Issued: 30.08.2001
Version: A
Program revision: 1.0

Installation and Commissioning Guide

Notice

The information in this document is subject to change without notice and should not be construed as a commitment by ABB. ABB assumes no responsibility for any error that may occur in this document.

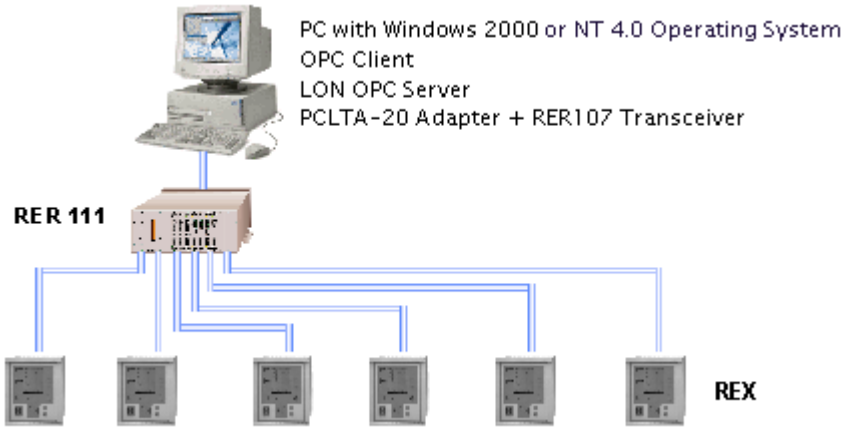
Trademarks

Contents:

1. Introduction

The LON OPC Server provides methods for OPC Clients to exchange data with units communicating via LON/LAG protocol. LON/LAG is a proprietary method of ABB for secured and efficient data transfer on top of standard LON protocol.

The LON OPC Server supports a subset of LON/LAG version 1.4 functions. The focus is set to the sliding window communication and the functions most commonly used in SA systems. The LON OPC Server does not support network variable communication, including communication with LON SPA Gateway and 3rd party LONMARK®[1] devices.



LON_OPC_Server.tif

Fig. 1.-1 An example system overview

The LON OPC Server supports OPC Data Access v.1.0/2.0 and OPC Alarm and Event interfaces. It can be run on the same computer with an OPC Client (see Fig. 1.-1) or on a separate communication server computer. If the LON OPC server is running on a separate computer, the communication between the client and the server is based on Distributed Component Object Model (DCOM).

1. LONMARK is a trademark of Echelon Corporation registered in the United States and other countries.

2. Safety information

The purpose of this chapter is to provide information about the prevention of hazards.

2.1. Backup

All the project specific data is stored into the configuration file OPCS_NET.INI. Therefore it is usually enough to take a backup only from this file. Reinstalling the LON OPC Server software will restore the rest of the required files.

There is software available for taking a complete image backup of the system. The backup image will then contain both the system and application specific files.

2.2. Fatal errors

A fatal error is an error that causes a break-down or a locked situation in the program execution.

Handling

In case of a fatal error:

1 Write down the possible LON OPC Server error messages.

2 If necessary, shut down the LON OPC Server program in the Windows[TM2] Task Manager.

3 The data kept in the main memory at the moment of a fatal error is placed in the drwtsn32.log file. It is placed in a system folder, for example Winnt. Analyze and copy the data in this file.

Report the program break-down together with the possible error messages and the information from drwtsn32.log file to the LON OPC Server supplier.

2. Windows is a trademark of Microsoft Corporation.

3. Installation

This chapter provides information about the system requirements for the LON OPC Server. It also describes the installation procedure of the LON OPC Server and the installation and basic configuration of the PCLTA-20 LONTALK®[3] Adapter.

3.1. System requirements

The LON OPC Server runs under the Windows 2000 and Windows NT 4.0 Operating System. A PC capable of running one of these operating systems and applications is usually sufficient also for the LON OPC Server.

Other system requirements can be seen below.

- 10 MB free hard disk space
- PCLTA-20 adapter + RER107 Transceiver

The following issues affect the performance of the system, and should also be considered when choosing the hardware:

- The number of connected units
- The number of signals per unit
- The frequency of signal changes
- The number of OPC Clients connected to the server
- Whether the OPC Client is run on the same computer or on a separate one

3.2. Installation and configuration of the PCLTA-20 adapter

This section explains how to install PCLTA-20 adapter software to Windows 2000. Installation for Windows NT 4.0 follows the same steps. Installation package can be obtained from Echelon's website (www.echelon.com) as mentioned in LONWORKS Network Interface Cards Installation Instructions.

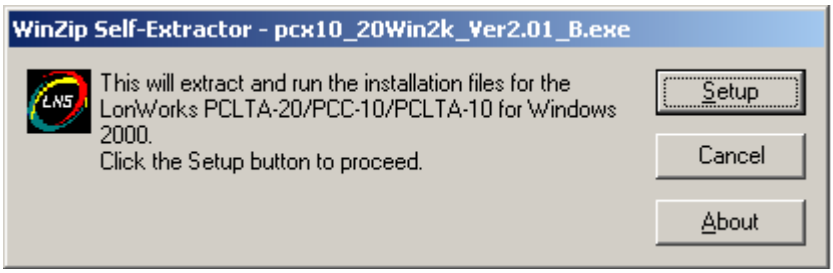
3.2.1. PCLTA-20 software installation

To install the PCLTA-20 software on a Windows NT 4 or Windows 2000 PC, follow the instructions below.

1 Close all the open programs. Do not install PCLTA-20 into PCI slot at this time.

2 Start the PCLTA-20 software installation program. Choose Setup in the WinZip Self-Extractor dialog (see Fig. 3.2.1.-1).

