommissionStatus property of each AppDevice object. If you are receiving persistent update failures on a device, you should re-commission the device with the Commission method. Most file update errors map to this status during an update.

lcaErrNsUpdateWhenUncnfg   NS, #4035

The database updates were successful, but one or more nodes could not be updated because they are not configured. You should ensure that all applicable devices are configured when this error occurs. The LNS will continue to try to update the nodes in the background if the UpdateInterval property of the System object is set to a non-zero value, and you can force a retry with the RetryUpdates method.

You can keep track of which devices are up to date using commissioning events, and by reading the CommissionStatus property of each AppDevice object. If you are receiving persistent update failures on a device, you should re-commission the device with the Commission method.

lcaErrNsUpdateWhenAppless   NS, #4036

The database updates were successful, but one or more nodes could not be updated because they are applicationless. You can load a node's application by calling the Load method on the AppDevice object associated with the node. The LNS will continue to try to update the nodes in

the background if the UpdateInterval property of the System object is set to a non-zero value, and you can force a retry with the RetryUpdates method.

You can keep track of which devices are up to date using commissioning events, and by reading the CommissionStatus property of each AppDevice object. If you are receiving persistent update failures on a device, you should re-commission the device with the Commission method.

lcaErrNsUpdateCpFileTooShort   NS, #4037

The database updates were successful, but one or more devices was not updated because a configuration property file in the device is shorter than specified in the database. This is usually due to an inconsistency between the device's configuration property template file and its data file. The configuration property template file should be the same for all devices of a given type. If you encounter this type of error, check to make sure that the device is using the appropriate external interface file. The LNS will continue to try to update the nodes in the background if the UpdateInterval property of the System object is set to a non-zero value, and you can force a retry with the RetryUpdates method.

You can keep track of which devices are up to date using commissioning events, and by reading the CommissionStatus property of each AppDevice object. If you are receiving persistent update failures on a device, you should re-commission the device with the Commission method.

lcaErrNsUpdateCpFailure   NS, #4038

The database updates were successful, but the configuration properties on one or more devices could not be updated as a result of invalid configuration property definitions. The LNS will continue to try to update the nodes in the background if the UpdateInterval property of the System object is set to a non-zero value, and you can force a retry with the RetryUpdates method.

You can keep track of which devices are up to date using commissioning events, and by reading the CommissionStatus property of each AppDevice object. If you are receiving persistent update failures on a device, you should re-commission the device with the Commission method.

lcaErrNsUpdatesDeferred   NS, #4039

The database updates were successful, but the configuration properties on one or more devices could not be updated because the MgmtMode is currently set to **lcaMgmtModeDeferConfigUpdates (1)**. You can update all the devices on the network with all pending updates by setting the MgmtMode property to **lcaMgmtModePropagateConfigUpdates (0)**.

You can also update a single device with its pending device-only configuration changes while the MgmtMode property is still set to **lcaMgmtModeDeferConfigUpdates (1)** with the PropagateDeviceConfigUpdates method.

lcaErrNsWarningLast   NS, #4089

A symbol denoting the end of the error number range reserved for warning codes.

Network Interface Errors

These errors are returned by the LNS Network Interface. These errors will appear in the format NI: #<Error Number>.

lcaErrNiNoDevice   **NI, #1**

No network device is present.

lcaErrNiDriverNotOpen   **NI, #2**

The network driver could not be opened.

lcaErrNiDriverNotInit   **NI, #3**

The network driver required initialization

lcaErrNiDriverNotReset   **NI, #4**

The network driver required a reset.

lcaErrNiDriverError   **NI, #5**

An error occurred while communicating with the network driver.

lcaErrNiNoResponses   **NI, #6**

No message was received during the wait time.

lcaErrNiResetFails   **NI, #7**

The reset completion code was not received.

lcaErrNiTimeout   **NI, #8**

Could not communicate with the network interface. A message timed out.

lcaErrNiUplinkCmd   **NI, #9**

An uplink command was received from the LNS instead of a message.

lcaErrNiInternalErr   **NI, #10**

An internal error occurred, probably an invalid completion code.

lcaErrNiFileOpenErr   **NI, #11**

The log file could not be created or opened.

lcaErrNiNotLNS   NI, #12

The network interface received an LNS command, when no LNS was present.

lcaErrNiInvalidData   NI, #13

The parameter or return data was invalid.

lcaErrNiMsgRouted   NI, #14

The message was read, but then processed by a message callback function.

lcaErrNiResourceProblem   NI, #15

The required resources were not available to complete the function.

lcaErrNiInvalidNsi   NI, #16

The attached network interface does not contain NSI firmware.

lcaErrNiBadMessage   NI, #17

Poorly formed message.

lcaErrNiVniSendMsgFailure   NI, #18

The VNI had a message send failure.

lcaErrNiVniSendRespFailure   NI, #19

The VNI failed to send a response.

lcaErrNiVniReadFailure   NI, #20

The VNI failed to read a message.

lcaErrNiVniLostRefId   NI, #21

The VNI lost a network interface reference message ID.

lcaErrNiVniNotOpen   NI, #22

The LonTalk stack has not been opened.

lcaErrNiVniInitFailure   NI, #23

Could not initialize the LonTalk stack.

This is a network interface failure.

For USB U1/U20:

Open the LonWorks Interfaces Applet (Windows control panel), and see if you have three tabs (one for USB). If not then it is possible that this got changed with a new software update. Run repair on OpenLDV (in Windows control panel "Add or Remove Programs") or install OpenLDV 3.4 or later from the Echelon download website.

For all other network interfaces

See KB403**lcaErrNiObsolete**   **NI, #24**

This is an internal error and should be reported to customer support.

lcaErrNiBadRefId   **NI, #25**

Bad network interface reference message ID.

lcaErrNiImplicitAddrDisallowed   **NI, #26**

May not send implicit tags via NI layer. Must use a message monitor point.

lcaErrNiUnimplemented   **NI, #27**

This is an internal error and should be reported to customer support.

lcaErrNiBufferSize   **NI, #28**

The message length is too long.

lcaErrNiNoMessages   **NI, #29**

Message not found.

lcaErrNiNegativeResponse   **NI, #30**

Negative response.

lcaErrNiStateError   **NI, #31**

Unable to change the state of the network interface.

See KB409

caErrNiReadBusy   **NI, #32**

Previous response messages have not been read.

lcaErrNiMaxInterfacesOpen   NI, #33

There are no network interface resources available.

lcaErrNiInvalidNiHandle   NI, #34

Invalid network interface resource.

lcaErrNiOpenNiDeviceMismatch   NI, #35

The network interface has already been opened with a different network driver. This error will be generated if you open a network that is already opened using a different network interface, LNS type, or database directory

lcaErrNiNoWinsockDll   NI, #36

LNS attempted to use IP without the winsock.dll file installed.

lcaErrNiCantOpenIpLink   NI, #37

An LNS network interface failed to open an IP connection.

lcaErrNiInvalidIpAddress   NI, #38

An LNS network interface uses an IP address not defined on this PC.

lcaErrNiDeviceOpenFailure   NI, #39

 Device driver open failure

NI, #39 is a network interface error. It indicates that the network interface driver failed to open the network interface. For the i.LON 10, i.LON 100 or i.LON SmartServer, this error occurs when the LonWorks Interfaces driver configuration on your computer does not match the network interface settings in your i.LON 10, i.LON 100, or i.LON SmartServer.

To correct the error, open the LonWorks Interfaces Control Panel program and verify that the port setting, interface authentication key, hostname, and DNS suffix match the settings in your i.LON. To open LonWorks Interfaces, click Start, click Control Panel, double-click LonWorks Interfaces, and then click the RNI tab.

Make sure the Interface property is set to "Other". If the Interface property is set to "LonScanner" then change it to "Other" and try using the RNI again. The LonScanner Interface can only be used with the LonScanner program.

Please reference the i.LON 10, i.LON 100, or i.LON SmartServer user guides or online help for instructions in modifying the configuration of an RNI.

When opening your network, make sure you select the correct network interface. For an i.LON 10, .LON 100, i.LON SmartServer RNI, the name that you select in the LonMaker tool must match the name that you see on the LonWorks Interfaces RNI tab.

lcaErrNiNoMoreLeft   NI, #40

Signal iteration complete

lcaErrNiMipInUse   NI, #41

If using a regular (i.e. not high-performance) LNS network interface, only one network may be opened at a time. This error will be returned if you attempt to open a second network.

Connection Errors

Connection errors are errors returned when attempting to make network variable or message tag connections. These errors will appear in the format CONNERR: #<Error Number>.

lcaErrConnInvalidTargetCount   CONNERR, #1

An incorrect number of targets were specified. You must specify at least one and no more than 25 targets when adding members. To add more members, you must invoke the Connect method multiple times.

lcaErrConnPropertyNotFound   CONNERR, #2

There is no ConnectDescTemplate object with the specified handle.

lcaErrConnConflictingProperties   CONNERR, #3

One or more of the fields of the connection descriptions used by intersecting connections are incompatible with one another. This may include any of the fields of a connection description, including the service type, the use of priority, and the use of authentication. Several of the fields of a connection description may cause additional exceptions to be thrown if those fields conflict. Those error codes start at CONNERR:#28 lcaErrConnConflictingPrioritySettings and end with CONNERR:#32 lcaErrConnConflictingBroadcastSettings. Note that these error codes (and the CONNERR:#3 lcaErrConnConflictingProperties exception) do not apply to monitor connections.

lcaErrConnNodeNotFound   CONNERR, #4

An invalid device handle was specified for the hub or for one of the targets.

Each device in the network has a unique handle that allows a client to identify the device. Once assigned, the handle is permanently associated with the device until that device is removed.

lcaErrConnNvmtOutOfRange   CONNERR, #5

The network variable or message tag index is larger than the maximum declared for that node.

The index is the number used to identify a message tag or a network variable. The Neuron C compiler assigns indices to message tags and network variables in the order in which they are declared. The first network variable declared in the code is assigned index 0, the second is assigned index 1, and so on. The same applies to the message tags.

lcaErrConnCantConnectNvToMt   **CONNERR, #6**

You can only connect network variables to network variables and message tags to message tags.

If you attempt to connect a message tag to a network variable, or vice versa, you will generate this error message.

lcaErrConnTypesDoNotMatch   **CONNERR, #7**

The network variables are not all of the same type.

If SNVTs are used for the connection then, all members must be of the same type.

lcaErrConnTypesDifferentSizes   **CONNERR, #8**

The network variables are not all of the same size.