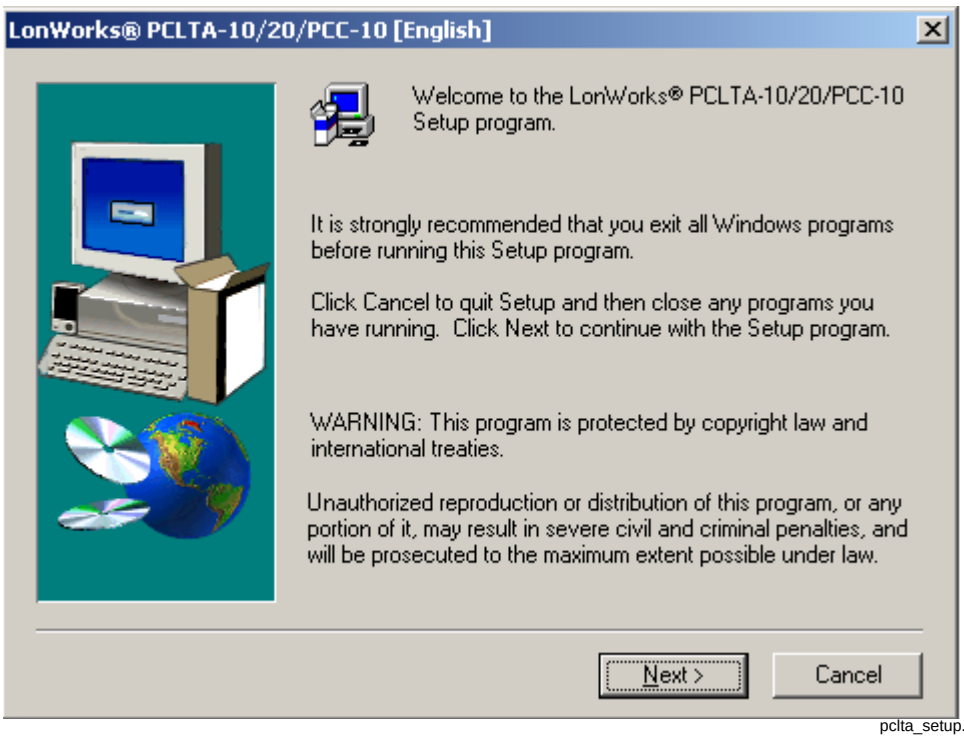
3. LONTALK is a trademark of Echelon Corporation registered in the United States and other countries.



pclta_inst.tif

Fig. 3.2.1.-1 The WinZip Self-Extractor dialog

3 In the Welcome dialog, select Next to continue (see Fig. 3.2.1.-2).



pclta_setup.tif

Fig. 3.2.1.-2 The Welcome dialog

4 The Software License Agreement dialog appears next (see Fig. 3.2.1.-3). Choose Yes to accept the terms of the licence agreement and to continue with the installation.

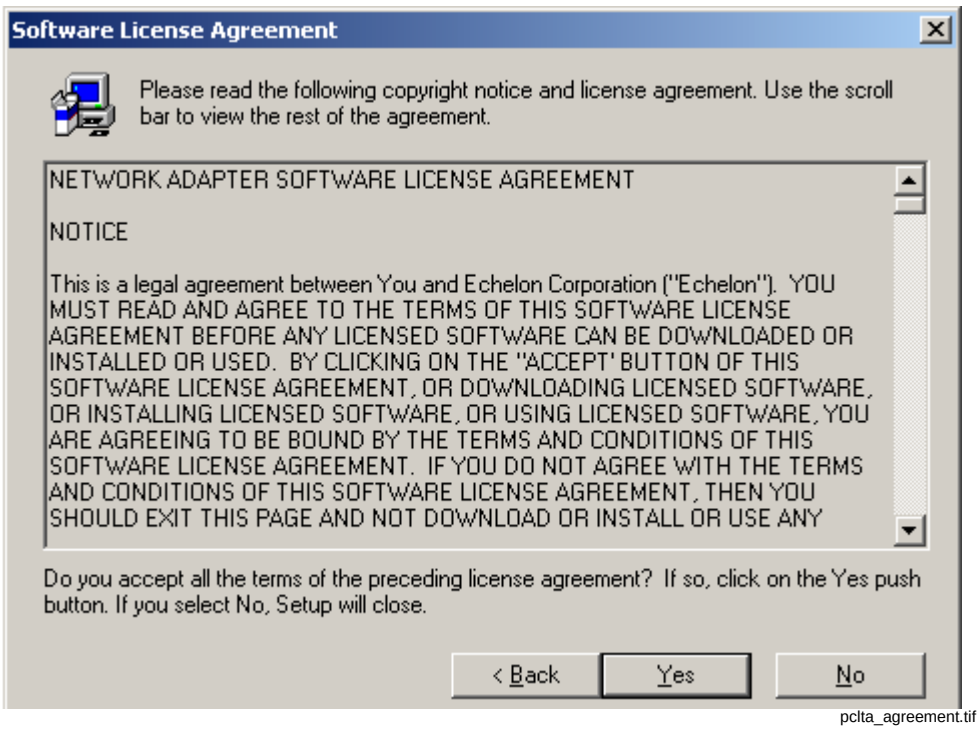
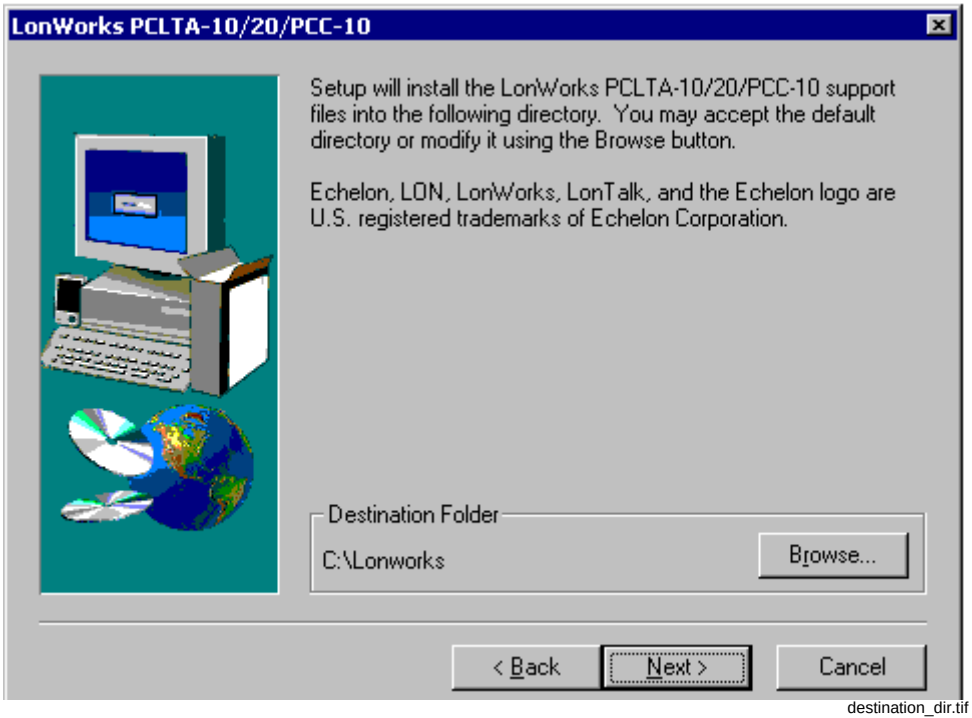