If the connection members are not SNVTs, then all members must have the same length.

lcaErrConnDirectionOrPollingMismatch   **CONNERR, #9**

An input network variable can only be connected to an output network variable, a polled output network variable can only be connected to a polling input network variable, and the implicit message tag *msg_in* can only be connected to an explicitly declared message tag.

This error will occur if an attempt was made to connect network variables of the same direction, network variables with mismatched polling attributes (a polled output network variable to non-polling input network variable), or msg_in to msg_in.

lcaErrConnMtagCantUseTurnaround   **CONNERR, #10**

An attempt was made to connect message tags in the same node (i.e., to create a turnaround message tag connection, which is not possible).

lcaErrConnPriNotConfigurable   **CONNERR, #11**

An attempt was made to modify the priority attribute; that attribute was not specified as configurable.

For network variables, the *priority* keyword may be included as part of the connection information in the Neuron C variable declaration. The *config* and *nonconfig* keywords specify whether the priority designation can be modified with a network management tool. The default is *config*.

The *priority* keyword affects output or polled input network variables.

Please refer to the Neuron C Reference Guide for more information.

lcaErrConnAuthNotConfigurable   CONNERR, #12

An attempt was made to modify the authentication attribute and that attribute was not specified as configurable.

For network variables, the *authentication* keyword may be included as part of the connection information in the Neuron C variable declaration. Including the keyword *config* in the declaration allows the network management tool to change the authentication status of this network variable after a node has been installed. Setting *nonconfig* prevents the authentication status from ever being changed for this network variable. The default is *config*. Please refer to the Neuron C Reference Guide for more information.

lcaErrConnSvctypeNotConfigurable   CONNERR, #13

An attempt was made to modify the protocol service type attribute, and that attribute was not specified as configurable.

For network variables, the service type can be specified in the connection information in the Neuron C variable declaration. The options are unackd, nackd_rpt and ackd. The keyword *config* is the default. This allows a network management tool to change the service specification at installation time. The keyword *nonconfig* indicates that the service type cannot be changed by a network management tool. Please refer to the *Neuron C Reference Guide* for more information.

lcaErrConnNvSelectorConstraint   CONNERR, #14

At most, one input network variable in a device, and in some situations only one output network variable in a device, can use the same network variable selector, which is shared by connections that intersect each other. This error will be generated when an attempt is made to form a connection that violates this constraint, and no network variable alias was available to overcome the constraint. If network variable aliases are available on the devices in the connection, they will normally be automatically used to avoid this error.

A selector is a number assigned to the network variable during the connection process in order to identify connected network variables. All members of the same connection, and any intersecting connections, share this selector. Without aliases, if an attempt is made to connect several inputs or several outputs (one or more of which polled by an input) on the same node in the same connection (or intersecting connections), this error will be generated. The use of aliases allows a single connection of several intersecting connections to have several different selectors (one per alias). If aliases are available, this error will normally not occur.

lcaErrConnNoMoreAddressSlots   CONNERR, #15

This error will occur when a device involved in an attempted connection does not have an available address table slot to complete the connection. This can also occur when multiple connections are sharing an address table slot, and one of them changes. As a result, they can no longer share address table slots.

In some cases broadcast addressing can be used to reduce address table usage, because output connections to different nodes may be able to use the same broadcast address table entry. In addition, if the connection is a multicast connection, the use of broadcast addressing will not require a group address table entry on the target device. The use of broadcast messaging is

controlled by the BroadcastOptions property of the ConnectDescTemplate object being used by the connection.

In some cases, aliases may be used to shift address table entry usage from one device to another. Multicast messages using a group use one address table entry on the source device, and one on each target device (note that this address table entry may be used for other connections as well). Using aliases to support multicast will result in using an address table entry for each target on the source node, but require no address table entries on the target node. The use of aliases for multicast is controlled by the AliasOptions property of the ConnectDescTemplate object.

lcaErrConnCannotCreateGroup   CONNERR, #16

This error will occur when the requested connection requires a group, but there are no more groups available in the domain.

A domain can contain 256 groups. Groups and group membership are assigned when connections are created. If there are several targets within a connection, a group will normally be formed. The use of aliases may prevent a group from being used, and broadcast addressing can be used as an alternative to groups. Use of aliases for multicast and broadcast can be controlled via the AliasOptions and BroadcastOptions properties of the ConnectDescTemplate object being used by the connection.

Groups can be reused for multiple connections within the same domain. Connection timers used by the different connections must be compatible, and the collection of nodes must either be identical in all the connections, or all the connections must at least have the same set of destination nodes (the sources can be different). This last case is called group overloading. Reusing groups is done automatically with the LNS Object Server.

lcaErrConnNoMoreNetvarSelectors   CONNERR, #17

An attempted connection failed because all network variable selector values were in use.

A network can use up to 12288 selector values. Unrelated connections will normally require unique selector values. Intersecting connections automatically use the same selector Network variable aliases are used automatically to resolve selector conflicts on a node (i.e. to eliminate this exception from occurring), although this uses up one selector for each alias. Starting in LNS 2.0, selector values are grouped into categories, with one category being "shareable". Non-intersecting, point-to-point connections may reuse selector values from this category any number of times. This is done automatically by the LNS Object Server.

lcaErrConnMtagInConnAlready   CONNERR, #18

With the exception of the special message tag msg_in, a message tag may only appear in one connection. This error will occur when a connection was attempted that violated this constraint.

lcaErrConnAuthenticationMismatch   CONNERR, #19

This error will occur when not all members of an attempted connection have the same setting for authentication.

It is possible to connect an authenticated output to a non-authenticated input, but an authenticated input must be connected to an authenticated output. A polling input must be authenticated if any of the targeted outputs are authenticated.

lcaErrConnGrpTooLargeForAckd   CONNERR, #20

This error will occur when an attempt is made to form a connection of more than 64 members that uses acknowledged service. Only unacknowledged or unacknowledged/repeat service is allowed for connections with more than 64 members. Such a connection must be built using a connection description that specifies one of these service types.

This error may also occur when polled output network variables or polling input network variables are involved in the connection.

lcaErrConnUnackdWithAuth   CONNERR, #21

An attempt was made to form a connection using authentication but not using the acknowledged service type.

The authentication process is composed of four messages, 2 of which are using the Acknowledged service. This is why using the authentication service without the Acknowledged service is illegal.

lcaErrConnAuthOnUnauthNode   CONNERR, #22

An attempt was made to form an authenticated connection with an unauthenticated node. Only nodes with the AuthenticationEnabled set to True are permitted to participate in authenticated connections.

lcaErrConnBcastSubnetsDiffer   CONNERR, #23

An attempt was made to subnet broadcast, but the devices are not all in the same subnet. This error is not currently generated, since the broadcast type used is determined automatically.

lcaErrConnNewBcastCausesNvLeak   CONNERR, #24

An attempt was made to use broadcast addressing, but a network variable leak was detected on another node. A network variable leak means that update messages for the network variable may be received by connections and devices that it is not intended to. LNS attempts to avoid this problem by the appropriate allocation of network variable selectors. However, some connection intersections make the use of broadcast addressing impossible, and so you may need to avoid using broadcast addressing in some situations. One solution for this problem is to use aliases for unicast connections, instead of using multicast connections. See the *Optimizing Connection Resources* section in Chapter 8 of the *LNS Programmer's Guides* for more information on this.

lcaErrConnExistingBcastCausesNvLeak   CONNERR, #25

An attempt was made to create a connection (of any type), but a detected network variable leak was caused by the use of broadcast addressing on another device. A network variable leak

means that update messages for the network variable may be received by connections and devices that it is not intended to. LNS attempts to avoid this problem by the appropriate allocation of network variable selectors. However, some connection intersections are not possible when broadcast addressing is in use. One solution for this problem is to use aliases for unicast connections, instead of using multicast connections. See the *Optimizing Connection Resources* section in Chapter 8 of the *LNS Programmer's Guides* for more information on this.

lcaErrConnBcastCantAck   **CONNERR, #26**

Acknowledged services cannot be used with broadcast addressing.

Note that subnet broadcast is not allowed with ACKD service, and domain broadcast is not allowed with either ACKD or UNACKD_RPT services.

lcaErrConnDomainBcastMustUseUnackd   **CONNERR, #27**

Domain broadcast addressing is restricted to the unacknowledged service.

Please note that UNACKD_RPT service is not allowed with Domain broadcast.

lcaErrConnConflictingPrioritySettings   **CONNERR, #28**

The connection descriptions for intersecting connections have incompatible settings for priority. The priority option is available only for network variables. If an output network variable that is not declared as polled is a member of multiple connections, the priority settings for all of those connections must be identical. If a polling input network variable is a member of multiple connections, the priority settings for those connections must also be identical.

lcaErrConnConflictingAuthSettings   **CONNERR, #29**

The authentication settings for the network variables in a connection are incompatible. If a target network variable (an input network variable, or polled output network variable) in a connection is using authenticated messaging, then all network variables involved in the connection must also use authenticated messaging.

lcaErrConnConflictingSvctypeSettings   **CONNERR, #30**

The connection descriptions for intersecting connections have incompatible settings for service type. If an output network variable that is not declared as polled is a member of multiple connections, the service type settings for all of those connections must be identical.

lcaErrConnConflictingTimerSettings   **CONNERR, #31**

The connection descriptions for intersecting connections have incompatible settings for one of the timer values. If an output network variable that is not declared as polled is a member of multiple connections, the timer settings for all of those connections must be identical. If a polling input network variable is a member of multiple connections, the timer settings for those connections must also be identical.

lcaErrConnConflictingBroadcastSettings   CONNERR, #32

The connection descriptions for intersecting connections have incompatible settings for broadcast addressing. If an output network variable that is not declared as polled is a member of multiple connections, the broadcast options for all of those connections must be identical. If a polling input network variable is a member of multiple connections, the broadcast options for those connections must also be identical.

lcaErrConnCantDirectlyManageMonitorConn   CONNERR, #33

Monitor connections cannot be directly managed. This error indicates that the application is attempting to directly manage internal monitor connections, rather than implicitly manipulating those connections by using monitor points.

lcaErrConnConflictingMonitorProperties   CONNERR, #34

The connection descriptions used for intersecting monitoring connections have conflicting settings. If the connection descriptions used by intersecting connections conflict, and neither connection is a monitoring connection, then any of the following errors may be thrown:

CONNERR:#3 lcaErrConnConflictingProperties

CONNERR:#28 lcaErrConnConflictingPrioritySettings

CONNERR:#29 lcaErrConnConflictingAuthSettings

CONNERR:#30 lcaErrConnConflictingSvctypeSettings

CONNERR:#31 lcaErrConnConflictingTimerSettings

CONNERR:#32 lcaErrConnConflictingBroadcastSettings

Object Server Errors

These errors are returned by the Object Server component. These errors will appear in the format LCA: #<Error Number>.

lcaErrStringInvalidSize   **LNS, #1**

A string passed did not match the required size, or exceeded the maximum size.

lcaErrInvalidType   **LNS, #2**

An argument was passed with the wrong data type, or an assignment to a property has been made using an inappropriate data type.

lcaErrDuplicateKey   **LNS, #3**

This error indicates that the object with this key or name already exists. Names or Keys of items being explicitly added to a collection object must be unique. It should be noted that this restrictions does not apply to implicitly added items.

lcaErrInvalidOleObject   **LNS, #4**

An invalid OLE object has been passed to the object server by making assignments to LNS properties or calls to LNS methods.

lcaErrUninitializedDb   **LNS, #5**

Contact technical support.

lcaErrObjectNotFound   **LNS, #6**

An object has been requested by name or index that could not be found in the databases. Typically, this would be a request from a LNS collection object with a bad name or index.

It should be noted that this error can also occur in some implicit assignments. For example, if the network interface to be used has not been explicitly assigned to the appropriate property, LNS will continue using the network interface previously assigned to the ActiveRemoteNI or NetworkInterface property, respectively.

In such a scenario, this exception could occur even though no explicit assignment has been made, indicating that an implicit assignment maps to an unavailable object.

lcaErrCannotCreateObject   **LNS, #7**

The Object Server was unable to created the object.

lcaErrInternal   **LNS, #8**

An unexpected error occurred. Please contact technical support on LonSupport@Echelon.com

lcaErrFeatureNotAvailible   **LNS, #9**

A feature has been accessed which is not available in the current operating mode. Most likely, this error message is caused by an attempt to access properties that are not available to remote clients.

lcaErrFileOpen   **LNS, #10**

The Object Server is attempting to open a file that is already opened by another application.

lcaErrBadApbFile   **LNS, #11**

The file containing the binary version of the downloadable application image has been corrupted. The device manufacturer should be contacted for providing a valid APB file, of said file should be re-built using the NEURON C compiler and linker.

Alternatively, the APB file can be re-build using the NXE32BIN.EXE utility, which itself requires the NXE and XFB files as input data.

lcaErrBadOrMissingXif   **LNS, #12**

The file containing the external interface information of the device has been corrupted, or is missing. The device manufacturer should be contacted to provide a valid XIF file, or the file should be re-built using the Neuron C compiler and linker.

Alternatively, the device could be installed using an "Ad-Hoc" installation scenario, in which XIF files are not required. This requires the device to be available, online, functioning, and not having the self-documentation features disabled.

lcaErrBadOrMissingXfb   **LNS, #13**

The file containing the binary version of the external interface file has been corrupted, or is missing. The device manufacturer should be contacted for providing a valid APB file, or the file should be re-built using the NEURON C compiler and linker.

Alternatively, the XFB file might be re-build from the XIF file, using the XIF32BIN.EXE utility.

lcaErrInvalidPgmType   **LNS, #14**

The ProgramType property contains an invalid value.

lcaErrInvalidCollectionIndex   **LNS, #15**

A query to an LNS collection object has been made using an invalid index number. The range of valid indices starts with 1 and ends with the current item count.

lcaErrCannotUpdateUnattachedObject   **LNS, #16**

An attempt was made to update a device that is not attached to the network.

lcaErrCannotUpdateWhenOffnet   LNS, #17

An action has been requested that can't be performed while the network management mode is set to the **lcaMgmtModeDeferConfigUpdates (1)**. Change the MgmtMode property of the System object to **lcaMgmtModePropagateConfigUpdates (0)** to allow this action to be performed.

lcaErrCannorRemoveReservedSubsystem   LNS, #18

An attempt has been made to remove a reserved Subsystem object. Such Subsystem objects are all subsystem objects in the "ALL" and "Discovered" subsystems. Other read-only subsystems collection objects are the Router object's Subsystems property and the AppDevice object's Subsystems property.

lcaErrCannotAddToReservedSubsystem   LNS, #19

An attempt has been made to add a subsystem object to a reserved subsystem area. Such areas are all subsystem objects in the "ALL" and "Discovered" subsystems. Other read-only subsystems collection objects are the Router object's Subsystems property and the AppDevice object's Subsystems property.

lcaErrInvalidPath   LNS, #20

An invalid subsystem path has been specified.

lcaErrDatabaseExists   LNS, #21

An attempt has been made to create a database that already exists. The database path should be changed to point to a different location.

lcaErrDatabaseNoAccess   LNS, #22

Access to the database files has been denied by the operating system. When using operating systems with file security, access permissions should be granted to applications with the appropriate system tools.

lcaErrDatabasePathTooLong   LNS, #23

The database path passed to the Networks collection's Add method is too long. The path to a network database must not exceed 26 characters.

lcaErrMaxClientsExceeded   LNS, #24

The LNS maximum of 254 simultaneous clients has been exceeded.

lcaErrDatabase   LNS, #25

An unexpected database error occurred. Please take a note of the full error message, and contact technical support at LonSupport@Echelon.com

LcaErrInvalidDeviceTemplate   **LNS, #26**

The DeviceTemplate object has been corrupted.

lcaErrNoDatabase   **LNS, #27**

The network database has not been found in the location previously registered.

lcaErrNoDbDictionary   **LNS, #28**

The LNS dictionary files _objects.dat and _objects.idx are missing, or they can not be found. The default location is C:\LONWORKS\ObjectServer\Dictionary.

lcaErrInvalidSystemObject   **LNS, #29**

The System object associated with this network database is invalid. To resolve this, the network database must be rebuilt from scratch.

lcaErrNetworkDbNotOpen   **LNS, #30**

A service has been requested prior to invoking the Network object's Open method.

lcaErrCannotRemoveSubsystemWithDevices   **LNS, #31**

An attempt to remove a subsystem that still contains AppDevice or Router objects has been made. These objects must be removed first.

lcaErrObjectLocked   **LNS, #32**

An attempt has been made to modify an object that has been internally locked by LNS.

lcaErrCantCreateDbServer   **LNS, #33**

This error is returned by a remote client when the host is unable to launch the LNS Server.

lcaErrUnsupportedEvent   **LNS, #34**

Unsupported variant type.

lcaErrGeneric   **LNS, #35**

Contact customer support.

lcaErrDbServer   **LNS, #36**

An LNS Server error. Contact customer support.

lcaErrCommissionCausesReplace   **LNS, #37**

An attempt has been made to commission a device after changing the device's Neuron ID. In this situation, the Replace method should be used.

lcaErrAppdeviceRequired   **LNS, #38**

The Object Server was unable to communicate with the Appdevice.

lcaErrWrongWriteDataSize   **LNS, #39**

Contact customer support.

lcaErrInvalidFormat   **LNS, #40**

Contact customer support.

lcaErrCompiler   **LNS, #41**

Generic compiler error. See the LNS Field Compiler documentation for details.

lcaErrProjectMgrNotAvailable   **LNS, #42**

The LNS Field Compiler's project manager and dependency checker is unavailable. See the LNS Field Compiler documentation for details.

lcaErrInvalidLinkerOptions   **LNS, #43**

The options passed to the NEURON Linker were invalid. See the LNS Field Compiler documentation for details.

lcaErrInvalidBuildTemplate   **LNS, #44**

The BuildTemplate object is invalid, as it does not comply with the requirements and constraints of the LNS Field Compiler. See the LNS Field Compiler documentation for details.

lcaErrInvalidHardwareTemplate   **LNS, #45**

The HardwareTemplate object is invalid, as it does not comply with the requirements and constraints of the LNS Field Compiler. See the LNS Field Compiler documentation for details.

lcaErrInvalidProgramTemplate   **LNS, #46**

The ProgramTemplate object is invalid, as it does not comply with the requirements and constraints of the LNS Field Compiler. See the LNS Field Compiler documentation for details.

lcaErrInvalidExportOption   **LNS, #47**

The options passed to the Neuron Exporter were invalid. See the LNS Field Compiler documentation for details.

lcaErrCantCallDtssComponent   LNS, #48

An attempt to communicate with a LNS Device Control failed.

lcaErrCantLoadDevice   LNS, #49

The application could not be loaded into the device.

lcaErrCantRemoveFromReservedSubsystem   LNS, #50

An attempt has been made to remove an item from a reserved subsystem. Reserved subsystems are all subsystem objects in the "ALL" and "Discovered" subsystems.

Other read-only subsystems collection objects are the Router object's Subsystems property and the AppDevice object's Subsystems property.

lcaErrDeviceTemplatePgmMismatch   LNS, #51

The program ID from the DeviceTemplate object does not match the one found on the device during an attempt to commission the device.

lcaErrMustSetNeuronIdFirst   LNS, #52

The Neuron ID of an object must be set prior to querying this property or invoking the method. This error will occur if you invoke the Test method on an AppDevice or Router object before setting the object's Neuron ID.

lcaErrInvalidOnDiscoveredDevice   LNS, #53

Operation is invalid on a discovered device. You can only perform the requested operation on a device that resides in a user subsystem (i.e. not in one of the discovered subsystems).

lcaErrCantCopyLNSDirectory   LNS, #54

Internal error. Contact customer support.

lcaErrCantRecoverNotNewDb   LNS, #55

A call to the RecoverFromNetwork method has been made, using a previously used network database. For the recovery process, a newly created network database is required.

lcaErrCantRecoverMissingLNSDbFiles   LNS, #56

Some parts of the network database are lost and can not be recovered. The network must be re-build from scratch, if no database backup is available.

lcaErrCompilerInit   **LNS, #57**

Initialization of the NEURON C Compiler failed. See the LNS Field Compiler documentation for details.

lcaErrAssembler   **LNS, #58**

Generic error message, indicating an unspecified NEURON Assembler error. See the LNS Field Compiler documentation for details.

lcaErrAssemblerInit   **LNS, #59**

Initialization of the NEURON Assembler failed. See the LNS Field Compiler documentation for details.