Fig. 3.2.1.-3 The Software License Agreement dialog

5 In Destination directory dialog, enter the desired installation directory. By default, this directory is c:\lonworks (see Fig. 3.2.1.-4).

If you want to modify the path, use the Browse button. However, if any other directory than c:\lonworks is chosen, the PCLTA-20 images path has to be specified to enable the use of the PCLTA-20 adapter. You can specify the PCLTA-20 images path during the PCLTA configuration.

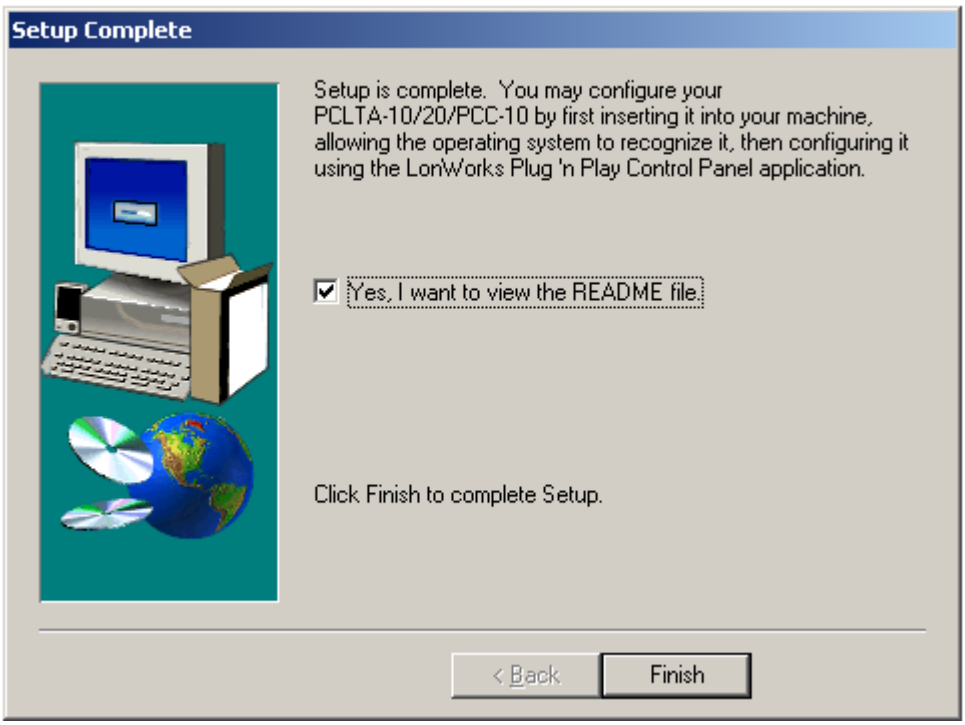
Choose Next to continue.



destination_dir.tif

Fig. 3.2.1.-4 The Destination directory dialog

6 The Setup complete (readme) dialog is displayed. You can view the Readme file by checking the Readme check box. Choose Finish to complete the installation (see Fig. 3.2.1.-5).



finish_setup.tif

Fig. 3.2.1.-5 The Setup Complete (Readme) dialog

7 Software installation is now complete. When prompted to restart computer, choose "No, I will restart my computer later" option (see Fig. 3.2.1.-6). Then choose Finish.

Finally, shut down the computer, install the PCLTA-20 adapter and then reboot the computer.

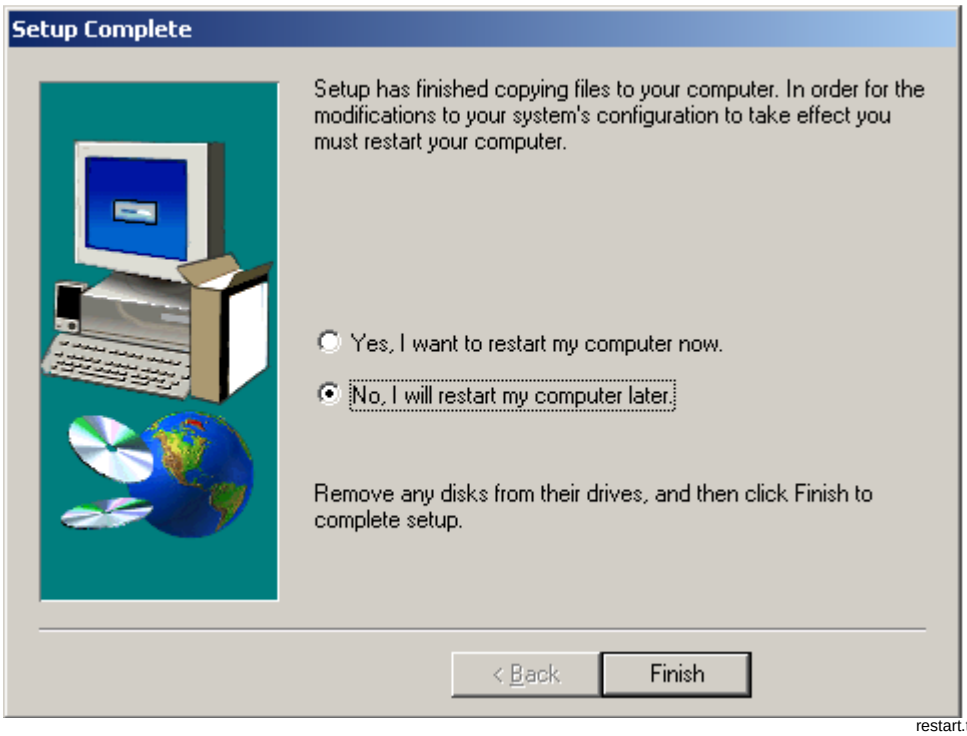


Fig. 3.2.1.-6 The Setup complete dialog

3.2.2. PCLTA-20 hardware installation

Please, make sure to check that RER107 Transceiver is attached into PCLTA-20 adapter before inserting the PCLTA-20 card into the slot.