lcaErrLinker   **LNS, #60**

Generic error message, indicating an unspecified NEURON Linker error. See the LNS Field Compiler documentation for details.

lcaErrLinkerInit   **LNS, #61**

Initialization of the NEURON Linker failed. See the LNS Field Compiler documentation for details.

lcaErrExport   **LNS, #62**

Generic error message, indicating an unspecified NEURON Exporter error. See the LNS Field Compiler documentation for details.

lcaErrExportInit   **LNS, #63**

Initialization of the NEURON Exporter failed. See the LNS Field Compiler documentation for details.

lcaErrGlobalDbNotOpen   **LNS, #64**

An attempt has been made to obtain a service from LNS prior to opening the LNS global database. This database can be opened and closed by opening and closing the ObjectServer object.

lcaErrOnlyAllowedOnHostIntf   **LNS, #65**

The requested operation is only allowed on an Interface object contained by a device that supports dynamic network variables, dynamic message tags, or dynamic LonMarkObjects.

lcaErrAnotherNetworkAlreadyOpen   **LNS, #66**

An attempt to open a network database failed because another network object is still held open by the same client process. This error should not occur in LNS 3 or greater. If it does, contact customer support.

lcaErrSystemNotOpen   LNS, #67

A service that requires the system to be open has been accessed prior to successfully opening the System object. There are many circumstances that could cause this. For example, if you call the Validate method to validate a network database before opening the system, or if you call the Backup method to copy a network database before opening the system, this exception will be thrown.

lcaErrDbConversionRequired   LNS, #68

The global database or local database is from an older version of LNS, and the appropriate CompactDb method must be called prior to opening the object. The CompactDb method will compact the database, reorganize its contents, and automatically upgrade to the latest version.

lcaErrPropertyCantBeSetWhenOpen   LNS, #69

An attempt has been made to change a property while the associated key object (the System, Network, or ObjectServer object) is open. This property requires the object to be closed prior to accepting changes.

lcaErrCantRecoverReadOnlyFile   LNS, #70

Contact customer support.

lcaErrStringIsEmpty   LNS, #71

Contact customer support.

lcaErrWrongServerDll   LNS, #72

This error may occur if the RemoteFlag property is set to False in an application that has been distributed as an LNS Remote redistributable. Other occurrences of this error should be reported to customer support.

lcaErrInvalidProgramType   LNS, #73

Contact customer support.

lcaErrNotAvailableOnRemoteClient   LNS, #74

The requested property of method is not available on the remote client.

lcaErrDuplicateApplicationName   LNS, #75

The value assigned to the ApplicationName property was invalid, as it already exists.

lcaErrCantRecoverNotAttachedToNetIntf   **LNS, #76**

Recovery fails due to a lack of an LNS network interface.

lcaErrDevTemplateWithThisProgramExists   **LNS, #77**

Implicit or explicit creation of a DeviceTemplate object failed due to the program ID already being in use with a different device template in the same network database.

The program ID is a unique identifier for each device type (i.e. device template object). Devices certified by the LonMark Interoperability Association will have a unique program id, using a globally unique manufacturer ID allocated by LonMark.

lcaErrCantCreateLcaServer   **LNS, #78**

The LNS Server cannot be created.

lcaErrCantLaunchLcaServer   **LNS, #79**

The LNS Server cannot be launched.

lcaErrNoMemory   **LNS, #80**

No more system memory available.

lcaErrDbInvalidVersion   **LNS, #81**

The database has an invalid version. Note that LNS databases can be upgraded using the CompactDb method, but also note that LNS databases are not backwards compatible. Once updated to version N+1, the database will not be accessible for a version N client any more.

lcaErrDictionaryInvalidVersion   **LNS, #82**

The LNS database dictionary files are using an invalid version.

lcaErrDbAndDictionaryMismatch   **LNS, #83**

The LNS dictionary files do not match the requirements of the global database and network database. After making a backup copy, try upgrading the databases using the CompactDb method. Note that LNS databases can be upgraded using the CompactDb method, but also note that LNS databases are not backwards compatible. Once updated to version N+1, the database will not be accessible for a version N client any more.

lcaErrCouldNotOpenNeuronCFile   **LNS, #84**

The NEURON C source code file could not be opened. It may be locked by another application, it may be corrupted, or the operating system's security mechanism might prevent LNS from accessing the file.

lcaErrCouldNotOpenLangResourceFile   LNS, #85

The language resource file, which is part of the LonMark device resource files, could not be opened. It may be locked by another application, it may be corrupted, or the operating system's security mechanism might prevent LNS from accessing the file.

lcaErrCouldNotOpenFuncProfileFile   LNS, #86

The functional profile template file, which is part of the LonMark device resource files, could not be opened. It may be locked by another application, it may be corrupted, or the operating system's security mechanism might prevent LNS from accessing the file.

lcaErrStringInvalidChar   LNS, #87

The string passed contains one of more characters that are invalid in the current environment. For example, file names must follow, style conventions taken from the Windows operating system on the PC running an application. When writing to the Name property of any object, use of the forward slash (/), back slash (\), period (.), and colon (:) characters will cause this exception to be thrown.

lcaErrAuthKeyNotSet   LNS, #88

The authentication key has not been set prior to accessing authenticated objects. Such an object can be any AppDevice or Router object, including the LNS Server itself. Note that prior to opening a remote system, a remote client needs to specify the authentication key.

lcaErrReadOnlyCollection   LNS, #89

An attempt to change a member of a read-only collection has been made. Such collection objects do not support adding or removing items at runtime. It should be noted that this exception may also occur with collections which are usually write-enabled, as the context might not allow for changes.

For example, the Subsystems collection would typically be seen in the hierarchy System::Subsystems context, to which further items can be added using the Subsystems collection's Add method. The AppDevice and Router object's Subsystems collection objects, however, represent all subsystem objects to which the device belongs. Such a collection would be a read-only collection in this particular context, and a write-enabled collection in another context.

lcaErrValueOutOfRange   LNS, #90

The value supplied to the property exceeds the range limit for that property.

lcaErrCantUpdateWhenPaused   LNS, #91

A requested service cannot be completed due to the DataServer being paused. DsPause should be set to FALSE to re-start the DataServer.

lcaErrCantCreateLcaMonitor   **LNS, #92**

The Lca Monitor, a utility component for internal purposes, could not be created. This failure is caused by a lack of memory available to LNS.

lcaErrCantLaunchLcaMonitor   **LNS, #93**

The Lca Monitor, a utility component for internal purposes, could not be launched. This utility, LCAMON.EXE, would typically be located in the C:\LONWORKS\Bin folder, which must be part of the search path or otherwise visible from the working directory of the calling tool. The 93 exception would indicate any problem launching the LCA Monitor utility tool like memory shortage, bad permissions, executable not found, etc.

lcaErrMustSetDsPauseToChange   **LNS, #94**

The DataServer must be paused prior to applying the requested changes. Use DsPause to pause the DataServer.

lcaErrNotAvailableOnLocalClient   **LNS, #95**

A service only available to remote clients has been requested from a local client.

lcaErrObjectDeleted   **LNS, #96**

The object has been deleted.

lcaErrInvalidRmcObject   **LNS, #97**

Invalid object received through remote procedure call. Contact customer support.

lcaErrInvalidRmcServer   **LNS, #98**

Contact customer support.

lcaErrInvalidRmcClient   **LNS, #99**

Contact customer support.

lcaErrInvalidRmcVersion   **LNS, #100**

The client and server are running different versions of LNS and the Object Server was unable to compensate.

lcaErrInvalidRmcMethod   **LNS, #101**

Contact customer support.

lcaErrNeuronIdInUse   **LNS, #102**

The Neuron ID assigned is already being used by another device in the network.

lcaErrDbCantBeCompactedWhenOpen   **LNS, #103**

CompactDb() methods must be called prior to opening the associated object.

lcaErrDbCantBeRemovedWhenInUse   **LNS, #104**

The object can not be removed as it still is in use.

lcaErrCantRemoveDefaultAccount   **LNS, #105**

The default Account object can not be removed.

lcaErrCantRemoveAccountWithCharges   **LNS, #106**

An account object needs to be empty when being removed.

lcaErrCantRemoveLNSNode   **LNS, #107**

The device can not be removed, as it is associated with the network interface currently in use.

lcaErrCantConnectToServer   **LNS, #108**

The remote client fails to connect to the LNS Server.

lcaErrTcpSocketError   **LNS, #109**

An unspecified socket error occurred, causing the IP connection to be lost.

lcaErrBadPermissionFormat   **LNS, #110**

Malformed data was assigned to the PermissionString property.

lcaErrCantModifyInSharedMode   **LNS, #111**

An attempt has been made to modify a shared DataServer property. Change the DsMode property to enter exclusive mode, or refrain from modifying the property.

lcaErrDataUnderrun   **LNS, #112**

An IP remote client did not receive expected data in time. The connection might be lost.

lcaErrDataOverrun   **LNS, #113**

An IP remote client received unexpected data. The configuration should be verified.

lcaErrIllegalInTransaction   LNS, #114

A service that can not be part of an explicit transaction has been requested within an explicit transaction. Complete or dismiss the transaction using the CancelTransaction method or CommitTransaction method prior to invoking this service.

lcaErrNoPermission   LNS, #115

Permission to an item has been requested and denied. Use the PermissionString property local to the server to change permission preferences.

lcaErrStaleObject   LNS, #116

An attempt was made to modify or use a stale LNS object.

For example, once the system object has been successfully closed, all objects accessed through the system object will be stale, and should be de-referenced. Or, if the Network Service Device has been replaced, all collection objects accessed before the replacement will be stale, and should be de-referenced.

lcaErrNotAvailableOnTcpClient   LNS, #117

A service which is only available local to the server or on a Full client has been requested from a Lightweight client.

lcaErrNotAnNsiHost   LNS, #118

The (hosted) device does not execute the NSI firmware, and as a result, the requested service can not be completed.

lcaErrInterfaceNotModifyable   LNS, #119

An attempt to dynamically modify a static interface was made. This error will occur if you attempt to use the Add method to define a dynamic LonMarkObject on a device that does not support dynamic function blocks.

To support dynamic interface modifications, you can implement the wink command extensions as documented in the LonMark Application Layer (7) Interoperability Guidelines, Version 3.1 or higher.

lcaErrNoConnectionToServer   LNS, #120

There is no communication with the LNS Server.

lcaErrInvalidInitString   LNS, #121

The AddNvFromString or InitFromString method was provided with an invalid string.

lcaErrReadOnlyInContext   **LNS, #122**

An attempt was made to write to, or delete, an object that is read-only. LNS contains multiple read-only objects that could cause this exception to be thrown. For example, if you use the Remove method to delete a pre-defined FormatLocale object, this exception will be thrown.

lcaErrInvalidPropChangeEvent   **LNS, #123**

Contact customer support.

lcaErrNotAllowedWhenIndependent   **LNS, #124**

The called service doesn't work in server-independent mode.

lcaErrNetworkNotOpen   **LNS, #125**

The network is not open.

lcaErrCantReadMsgPoint   **LNS, #126**

The object server is unable to read the message point.

lcaErrOpenCloseTypeMismatch   **LNS, #127**

If the Network object's Open method is used to open a network, the Close method must be used to close it. If the OpenIndependent method is used to open the network, the CloseIndependent method must be used to close it. This error may occur when these methods are not matched properly.

LcaErrInvalidDomainLength   **LNS, #128**

The domain is the wrong length.

lcaErrDatabaseRecoveryFailed   **LNS, #129**

The database recovery operation failed.

lcaErrOnlyAllowedOnLocalVNI   **LNS, #130**

The requested service is not available on a remote Full client.

lcaErrInvalidHexValue   **LNS, #131**

An invalid hexadecimal value was supplied.

lcaErrUniqueNVNameRequired   LNS, #132

A unique network variable name is required for this service. Some devices, such as the *i*.LON 100, require that all network variables within the device have a unique name. If you attempt to assign a duplicate name to a network variable on such a device, the operation will fail, and this exception will be thrown.

lcaErrInternalErrBadExceptionCode   LNS, #133

This is an internal error and should be reported to customer support.

lcaErrInternalErrBadHresultExceptionCode   LNS, #134

This is an internal error and should be reported to customer support.

lcaErrInternalErrBadVNIDataServer   LNS, #135

This is an internal error and should be reported to customer support.

lcaErrInternalErrBadNsdHandle   LNS, #136

This is an internal error and should be reported to customer support.

lcaErrInternalErrStartTransaction   LNS, #137

This is an internal error and should be reported to customer support.

lcaErrNotAllowedOnVniNetwork   LNS, #138

An attempt was made to call the OpenIndependent method on a Network object that was retrieved from the VNINetworks collection.

lcaErrInvalidAddressingMode   LNS, #139

When using the GetMessagePoint method, an invalid addressing mode was supplied.

lcaErrNotAllowedOnPermanentNsd   LNS, #140

An attempt was made to remove a permanent network service device.

lcaErrOnlyAllowedWhenIndependent   LNS, #141

The requested service can only be performed on networks opened with the OpenIndependent method.

lcaErrDatabaseDiskFullErr   LNS, #142

The disk containing the LNS database is full.

lcaErrNotAllowedWithoutNetworkInterface   LNS, #143

This operation is not allowed because the Object Server global flag lcaNoNetworkInterface is set.

lcaErrDbTcpIpNetworkError   LNS, #144

Please verify the TCP/IP stack is installed. You may need to add a dial-up connection to initialize the stack.

lcaErrXDriverBusy   LNS, #145

The profile selected is already in use by another OpenLDV xDriver listener application. Only one application may use an xDriver profile at any given time. For more information on xDriver profiles, see the *OpenLDV Programmer's Guide, xDriver Supplement.*

lcaErrXDriverCantConnect   LNS, #146

There was no response from the xDriver connection broker. Make sure that the xDriver connection broker service has been started. For more information on the xDriver connection broker, see the *OpenLDV Programmer's Guide, xDriver Supplement.*

lcaErrXDriverNotListener   LNS, #147

The current process is not registered for xDriver uplink events for this profile. For more information on xDriver profiles and uplink events, see the *OpenLDV Programmer's Guide, xDriver Supplement.*

lcaErrXDriverNoSuchProfile   LNS, #148

The xDriver profile selected does not exist, or the xDriver connection broker is not currently running. Make sure that the profile you are using exists, and that the xDriver connection broker service has been started if you encounter this error. For more information on the xDriver connection broker, see the *OpenLDV Programmer's Guide, xDriver Supplement.*

lcaErrInvalidExportFile   LNS, #149

LNS was unable to export the database validation report when the Export method was called. This may occur if the target directory supplied to the method as the *path* element cannot be opened for writing, or if the target directory already exists, and the *overwriteExisting* element (which indicates whether or not the exported report should overwrite the contents of the target directory) is set to False.

lcaErrNotYetSet   LNS, #150

This exception will be thrown if you attempt to read the RemoteChannel property, and the property has not yet been set. The RemoteChannel property must be explicitly set by a Full client application before it can be read by any other application.

lcaErrUniqueNameRequired   **LNS, #151**

You attempted to create an object with a name that is already being used by an object of that type. For example, the name assigned to all LonMarkObject objects must be unique. Therefore, if you use the Add method to create a LonMarkObject and specify a name that is already being used by another LonMarkObject, this exception will be thrown.

lcaErrInvalidServiceType   **LNS, #152**

The service type(s) specified for the operation is invalid. This exception will be thrown if you attempt to create a network variable that uses both the acknowledged and unacknowledged messaging services with the Add method.

lcaErrDbValidationCancelled   **LNS, #153**

If you use the CancelValidation method to cancel a database validation, this exception will be thrown to notify you that LNS has successfully stopped the database validation procedure.

lcaErrUnavailableResourceFiles   **LNS, #154**

LNS was unable to locate the resource file referenced by a TypeSpec object. This exception will be thrown if you call the Lookup method to verify that the program ID and scope of a TypeSpec object reference a valid resource file, and LNS is unable to locate the resource file. LNS may be unable to locate the resource file if the value of the program ID or scope property is entered incorrectly, or if the LdrfCatalogPath property does not correctly point to the location of the resource files.

lcaErrNotFoundInResourceFiles   **LNS, #155**

LNS unable to locate the format name referenced by a TypeSpec object. This exception may occur if you call the Lookup method to verify that the program ID, scope and format name of a TypeSpec object reference a valid format type. If LNS is able to locate the resource file referenced by the program ID and scope of the TypeSpec object, but does not find a definition of the specified format name in that file, this exception will be thrown. In this case, check to make sure that the value of the FormatName property was written correctly, and that the specified resource file contains a definition of that type.

lcaErrTypeLengthTooLong   **LNS, #156**

The maximum length of a network variable is not long enough to support the type assigned to the network variable. This exception will be thrown if you change the type assigned to a network variable by writing to the TypeSpec property, and the Length of the newly selected TypeSpec object exceeds the MaxLength of the network variable.

lcaErrTypeNotChangeable   **LNS, #157**

The network variable does not support changeable types. This exception will be thrown if you attempt to write to the TypeSpec property of a network variable that does not support changeable types. You can check if a network variable supports changeable types by reading its ChangeableTypeSupport property.

lcaErrOutOfRange   **LNS, #158**

A value was supplied to a property or method parameter that is outside the valid range for that property or parameter. For example, the TxTimer property accepts a range of encoded values from 0 to 15, and you can write the value 254 to the property to restore it to its default value. If you wrote any other value to the property, this exception would be thrown.

lcaErrUnableToCreateBackup   **LNS, #159**

LNS was unable to create a backup copy of the network database when the Backup method was called. This may occur if there was an error copying the network database, or if the *systemPath* element supplied when the method was called did not reference a valid system path.

lcaErr12HourTimeFormatNotSupported   **LNS, #160**

This exception will be thrown if you write a value to the ShortTimeFormat property that would cause LNS to display time values in 12-hour format. For more information on this, see the online help for the ShortTimeFormat property.

lcaErrNotAllowedOnTemporaryObject   **LNS, #161**

The requested operation can only be performed on permanent monitor sets and monitor points. For example, if you are adding a MsgMonitorPoint object to a temporary monitor set, and specify a MessageTag as the *targetDevice* element, this exception will be thrown. See the online help for the MsgMonitorPoints collection's Add method for more information on this.

Data Server Errors

These are errors returned by the LNS Data Server. The Data Server handles network variable and message tag monitoring and control. These errors will appear in the format DS: #<Error Number>.

lcaErrDsBusy   **DS, #1**

Could not get semaphore to perform Data Server operation.

lcaErrDsAlreadyStarted   **DS, #2**

Cannot start Data Server, as it was already started.

lcaErrDsNoServer   **DS, #3**

Invalid Data server handle passed in.

lcaErrDsOptionOutOfRange   **DS, #4**

Invalid option passed in.

lcaErrDsNotImplemented   **DS, #5**

Function not implemented.

lcaErrDsNotStarted   **DS, #6**

Function unavailable until the Data Server has been started.

lcaErrDsCantActivate   **DS, #7**

Cannot activate

lcaErrDsCantDeactivate   **DS, #8**

Cannot deactivate.

lcaErrDsIllegalObjectHandle   **DS, #9**

Illegal object handle was passed in.

lcaErrDsReentry   **DS, #10**

Tried to perform an exclusive operation, such as updating a particular network variable on a node, that was already in progress.

lcaErrDsRegistrySetErr   **DS, #11**

Unused.

lcaErrDsAlreadyCreated   **DS, #12**

Tried to create a Data Server when one already existed for this process.

lcaErrDsException   **DS, #13**

A system exception was thrown.

lcaErrDsInvalidObjectType   **DS, #14**

Unused.

lcaErrDsUserCancel   **DS, #15**

The operation specified was canceled interactively at the user interface.

lcaErrDsVersionIncompatibility   **DS, #16**

Some of the DLLs used by the Data Server are older, incompatible versions.

lcaErrDsInitValidationError   **DS, #17**

This indicates that the Data Server failed to initialize properly. This exception could be thrown if your application specifies an invalid name or index number when retrieving an object from the LNS database.