1 Turn off the PC and remove the power cord.

2 Open the PC case and locate an empty 32-bit PCI slot. Remove the corresponding blank panel from the rear of the PC. Set aside the screw.

3 Insert the PCLTA-20 card into the slot. Make sure that the edge connectors are fully mated and the slot in the rear panel mounting plug of PCLTA-20 card is aligned with the threaded hole in the PC chassis.

4 Replace the screw to hold the PCLTA-20 card firmly in place.

5 Reinsert the power cord and then restart the PC. A New Hardware Found window will be displayed briefly when Windows recognizes the PCLTA-20 card.

3.2.3. Configuration settings for PCLTA-20 adapter

PCLTA-20 configuration is accomplished by using the LONWORKS®[4] Plug 'n Play control panel application. Open the control panel application by selecting the LONWORKS Plug 'n Play icon in the Windows Control Panel.

For LON OPC Server, there are five important settings in the LONWORKS Plug 'n Play control panel application. They should be set as described below and as shown in Fig. 3.2.3.-1 and Fig. 3.2.3.-2.

Device Selected

Controls which PCLTA-20 adapter is selected for configuration.

NI Application

Controls the type of image or application to be used. A PCLTA-20 adapter can only hold one image at a time. Image for the LON OPC Server should be set to PCC10L7.

Automatic Flush Cancel

Should be checked.

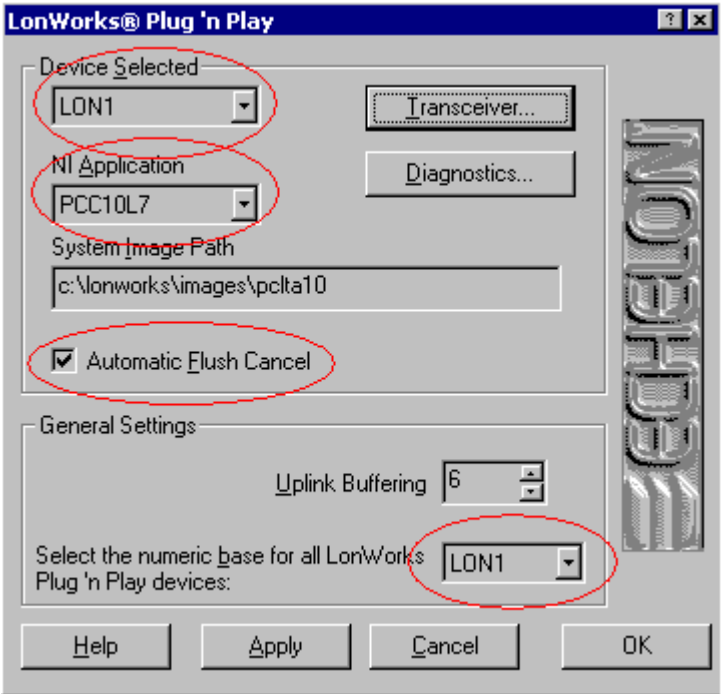
Transceiver

Opens the PCLTA-20 Transceiver dialog. Settings should be set as shown in Fig. 3.2.3.-2.

System Image Path

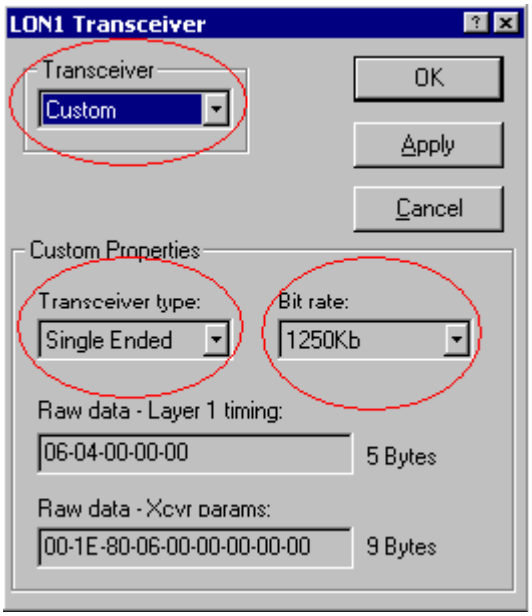
Specifies the full directory path for the PCLTA-20 system images. This path is set by the PCLTA-20 Installation software but may be modified by the user.

4. LONWORKS is a trademark of Echelon Corporation registered in the United States and other countries.



lonworks_pp.tif

Fig. 3.2.3.-1 PCLTA Control Panel



lon1_trans.tif

Fig. 3.2.3.-2 Transceiver Control Panel

With these settings the PCLTA-20 adapter should work with the LON OPC Server. For more details concerning the PCLTA-20 adapter, see the PCLTA-20 PCI Interface User's Guide by Echelon Corporation.

3.3. Installation of the LON OPC Server

To install the LON OPC Server:

1 Start the installation program. The Welcome dialog is displayed. Click OK to continue with the installation or choose Cancel if you do not want to install (see Fig. 3.3.-1).

Fig. 3.3.-1 The Welcome dialog

2 The Product License Agreement dialog appears on screen. Read the license agreement and click Yes to accept the terms of the Product License Agreement and to continue (see Fig. 3.3.-2).

license_agreement.tif

Fig. 3.3.-2 The Product License Agreement dialog

3 The LON OPC Server Information dialog is displayed. Click Next to continue the installation or Exit to cancel the installation (see Fig. 3.3.-3).