See KB657

lcaErrDsUpdateRegistryFailure   **DS, #18**

Could not update the Windows registry.

lcaErrDsNsCreateClientFailed   **DS, #19**

Failed to create a Network Services client for data services.

lcaErrDsNsInitFailed   **DS, #20**

Failed to initialize the Network Services Server.

lcaErrDsNsNotOpen   **DS, #21**

Error on shutdown, when attempting to close the Network Services server, the server is not open.

lcaErrDsGetHandleFromNameFail   **DS, #22**

Could not get the node handle from its name.

lcaErrDsInvokeNodeAddrFail   **DS, #23**

Could not get the node's address on the network from the Network Services server.

lcaErrDsGetRegistryFailure   **DS, #24**

Failed to retrieve an item from the Windows registry.

lcaErrDsSetRegistryFailure   **DS, #25**

Failed to write an item to the Windows registry.

lcaErrDsInvalidRegKey   **DS, #26**

Failed to set a Windows registry key because the new key passed in was invalid.

lcaErrDsInvalidNodeName   **DS, #27**

An invalid node name was specified, or no name services provider was specified for the LNS.

lcaErrDsNodeDoesNotExist   **DS, #28**

Node does not exist on this network, probably because it has not yet been configured with a network address.

lcaErrDsNvDoesNotExist   **DS, #29**

A network variable operation was attempted on an NV that does not exist.

lcaErrDsNvGetConnInfoFailed   **DS, #30**

A Network Services call to get connection information for this network variable has failed.

lcaErrDsNvGetTypeInfoFailed   **DS, #31**

A Network Services call to get type information for this network variable has failed.

lcaErrDsStartNvObjectFailed   **DS, #32**

Could not start the network variable object.

lcaErrDsDeleteNvObjectFailed   **DS, #33**

Could not delete the network variable object.

lcaErrDsNvSetObjDataFailed   DS, #34

Failed to write data to the network variable object.

lcaErrDsDataFormatFailure   DS, #35

An attempt to format the requested data has failed.

lcaErrDsCreateThreadFailure   DS, #36

Could not create the thread to receive LONWORKS messages from the network.

lcaErrDsCreateEventFailure   DS, #37

Could not create the event to signal arriving LONWORKS messages from the network.

lcaErrDsEventSignalTimeout   DS, #38

Timed out waiting for a synchronizing event to be signaled by a LONWORKS network response.

lcaErrDsCannotForceDataRequest   DS, #39

Unused.

lcaErrDsNvIllegalRetry   DS, #40

The number of retries specified for this network variable is out of range.

lcaErrDsNvIllegalInterval   DS, #41

The interval specified for this network variable is illegal.

lcaErrDsInvalidDriverName   DS, #42

The network interface driver specified does not exist.

lcaErrDsMismatchTypeSize   DS, #43

The type specified is not the correct size to represent this network variable.

lcaErrDsFmtTypeNotFound   DS, #44

The format type referenced does not exist.

lcaErrDsNvIllegalField   DS, #45

The network variable field referenced does not exist.

lcaErrDsInvalidServerHandle   **DS, #46**

The Data Server handle passed in is invalid.

lcaErrDsOptionIsReadOnly   **DS, #47**

Option is read-only. The value may not be set.

lcaErrDsObjectNotStarted   **DS, #48**

Failed because the object referenced must be started first.

lcaErrDsObjectIsAlreadyStarted   **DS, #49**

Cannot start this object, as it is already started.

lcaErrDsSetInvalidServiceType   **DS, #50**

Tried to set the service type to an invalid value.

lcaErrDsNiIsBusy   **DS, #51**

Operation failed because the network interface was busy.

lcaErrDsNiOpenFailure   **DS, #52**

Error opening the network interface.

lcaErrDsNiUnknown   **DS, #53**

Error with the network interface.

lcaErrDsCantWriteRemoteNvOutput   **DS, #54**

It is not possible to write a value to a remote output network variable.

lcaErrDsFormattingInternalError   **DS, #55**

Error loading one of the SNVT or User type (.TYP) files.
See KB190.

lcaErrDsFmtFileLoadFailure   **DS, #56**

Error loading one of the SNVT or User format (.FMT) files.

lcaErrDsBufferTooSmall   **DS, #57**

The specified buffer is too small for the current operation.

lcaErrDsMsgSendFailure   **DS, #58**

Failed to send the current message.

lcaErrDsNoDataAvailable   **DS, #59**

No data is available for retrieval in the current operation.

lcaErrDsTypeFileNotFound   **DS, #60**

The specifed type file could not be found.

lcaErrDsOptionIsMsgobjOnly   **DS, #61**

The specified configuration option only applies to message objects.

lcaErrDsOptionIsNvOnly   **DS, #62**

The specified configuration option only applies to network variable objects.

lcaErrDsNoResource   **DS, #63**

System resources unavailable.

See KB 344.

lcaErrDsInvalidParamValue   **DS, #64**

One or more of the passed parameters has an illegal value.

lcaErrDsMtagDoesNotExist   **DS, #65**

The specified message does not exist.

lcaErrDsRegistryDataInvalid   **DS, #66**

Invalid data was found in a Data Server registry entry.

lcaErrDsFmttypeHasNoFields   **DS, #67**

The format type specified does not contain information about fields.

lcaErrDsEventSubscriptionFailed   **DS, #68**

Failed to subscribe to the specified event.

lcaErrDsCantFindFormatCatalog   **DS, #69**

The format catalog could not be found.

lcaErrDsCantFindStandardTypeFile   **DS, #70**

The specified standard type file could not be found.

lcaErrDsEventDesubscriptionFailed   **DS, #71**

Failed to unsubscribe to the specified event.

lcaErrDsInvalidVersion2TypeName   **DS, #72**

An invalid Version2 format name was supplied. Version 2 type names must start with SNVT, UNVT, SCPT or UCPT. Format names should be taken from the standard and user-defined resource file catalogs located in the directory specified by the LdrfCatlalogPath property.

lcaErrDsInvalidWhenNvFieldInUse   **DS, #75**

The requested operation must be performed on a network variable. It cannot be performed on an individual network variable field.

lcaErrDsInvalidWhenNvFieldNotSpecified   **DS, #76**

The requested operation must be performed on an individual network variable field. It cannot be performed on a network variable.

lcaErrDsGetAppmanagerFailed   **DS, #77**

Failed to specify the application manager.

lcaErrDsCantSetHostMsgOwner   **DS, #78**

Fatal error occurred when attempting to become host message owner. Only one process can be host message owner at any time.

lcaErrDsFmtIllegalBaseType   **DS, #79**

Format specifier is not valid for the type/field.

lcaErrDsNetworkError   **DS, #200**

Failed to communicate with target device.

You could see this error on the Monitor program (e.g., LonMaker Browser) if a device is rebooting, there are network interface problems, there problems with routers or repeaters in between the monitor program and the device, the channel bandwidth is overwhelmed, the LonTalk channel is down or the device stopped.

KB339

LNS will throw DS 200 in many cases. The most common reasons for this error are high bandwidth utilization (collisions that prevent communication) and physical network connectivity problems (lack of terminators in twisted pair networks or other wiring problems).
In rare cases, a race condition has been identified in LNS 3.0. It occurs when actively polling network variables at a high rate using a slow performance PC and a network interface with a layer 5 MIP (also known as an NSI interface). As a result of this problem, it is possible that LNS may stop polling a network variable indefinitely.

Solution
To work around this problem use a layer 2 MIP interface. If this does not resolve the problem, it may be necessary to upgrade to a faster PC or decrease the polling rate until reaching a work load the PC can handle.

Examples of a Layer 5 MIP interface:

PCLTA-10 and PCLTA-20 running NSIPCLTA NI Application
PCC-10 interface using a NSIPCC NI Application
i.LON 10 and iLon100
SLTA-10
U10/U20 when used with OpenLDV program

Examples of a Layer 2 MIP interface are:

PCLTA-10 and PCLTA-20 running PCL10VNI NI Application
PCC-10 interface using a PCC10VNI NI Application
iLon1000 communicating to the PC using a LonWorks/IP channel
U10/U20 when used with LonMaker/LNS/LonScanner

lcaErrDsTargetNodeFailure   **DS, #201**

Target node responded with failure code.

lcaErrLnsDsUnexpected   **DS, #365**

Unexpected failure. Contact customer support.

lcaErrLnsDsFailed   **DS, #366**

Generic failure. Contact customer support.

lcaErrLnsDsOutOfMemory   **DS, #367**

Memory allocation failure.

lcaErrLnsDsNotImpl   **DS, #368**

Not implemented.

lcaErrLnsDsPointer   **DS, #369**

Invalid pointer.

lcaErrLnsDsInvalidArg   **DS, #370**

Invalid argument. One way this error can occur is if you call the Write method on a DataPoint object obtained from a NetworkVariable or NvMonitorPoint object before reading or setting its value with the FormattedValue property, the RawValue property, or the Value property. This indicates that the DataPoint's buffer is empty.

lcaErrLnsDsRange   **DS, #371**

Argument supplied is out of valid range.

lcaErrLnsDsTimeout   **DS, #372**

Timeout occurred.

lcaErrLnsDsSizeTooSmall   **DS, #373**

Size is too small.

lcaErrLnsDsInvalidObject   **DS, #374**

Object not valid.

lcaErrLnsDsInvalidWindow   **DS, #375**

Invalid window.

lcaErrLnsDsInvalidOperation   **DS, #376**

Operation not valid, or the operation was attempted on an invalid object.

lcaErrLnsDsQueueFull   **DS, #377**

The message queue is full and the most recent message has been dropped.

lcaErrLnsDsQueueStopped   **DS, #378**

The message queue was stopped by a client application and the most recent message has been dropped.

lcaErrLnsDsOutstandingRefs   **DS, #379**

Outstanding object references still exist.

lcaErrLnsDsOutstandingClients   **DS, #380**

Outstanding clients still exist.

lcaErrLnsDsOutstandingNetworks   **DS, #381**

Outstanding networks still exist.

lcaErrLnsDsOutstandingVNIs   **DS, #382**

Outstanding VNIs still exist.

lcaErrLnsDsOutstandingMonitorSets   **DS, #383**

Outstanding monitor sets still exist.

lcaErrLnsDsNotFound   **DS, #384**

Object not found.

lcaErrLnsDsClientNotFound   **DS, #385**

Client not found.

lcaErrLnsDsNetworkNotFound   **DS, #386**

Network not found.

lcaErrLnsDsVniNotFound   **DS, #387**

VNI not found.

lcaErrLnsDsMonitorSetNotFound   **DS, #388**

Monitor set not found.

lcaErrLnsDsMonitorPointNotFound   **DS, #389**

Monitor point not found.

lcaErrLnsDsNodeNotFound   **DS, #390**

Node not found.

lcaErrLnsDsNetworkVariableNotFound   **DS, #391**

Network variable not found.

lcaErrLnsDsAppMessageNotFound **DS, #392**

Application message not found.

lcaErrLnsDsNotOpen **DS, #393**

Object must be opened before the requested operation can be performed.