Fig. 3.3.-3 The LON OPC Server Information dialog

4 The LON OPC Server Installation dialog appears on screen (see Fig. 3.3.-4). It shows the available software packages. By checking an approriate box, you can choose which package / packages you want to install.

serverinstall.tif

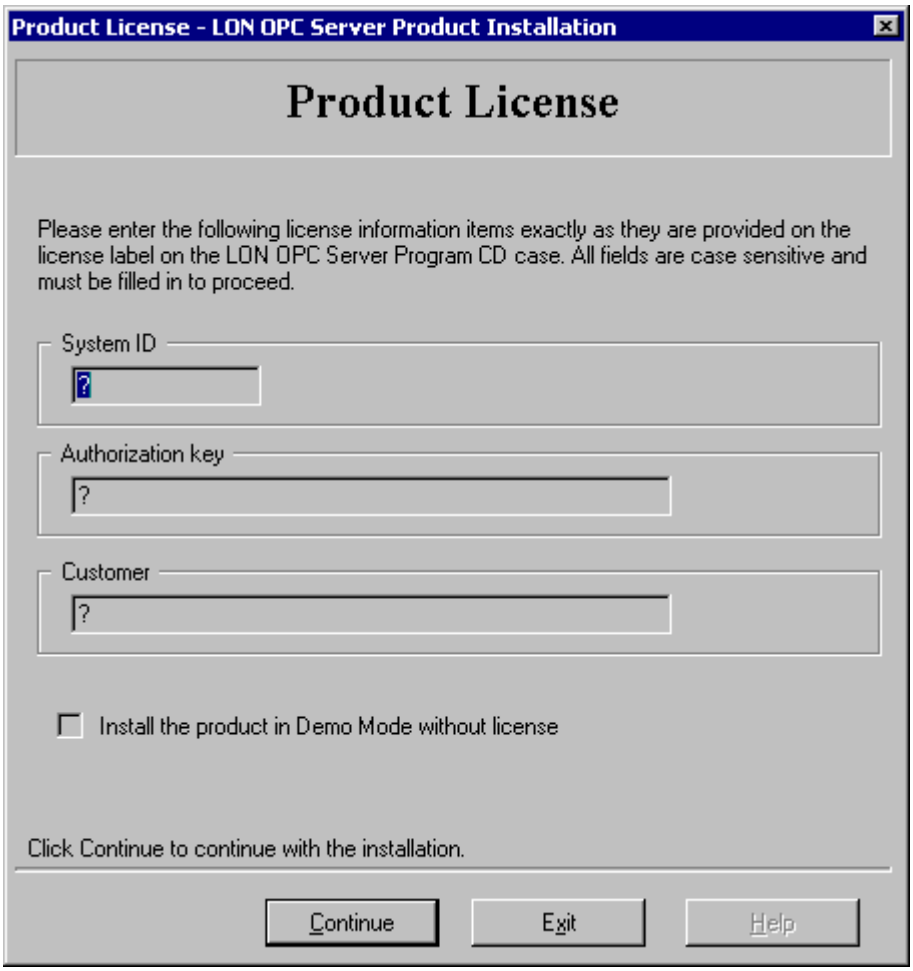
Fig. 3.3.-4 The LON OPC Server Installation dialog

In case you want to change the destination directory, click Change Directory and browse for the directory you want to install in. In the LON OPC Server Installation dialog (see Fig. 3.3.-4), you can also see the required and the available hard disk space.

To continue the installation, click Start. You can also cancel the installation at this point by selecting Exit.

5 In the Product License dialog (see Fig. 3.3.-5), enter the required values for the license information items and click Continue. The necessary values are provided on the program CD case.

You can also install the product without a license in Demo Mode by checking the “Install the product in Demo Mode without license” check box and then selecting Continue.



license.tif

Fig. 3.3.-5 The Product License dialog

6 In the Installation/Upgrade Complete dialog (see Fig. 3.3.-6), click OK.

complete.tif

Fig. 3.3.-6 The Installation/Upgrade Complete dialog

The installation of the LON OPC Server is now complete.

4. Features

The purpose of this chapter is to describe the basic features of the LON OPC Server. This chapter also describes concepts like OPC Data Access Namespace, OPC Alarms and Events Area Space and mapping of LON quality information to OPC item quality.

The software component structure of the LON OPC Server is shown in Fig. 4.-1.

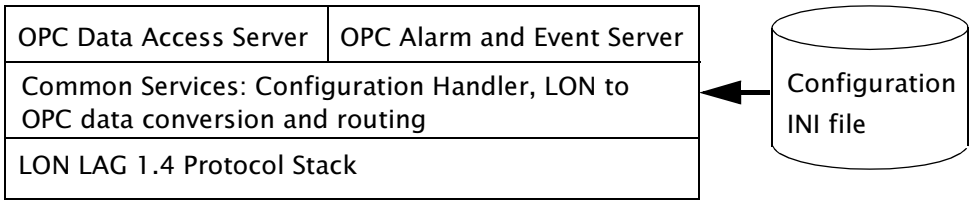
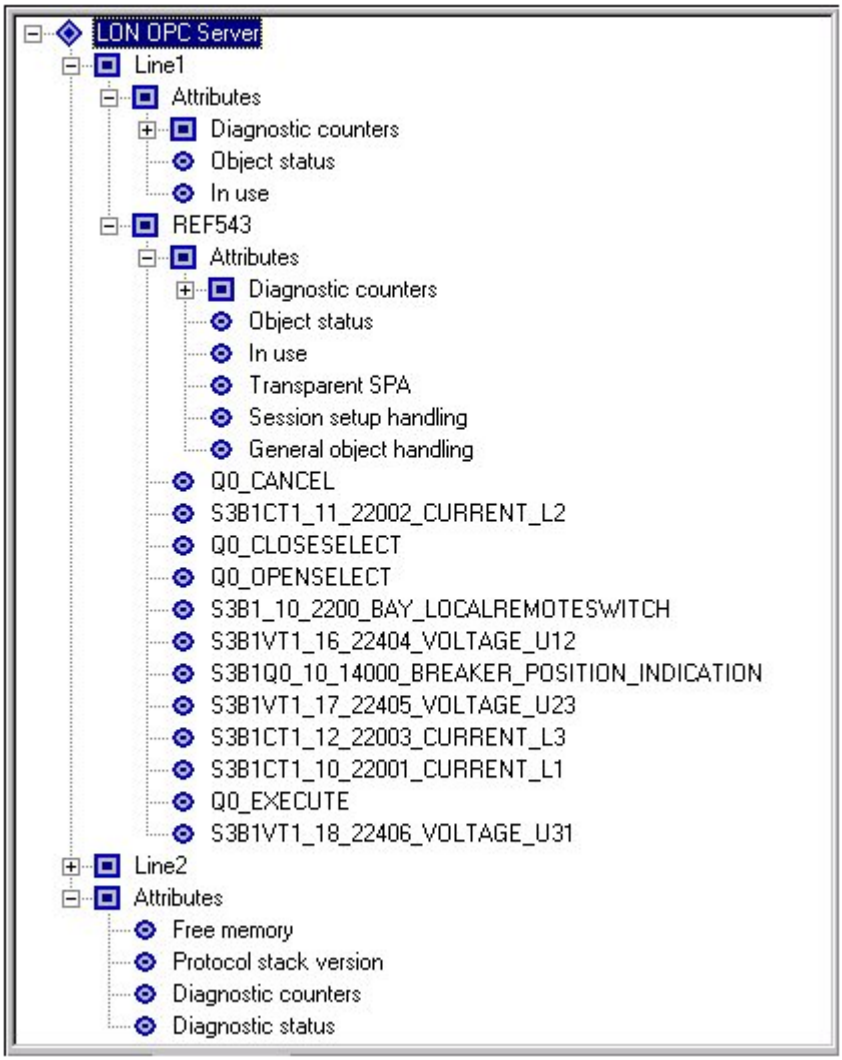


Fig. 4.-1 Software component structure of the LON OPC Server

4.1. OPC Data Access Namespace

An example of the OPC Data Access Namespace is shown in Fig. 4.1.-1. Table 4.1.-1 describes all the components shown in Fig. 4.1.-1.

Indentation is used to indicate parent-child relationship between the nodes. The LON OPC Server supports max. four lines and max. 30 units per each line.



us_namespace.tif

Fig. 4.1.-1 An example of the namespace

Table 4.1.-1 Namespace legend

LON OPC Server	The root of the namespace
Line (max. 4)	A node that represents a physical line.
Unit (max. 30 units / line)	A node that represents a physical station.
DataObject (multiple instances permitted)	An OPC item tag/node representing a data object, usually a command or an indication.
Attributes	A predefined node that contains item tags for controlling or retrieving status information for the parent node. The parent node can be the server, a line or a unit node.

Attributes

In addition to item tags for process data (indications and commands), the LON OPC Server also provides some item tags for controlling the units and retrieving status information from them. These item tags are called attributes.

There are three categories of attributes: server attributes, line attributes and station (unit) attributes. These attributes are described in sections 4.1.1.– 4.1.3.

4.1.1. Server attributes

Server attributes display information about the protocol stack. Included attributes are “Free memory”, “Version”, “Diagnostic status” and “Diagnostic counters”. The end user does not necessarily need any of these attributes.

4.1.2. Line attributes

The line attribute section contains “In use” and “Object status” tags. “Diagnostic counters” node is also included.

By using the “In use” tag, it is possible to take a line both out of use and in use. If a line is taken out of use, no events will be received from units connected to that line, neither will it be possible to send commands to units connected to it.

In order to take a line into use, write the integer 1 to “In use” tag. In order to take a line out of use, write the integer 0 to “In use” tag.

“Object status” tag displays the communication status code for a line. Status codes are defined in Appendix A. The end user does not necessarily need “Object status” tag. “Diagnostic counters” node has tags for monitoring the line communication information.

4.1.3. Station attributes

The station attribute section contains “In use”, “Object status”, “Transparent SPA”, “Session setup handling” and “General object handling” tags. It also contains “Diagnostic counters” node.

By using the “In use” tag, it is possible to take a unit both out of use and in use. If a unit is taken out of use, no events will be received from it, neither will it be possible to send commands to it.

In order to take a unit into use, write the integer 1 to “In use” tag. In order to take a unit out of use, write the integer 0 to “In use” tag.

“Object status” tag displays the communication status code for the unit. The end user does not necessarily need “Object status” tag. “Session setup handling” and “General object handling” tags are reserved for future use.

“Diagnostic counters” node has tags for monitoring the status communication information.

“Transparent SPA” tag enables a direct communication with the unit using the SPA messages. The SPA command is written to “Transparent SPA” tag, and when the answer is received it can be read from the same attribute.

The item tag for writing and reading SPA messages is found in the Attributes node for the corresponding unit. The name of the item tag is “Transparent SPA”.

An example of how to use Transparent SPA tag

Select the Transparent SPA item tag in a client and write, e.g. the following string to it:

RF:

Read the Transparent SPA item tag. If the communication is working properly, it returns the type of the unit. In this case, the type is REF543 (see below).

<50D:REF543:cr

The answer can only be read once.

4.2. OPC Alarms and Events Area Space

The area space resembles the Data Access Namespace, but instead of data objects there are alarm and event sources. The data objects that generate events are event sources in the area space. Data objects that do not generate events are not included in the area space.

The generated event type for a data object is always a condition event. The event categories used by the LON OPC Server are listed in Table 4.2.-1. They are described more precisely in sections 4.2.1. - 4.2.4.

Table 4.2.-1 Event categories and event types

Event category	Event type
System Message	simple
Operator Process Change	tracking
Discrete	condition
Trip	condition

The condition classes for the Discrete and Trip event categories must be defined in the configuration file in order to be used. The other event types do not need any configuration.

In each condition event notification the vendor specific attribute CV (= current value) is included. This attribute holds the current value of the data object that generated the event.

4.2.1. Operator Process Change (Command tracking) events

The LON OPC Server currently generates tracking events when a command is operated. The source of such an event is the full name of the command item tag. The message is currently hard-coded as "Operating".