lcaErrLnsDsNetworkNotOpen **DS, #394**

Network must be opened before the requested operation can be performed.

lcaErrLnsDsVniNotOpen **DS, #395**

VNI must be opened before the requested operation can be performed.

lcaErrLnsDsMonitorSetNotOpen **DS, #396**

Monitor set must be opened before the requested operation can be performed.

lcaErrLnsDsNotConnected **DS, #397**

The client must be connected to the server before the requested operation can be performed.

lcaErrLnsDsOpenFailed **DS, #398**

There was an error opening the object.

lcaErrLnsDsVniOpenFailed **DS, #399**

There was an error opening the VNI.

lcaErrLnsDsMonitorSetOpenFailed **DS, #400**

There was an error opening the monitor set.

lcaErrLnsDsCloseFailed **DS, #401**

There was an error closing the object.

lcaErrLnsDsVniCloseFailed **DS, #402**

There was an error closing the VNI.

lcaErrLnsDsThreadCreateFailed **DS, #403**

There was an error creating the thread.

lcaErrLnsDsNotTemporaryMonitorSet   **DS, #404**

The requested operation can only be performed on temporary monitor sets.

lcaErrLnsDsNotTemporaryMonitorPoint   **DS, #405**

The requested operation can only be performed on temporary monitor points.

lcaErrLnsDsLNSNotOpen   **DS, #406**

LNS must be opened before the requested operation can be carried out.

lcaErrLnsDsLNSCreateClientFailed   **DS, #407**

There was an error creating the LNS client.

lcaErrLnsDsLNSInitFailed   **DS, #408**

There was an error initializing LNS.

lcaErrLnsDsLNSException   **DS, #409**

Error occurred during an LNS operation.

lcaErrLnsDsReadFailed   **DS, #410**

Network variable read failed.

lcaErrLnsDsWriteFailed   **DS, #411**

Network variable write failed.

lcaErrLnsDsPollFailed   **DS, #412**

Network variable poll failed.

lcaErrLnsDsSendFailed   **DS, #413**

Application message transmission failed.

lcaErrLnsDsMsgRejected   **DS, #414**

Application message rejected, as network management commands are restricted.

lcaErrLnsDsRequestFailed   **DS, #415**

Application message request failed.

lcaErrLnsDsResponseFailed   **DS, #416**

Application message response failed.

lcaErrLnsDsAddressNotAvailable   **DS, #417**

The source node address is not available.

lcaErrLnsDsIllegalHexCharacter   **DS, #418**

An illegal hexadecimal character was specified in the string.

lcaErrLnsDsTooManyClients   **DS, #419**

There are too many clients connected to the server.

lcaErrLnsDsInvalidClientContext   **DS, #420**

Invalid or no client context was supplied.

lcaErrLnsDsImplicitAddress   **DS, #421**

The requested operation is not allowed, as the monitor point uses implicit addressing.

lcaErrLnsDsExplicitAddress   **DS, #422**

The requested operation is not allowed, as the monitor point uses explicit addressing.

lcaErrLnsDsNotOwner   **DS, #423**

The requested operation is not allowed, as the client is not the owner of the object or process affected.

lcaErrLnsDsNoNameServicesManager   **DS, #424**

Cannot look up the requested name, as there is no name service manager defined.

lcaErrLnsDsNameTooLong   **DS, #425**

The name specified is too long.

lcaErrLnsDsCountMismatch   **DS, #426**

Reference/Lock count mismatch. This could be caused by too many Release calls being made by the application.

lcaErrLnsDsNotPermanentMonitorSet   **DS, #428**

The requested operation can only be performed on a permanent monitor set.

lcaErrLnsDsNotPermanentMonitorPoint   **DS, #429**

The requested operation can only be performed on a permanent monitor point.

lcaErrLnsDsException   **DS, #430**

An unexpected exception occurred.

lcaErrLnsDsInitUpdateFailed   **DS, #431**

A network variable could not be updated.

Formatter Errors

These are errors returned by the LNS formatter. These errors may be returned when LNS fails to read data from a resource file properly. These errors will appear in the format Subsystem: Formatter, #<Error Number>.

lcaErrFormatNotFound   **Formatter, #1**

Not found.

lcaErrFormatNotImplemented   **Formatter, #2**

Not implemented.

lcaErrFormatInvalidLocaleData   **Formatter, #3**

Invalid locale data.

lcaErrFormatTypeNotFound   **Formatter, #4**

Type not found.

lcaErrFormatInvalidTypFile   **Formatter, #5**

Invalid type file.

lcaErrFormatReferenceEnumScopeNotFound   **Formatter, #6**

Reference enumeration scope not found.

lcaErrFormatTypeEntryNotFound   **Formatter, #7**

Type entry not found.

lcaErrFormatFieldNotFound   **Formatter, #8**

Field not found.

lcaErrFormatIllegalBaseType   **Formatter, #9**

Format specifier is not valid for this type or field.

lcaErrFormatAchNotFound   **Formatter, #10**

Format file not found.

lcaErrFormatStringIllegal   **Formatter, #11**

Illegal string.

lcaErrFormatStringExceedsBounds   **Formatter, #12**

Array index exceeds array size.

lcaErrFormatSpecNotFound   **Formatter, #13**

Format specification not found.

lcaErrFormatBuildAchFailed   **Formatter, #14**

Compilation of format file failed.

lcaErrFormatIllegalDataOnFormat   **Formatter, #15**

Illegal data on format.

lcaErrFormatIllegalDataOnUnformat   **Formatter, #16**

Illegal data on unformat.

lcaErrFormatNoUnformat   **Formatter, #17**

No format.

lcaErrFormatExceedsSize   **Formatter, #18**

Exceeds size.

lcaErrFormatIllegalEnum   **Formatter, #19**

Illegal enumeration.

lcaErrFormatExceedsMaxNumFields   **Formatter, #20**

Exceeds maximum number of fields.

lcaErrFormatNoResource   **Formatter, #21**

No resource.

lcaErrFormatInvalidBuiltInType   **Formatter, #22**

Invalid built-in type.

lcaErrFormatNotBuiltInSpec   **Formatter, #23**

Not built in spec.

lcaErrFormatInvalidPropSet   **Formatter, #24**

Invalid prop set.

lcaErrFormatInvalidBaseType   **Formatter, #25**

Invalid base type.

lcaErrFormatFmtNotFound   **Formatter, #26**

Format not found.

lcaErrFormatInvalidDefaultType   **Formatter, #27**

Invalid default type.

lcaErrFormatUnknownError   **Formatter, #28**

Unknown error.

lcaErrFormatFileNotFoundInCatalog   **Formatter, #29**

File not found in catalog.

lcaErrFormatCatalogNotOpened   **Formatter, #30**

Catalog not opened.

lcaErrFormatEnumNotFound   **Formatter, #31**

Enumeration not found.

lcaErrFormatTypeNotOpened   **Formatter, #32**

Type not opened.

lcaErrFormatNotVersion2TypeName   **Formatter, #33**

Not version 2 type name.

lcaErrFormatInvalidParameter   **Formatter, #34**

Invalid parameter.

lcaErrFormatInvalidBfSize   **Formatter, #35**

Invalid buffer size.

lcaErrFormatInvalidBfOffset   **Formatter, #36**

Invalid buffer offset.

lcaErrFormatEnumNonuniqueSubstr   **Formatter, #37**

Substring match was not unique.

lcaErrFormatRawInputTooSmall   **Formatter, #38**

Raw input too small.

lcaErrFormatNoFormatFileForTypeFile   **Formatter, #39**

No format file for type file.

lcaErrFormatStringDuplicate   **Formatter, #40**

Duplicate string provided. This exception will be thrown if you write to the
CategoryPreferenceList property and specify the same format more than once.

lcaErrFormatLocaleStringNotFound   **Formatter, #41**

Locale string not found.

lcaErrFormatCatalogNotFound   **Formatter, #200**

Catalog not found.

lcaErrFormatInvalidProgramId   **Formatter, #201**

Invalid program ID.

lcaErrFormatLdrfErr   **Formatter, #202**

LDRF error.

lcaErrFormatLdrfErrParam   **Formatter, #203**

LDRF error: bad parameter.

lcaErrFormatLdrfErrFileType   **Formatter, #204**

LDRF error: bad file type.

lcaErrFormatLdrfErrCrc   **Formatter, #205**

LDRF error: bad CRC.

lcaErrFormatLdrfErrNotFound　　　　　　　　**Formatter, #206**
LDRF error: not found.

lcaErrFormatLdrfErrFileInfo　　　　　　　　**Formatter, #207**
LDRF error: bad file info.

lcaErrFormatLdrfErrSys　　　　　　　　**Formatter, #208**
LDRF error: sys.

lcaErrFormatLdrfErrTrunc　　　　　　　　**Formatter, #209**
LDRF error: trunc.

lcaErrFormatLdrfErrStale　　　　　　　　**Formatter, #210**
LDRF error: stale.

lcaErrFormatLdrfErrVersion　　　　　　　　**Formatter, #211**
LDRF error: bad version.

lcaErrFormatLdrfErrNew　　　　　　　　**Formatter, #212**
LDRF error: new.

lcaErrFormatLdrfErrWrite　　　　　　　　**Formatter, #213**
LDRF error: write failed.

lcaErrFormatLdrfErrNoAccess　　　　　　　　**Formatter, #214**
LDRF error: no access.

lcaErrFormatLdrfErrFull　　　　　　　　**Formatter, #215**
LDRF error: full.

lcaErrFormatLdrfErrDuplicate　　　　　　　　**Formatter, #216**
LDRF error: duplicate.

lcaErrFormatLdrfErrNotCatalog　　　　　　　　**Formatter, #217**

LDRF error: not catalog.

lcaErrFormatLdrfErrNotResource   **Formatter, #218**
LDRF error: not resource.

lcaErrFormatLdrfErrNotType   **Formatter, #219**
LDRF error: not type.

lcaErrFormatLdrfErrNotFpt   **Formatter, #220**
LDRF error: not FTP.

lcaErrFormatLdrfErrNotFormat   **Formatter, #221**
LDRF error: not format.

lcaErrFormatLdrfErrTypeTree   **Formatter, #222**
LDRF error: bad type tree.

lcaErrFormatLdrfErrIncomplete   **Formatter, #223**
LDRF error: incomplete.

lcaErrFormatLdrfErrSequence   **Formatter, #224**
LDRF error: sequence.

lcaErrFormatLdrfErrNotSelected   **Formatter, #225**
LDRF error: not selected.

lcaErrFormatLdrfErrInternal   **Formatter, #226**
LDRF error: internal error.

lcaErrFormatObjReadOnly   **Formatter, #227**
Read-only object.

VNI Server Errors

VNI errors will be returned when an operation fails while using the MyVni property. These errors will appear in the format VNI: #<Error Number>.

lcaErrRmoMaxObjectsAllocated   V**NI, #1**

Maximum objects allocated.

lcaErrRmoBadObjectId   V**NI, #2**

Bad object ID.

lcaErrRmoProcessTimeout   V**NI, #3**

Process timeout.

lcaErrRmoNoServerObject   V**NI, #4**

Remote object does not exist.

lcaErrRmoNoIpcMsgAvail   V**NI, #5**

No Windows Inter Process Communication (IPC) message available.

lcaErrRmoIpcUnknownError   V**NI, #6**

Unknown Windows Inter Process Communication (IPC) error.

lcaErrRmoIpcMsgError   V**NI, #7**

Windows Inter Process Communication (IPC) messaging error.

lcaErrRmoIpcResourceProblem   V**NI, #8**

Windows Inter Process Communication (IPC) resource problem.

lcaErrRmoIpcOutOfMemory   V**NI, #9**

Windows Inter Process Communication (IPC) subsystem is out of memory.

lcaErrRmoIpcOutOfRange   V**NI, #10**

Windows Inter Process Communication (IPC) parameter is out of range.

lcaErrRmoIpcCantFindObject   V**NI, #11**

Windows Inter Process Communication (IPC) subsystem cannot find specified object.

lcaErrRmoIpcLockFailure   V**NI, #12**

Windows Inter Process Communication (IPC) subsystem lock failure.

lcaErrRmoCantLinkToObject   V**NI, #13**

Cannot link to remote object.

lcaErrLtaNoError   V**NI, #300**

LonTalk Adapter error.

lcaErrLtaInvalidParameter   V**NI, #301**

LonTalk Adapter invalid parameter.

lcaErrLtaNotQualified   V**NI, #302**

LonTalk Adapter not qualified.

lcaErrLtaMessageBlocked   V**NI, #303**

LonTalk Adapter message blocked.

lcaErrLtaMessageDeferred   V**NI, #304**

LonTalk Adapter message deferred.

lcaErrLtaAppMessage   V**NI, #305**

LonTalk Adapter application message.

lcaErrLtaFlexDomain   V**NI, #306**

LonTalk Adapter flex domain.

lcaErrLtaNoMessage   V**NI, #307**

LonTalk Adapter no message.

lcaErrLtaAppNameTooLong   V**NI, #308**

LonTalk Adapter name too long.

lcaErrLtaInvalidState   V**NI, #309**

LonTalk Adapter invalid state.

lcaErrLtaNoResources   V**NI, #310**

LonTalk Adapter no resources.

lcaErrLtaDuplicateObject   V**NI, #311**

LonTalk Adapter duplicate object.

lcaErrLtaNotImplemented   V**NI, #312**

LonTalk Adapter not implemented.

lcaErrLtaEndOfEnumeration   V**NI, #313**

LonTalk Adapter end of enumeration.

lcaErrLtaOwnerDoesNotExist   V**NI, #314**

LonTalk Adapter owner does not exist.

lcaErrLtaInvalidIndex   V**NI, #315**

LonTalk Adapter invalid index.

lcaErrLtaCantOpenPort   V**NI, #316**

LonTalk Adapter cannot open port.

lcaErrLtaNotFound   V**NI, #317**

LonTalk Adapter not found.

lcaErrLtaNoWinsockDll   V**NI, #318**

LonTalk Adapter no Winsock DLL.

lcaErrLtaCantOpenIpLink   V**NI, #319**

LonTalk Adapter cannot open IP link.

lcaErrLtaCantStartSnmp   V**NI, #320**

LonTalk Adapter cannot start SNMP.

lcaErrLtaNoLink   V**NI, #321**

LonTalk Adapter no IP link.

lcaErrLtaInvalidIpaddress   V**NI, #322**

LonTalk Adapter invalid IP address.

lcaErrLtaLocalMsgFailure   V**NI, #323**

LonTalk Adapter local message failure.

lcaErrLtaStaleNvIndex   V**NI, #324**

LonTalk Adapter stale network variable index.

lcaErrLtaInvalidAddress   V**NI, #325**

LonTalk Adapter invalid address.

lcaErrLtaErrorLogMask   V**NI, #428**

LonTalk Adapter error log mask.

lcaErrLtaNvLengthMismatch   V**NI, #430**

LonTalk Adapter network variable length mismatch.

lcaErrLtaNvMsgTooShort   V**NI, #431**

LonTalk Adapter network variable message too short.

lcaErrLtaEepromWriteFailure   V**NI, #432**

LonTalk Adapter EEPROM write failure.

lcaErrLtaBadAddressType   V**NI, #433**

LonTalk Adapter bad address.

lcaErrLtaInvalidDomain   V**NI, #438**

LonTalk Adapter invalid domain.

lcaErrLtaInvalidAddrTableIndex   V**NI, #441**

LonTalk Adapter invalid address table index.

lcaErrLtaNvUpdateOnOutputNv   V**NI, #443**

LonTalk Adapter update on output network variable.

lcaErrLtaUnknownPdu   V**NI, #446**

LonTalk Adapter unknown PDU.

lcaErrLtaInvalidNvIndex   V**NI, #447**

LonTalk Adapter invalid network variable index.

lcaErrLtaBadErrorNo   V**NI, #449**

LonTalk Adapter bad buffer number.

lcaErrLtaNetBufTooSmall   V**NI, #451**

LonTalk Adapter network buffer too small.

lcaErrLtaCnfgCsError   V**NI, #453**

LonTalk Adapter configuration checksum error.

lcaErrLtaXcvrRegOpFailure   V**NI, #455**

LonTalk Adapter transceiver register operation failed.

lcaErrLtaSubnetPartition   V**NI, #459**

LonTalk Adapter subnet partition.

lcaErrLtaAuthenticationMismatch   V**NI, #460**

LonTalk Adapter authentication mismatch.

lcaErrVniUnknownRegError   V**NI, #601**

Unknown registry error.

lcaErrVniRegBadParms   V**NI, #602**

Bad parameter (registry subsystem).

lcaErrVniRegOutOfMemory   V**NI, #603**

Out of memory (registry subsystem).

lcaErrVniRegCantFindObject   V**NI, #604**

Cannot find object (registry subsystem).

lcaErrVniRegResourceProblem   V**NI, #605**
Resource problem (registry subsystem).

lcaErrVniRegOutOfRange   V**NI, #606**
Parameter out of range (registry subsystem).

lcaErrVniRegNoMoreEntries   V**NI, #607**
No more entries (registry subsystem).

lcaErrVniRegCantOpenRegistry   V**NI, #608**
Cannot open registry.

lcaErrVniRegUnknownRegistryError   V**NI, #609**
Unknown error (registry subsystem).

lcaErrVniVniDoesNotExist   V**NI, #610**
Specified VNI does not exist.

lcaErrVniCantCreateVniProcess   V**NI, #611**
Cannot create the remote process.

lcaErrVniCantLockVniServerDir   V**NI, #612**
Cannot lock the server directory.

lcaErrVniOpenObjectConflict   V**NI, #613**
Tried to use two different objects to represent the same VNI object.

lcaErrVniOpenFailure   V**NI, #614**
VNI open failure.

lcaErrVniNoMoreMonitorPoints   V**NI, #615**
No more monitor points found.

lcaErrVniNoMoreMonitorSets   V**NI, #616**

No more monitor sets found.

lcaErrVniStackNotOpen V**NI, #617**

VNI stack is not open.

lcaErrVniNotImplemented V**NI, #618**

Not implemented.

lcaErrVniMonitorSetNotFound V**NI, #619**

Monitor set not found.

lcaErrVniNvPointNotFound V**NI, #620**

Network variable monitor point not found.

lcaErrVniMsgPointNotFound V**NI, #621**

Message monitor point not found.

lcaErrVniMustSpecifyMsgTag V**NI, #622**

Must specify message tag.

lcaErrVniMsgTagNotFound V**NI, #623**

Message tag not found.

lcaErrVniOutOfTemporaryMonitorPoints V**NI, #624**

Monitor set ran out of temporary monitor points due to capacity constraints.

lcaErrVniOutOfTemporaryMonitorSets V**NI, #625**

VNI stack ran out of temporary monitor sets due to capacity constraints.

lcaErrVniAddPointToPersistentMonitorSet V**NI, #626**

Attempted to add a point to a persistent monitor set.

lcaErrVniDelPointFromPersistentMonitorSet V**NI, #627**

Attempted to delete a point from a persistent monitor set.

lcaErrVniAbosoluteRegPathNotSupported V**NI, #628**

Absolute registry paths not supported.

lcaErrVniRegBadCollectionPath   VNI, #629

Cannot define a collection of VNIs at this level in the registry.

lcaErrVniOpenStackNiMismatch   VNI, #630

The VNI is already open with different network interface settings.

lcaErrVniTraceFileCantBeOpened   VNI, #631

The specified trace file cannot be opened.

lcaErrVniNoMoreMessageBuffers   VNI, #632

The client has more messages outstanding than permitted.

lcaErrVniIsNotAnIpDevice   VNI, #633

The specified device is not an IP device.

lcaErrVniL5MipInUse   VNI, #634

The requested operation is not supported while a Layer 5 network interface is in use.

lcaErrVniXifCannotBeFound   VNI, #635

The external interface file cannot be found.

lcaErrVniXifReadError   VNI, #636

Failed to read external interface file.

lcaErrVniAllocation   VNI, #637

VNI allocated error.

lcaErrVniProgramInterfaceMismatch   VNI, #638

Program interface mismatch.

lcaErrVniRegConfigFileNotFound   VNI, #639

Configuration file could not be found.

lcaErrVniRegCannotUpdateConfigFile   VNI, #640

Configuration file could not be updated.

lcaErrVniThreadCreationFailure   V**NI, #641**

Failed to create thread.

lcaErrVniDumpLtipXmlConfigFailed   V**NI, #642**

XML configuration failed.

lcaErrVniNotSupportedOnL5Mip   V**NI, #643**

The requested operation is not supported on a Layer 5 network interface.

lcaErrVniInvalidMessagePointOptions   V**NI, #644**

Invalid message point options.