If the command fails (the LON OPC Server gets a negative confirmation from the unit), another tracking event is issued with the same source name but with the message "Operation failed(status code number)". Status codes are defined in Appendix A.

4.2.2. System message events

The LON OPC Server generates system message events for some notifications from the units or lines. The notifications may be device_started, device_suspended, etc.

The source of such an event is the node name of the unit or line from which the notification originates. The message is "Operation failed(status code number)". For information on status codes, see Appendix A.

System message events may also be generated to inform about internal problems in the LON OPC Server, e.g. communication queues filling up.

4.2.3. Discrete (Indication) events

Discrete events describe a state change in a data object that has a fixed number of states, for example a data object that represents the position of a breaker. Discrete events have multiple sub-conditions; the number is defined in the configuration file.

4.2.4. Trip events

Trip events are connected to data objects where only the update of the value is relevant, not the actual values. Trip events are single-state conditions. They work so that the trip condition is activated (and an event generated) each time the value is updated.

A trip event could be used when a relay always sends a fixed value to a data object in order to inform about some condition.

4.3. Supported LON LAG 1.4 features

4.3.1. Network Supervision

The LON OPC Server supervises the units supporting the event session. This supervision is based on the idea that if the event session is working, then also the node is working. The event session is considered working, if the node either sends events or Acknowledgement messages. The status of the unit can be read from the Object Status OPC item of the station attribute section. An event is also generated from the OPC Alarm and Event server when the status of the unit changes.

4.3.2. Time Synchronization

The LON OPC Server can act as a time master for the following time synchronization methods: SNVT_time_stamp and NV_clock_warning - NV_clock sequence. The time synchronization mode is selected with the LK attribute of the line section in the configuration file. The line attributes are explained in detail in Table 5.1.4-1. The time for the synchronization is taken from the operating system time of the PC.

The LON OPC Server can also receive time synchronization in NV_clock_warning - NV_clock sequence format. When the LON OPC Server is configured for receiving time synchronization, it will update the operating system time of the PC.

Command Handling

The LON OPC Server supports both the IEC commands sent within event session and commands sent as transparent SPA messages.

4.3.2.1. Supported DMCD types for Process Data

- Single Command DMCD=45 (configured as IEC command type 45)

- Double Command DMCD=46 (configured as IEC command type 46)
- SI Single Indication DMCD=129 (configured as indication type 0 or 129)
- SI Single Indication with Time Tag DMCD=130 (configured as indication type 0 or 130)
- DI Double Indication DMCD=131 (reversed) (configured as indication type 0 or 131)
- Double Indication with Time Tag DMCD=132 (reversed) (configured as indication type 0 or 132)
- Measured Value Signed Integer 16 bit DMCD=135 (configured as digital value type 1 or 135)
- Measured Value Signed Integer 16 bit with Time Tag DMCD=136 (configured as digital value type 1 or 136)
- Pulse Counter Value with Time Tag DMCD=139 (configured as digital value type 1 or 139)
- DI Double Indication DMCD=143 (configured as indication type 0 or 143)
- Double Indication with Time Tag DMCD=144 (configured as indication type 0 or 144)
- Measured Value Signed Integer 32 bit DMCD=147 (configured as digital value type 1 or 147)
- Measured Value Signed Integer 32 bit with Time Tag DMCD=148 (configured as digital value type 148)
- Measured Value, Short Floating Point DMCD = 137
- Measured Value, Short Floating Point with Time Tag DMCD = 138
- Integrated Totals with Time Tag DMCD = 140
- Step Position Information DMCD = 141
- Step Position Information with Time Tag DMCD = 142

4.3.2.2. Transparent SPA

The LON OPC Server supports handling transparent SPA messages over LON. Transparent SPA tag in the station attribute section can be used for sending and receiving SPA messages.

4.4. Mapping of LON quality information to OPC item quality

4.4.1. LON quality identifiers

The quality information is taken from the quality descriptor of the indications or events. The included quality information can be different for different event types. The invalid bit (IV) in time tag information is also used.

Table 4.4.1-1 LON Identifiers and corresponding OPC Identifiers

LON Identifier	State	Used for events	OPC Identifier
IV	Valid/Invalid	All	Quality (status) good/bad
NT	Not topical	All	Quality (status) uncertain
SB	Substituted	All	Vendor Specific Quality

BL	Blocked	All	Vendor Specific Quality
OV	Overflow	Mostly analog values	Vendor Specific Quality
CY	Counter overflow	Counters	Vendor Specific Quality

4.4.2. OPC quality attribute identifiers

The OPC quality attributes are represented as a 16-bit integer. The lowest eight bits are defined by the OPC Data Access specification. The highest eight bits are available for vendor specific use.

Layout of bits 0-7: QQSSSSLL (QQ = quality, SSSS = substatus, LL = limits)

Table 4.4.2-1 Quality identifiers and their values

Quality identifier	Values
Quality (status)	Good, Uncertain, Bad
Substatus for Good	Local override
Substatus for Uncertain	Last usable value, sensor not accurate, engineering units exceeded, sub-normal
Substatus for Bad	Config error, not connected, device failure, sensor failure, last known value, comm failure, out of service
Limits	Not limited, low limited, high limited, constant

4.4.3. Mapping to vendor specific bits

The mapping adds eight **vendor** specific bits to the OPC quality attribute in addition the **standard** eight bits. VS bits layout: DDDDTBSS (DDDD = detail-quality, T = test, B = operator blocked, SS = source)

The detail-quality part supports 15 different values. Source supports four different values. Values for test and operatorBlocked are either false (0) or true (1).

Note that only **one** detail-quality value can be included in the vendor specific quality bits since all four bits reserved for the detail-quality are used to represent a detail-quality value. Detail-qualities for bad have preference over the detail-qualities for questionable.

Values for detail-quality

0 = No detail-quality specified.

1 = overFlow

2 = outOfRange

3 = badReference

4 = oscillatory

5 = failure

6 = oldData

7 = inconsistent

8..15 = Reserved for future detail-quality values.

Values for source

0 = process

1 = defaulted

2 = substituted

3 = No source specified.

Values for test

0 = false

1 = true

Values for operatorBlocked

0 = false

1 = true

5. Configuration

The purpose of this chapter is to describe both the server configuration and DCOM configuration.

In section 5.1, the configuration of the server is discussed. Section 5.2 concentrates on the DCOM configuration.

5.1. Server configuration

The Data Access Namespace, as well as the Event and Area Space must be configured before the server can be taken in use. The configuration is stored in a text file, which the server reads at start-up. The configuration file can be edited in a basic text editor.

The configuration file contains sections which describe different parts of the OPC namespaces. They can appear in any order in the file.

In the configuration file, there are definitions for the server, PCLTA-20 adapters, lines, stations, event condition classes, events and items. The required sections are the ones for PCLTA-20 adapter, line and station definitions. For example, in Fig. 4.1.-1, there are two lines and two units specified. This implies that the configuration file must also contain the sections Line1, Line2, Station1 and Station2. Because each line has to be connected to one physical communication interface, the sections LON1 and LON2 must also be included in the configuration file, although LON1 and LON2 are not visible in Fig. 4.1.-1.

The names of the sections have nothing to do with the names of the nodes in the OPC namespace or with the eventual communication line numbers. They are merely used to structure the configuration file. The node names in the namespace come from the “NodeName” key in the line and station sections.

5.1.1. Configuration file format

The server uses the configuration file opcs_net.ini, which is the standard Windows INI file. The format of INI files is defined by Microsoft in the Windows Software Development Kit. The INI files are comprised of zero or more sections. The section names are enclosed in square brackets. Within each section there may be zero or more key-item pairs, separated by an equals sign. Comments are introduced by a semicolon. Section names and key names are case-insensitive. Blank lines may appear anywhere within the file.

Syntax description using extended BNF notation:

ini-file	::= { section-definition }
section-definition	::= section-name { value-definition }
section-name	::= **[** alphanumeric-string **]**
value-definition	::= key-name **=** value
key-name	::= alphanumeric-string
value	::= alphanumeric-string

Note that the square brackets **[ ]** are not allowed in names and values. In general, only letters, digits and the underscore character _ are used in names. The item name cannot contain any of the following characters:

. , / \ \ ! | ` ´ ' # .

The values used in the LON OPC Server are:

- A number
- A string consisting of letters and digits
- A boolean
  False is either 0, false or no. True is either 1, true or yes. Boolean values are case unsensitive.
- A comma separated list of numbers and strings.

In general, only single line comments are used in INI files but end-of-line comments are also supported.

5.1.2. Server definitions

This section describes the server definitions and shows an example of the [options] section in the configuration file.

It is possible to change the behaviour of the server by changing the parameters in the [options] section. The parameters are in general simple constants of boolean, integer or string type.

Table 5.1.2-1 Server definitions

Name	Data type	Default value	Description
UseReversedDoublePositions	Boolean	False	Specifies whether the ON and OFF values are reversed in double position indications or not. When this option is True OFF = 2 and ON = 1.
EnableOPCv2Optimizations	Boolean	False	Enables faster and less memory consuming connections for the OPC v2 clients. **NOTE!** The OPC v1 clients are not fully supported when this option is enabled.
PCLTAConfChanged	Boolean	False	Specifies whether the PCLTA-20 adapter configuration has been changed. True value will launch card configurator program, which will do necessary changes, checks and turn value to false if operation was successful.

If the option parameter is not included in the configuration file, the default option value is used.

An example of how to define a server

```
[options]
EnableOPCv2Optimizations=1
UseReversedDoublePositions=1
PCLTAConfChanged=1
```

The formats of the other sections are shown in sections 5.1.3.-5.1.6. For this purpose, the example sections are used instead of a formal syntax.

5.1.3. PCLTA-20 adapter definitions

This section describes the PCLTA-20 adapter definitions in the configuration file. An example PCLTA-20 definition is shown below.

Table 5.1.3-1 PCLTA-20 adapter definitions

Name	Data type	Value	Description
SN	Integer	1...127	Subnet number of the communication channel of PCLTA-20 adapter
NN	Integer	1...127	Node number of the communication channel of PCLTA-20 adapter

An example of how to define PCLTA-20

```
[LON1]
SN=2
NN=125
```

The PCLTA-20 adapter is used as a physical communication interface between the LON OPC Server and the system. For each physical LON interface (LON1, LON2, etc.) there has to be a corresponding [LONx] section, which defines the node and the subnet address of the interface.

The system considers the PCLTA-20 adapter as a node. Each node in the system is identified with a unique subnet/node address, which are ‘sn’ and ‘nn’ key values of [LONx] section. If the PCLTA configuration program fails to operate due to some reason, the start procedure of the LON OPC Server will be aborted.

5.1.4. Line definitions

This section covers the line attributes in the configuration file. An example line definition is shown below, and Table 5.1.4-1 provides more detailed information on the attributes.

An example of how to define a line

```
[Line1]
Linenr=1
Protocol=27
NodeName=Line1
SD=LON1
LK=3
PS=200
```

Table 5.1.4-1 Line definitions

Name	Data type	Value	Description
Linenr	Linenr has to get a unique value between 1...4		
Protocol	Protocol defines the type of the protocol. The value is always 27.		
NodeName	NodeName is the name the line has in the OPC namespace. The value cannot contain any of the characters listed in 5.1.6.		

Name	Data type	Value	Description
SD (System Device name)	Text	The device name of the communication interface	The device name of the physical communication interface. Each physical connection of the PCLTA card (each LONWORKS channel) is associated with a specific device name. If the Echelon device driver is used, the device name for first channel is "LON1" and for second channel "LON2" etc.
PS (Buffer Pool Size)	Integer	1...250 Suggested: 200...250	The number of message buffers reserved for the line. Each buffer can contain one message.
LK (Link Type)	The clock synchronization of LONWORKS lines. The value in the attribute specifies clock synchronization as follows: 0 No clock sync 1 Clock synchronization based on NV Warning and NV Clock telegrams once per second. 2 Clock synchronization based on NV Time telegram once per minute. 3 Clock synchronization based on both methods: NV Warning, NV Clock and NV Time telegram 4 Receive NV Warning and NV Clock telegrams, synchronizes the operating system time according the received sync 5 Receive NV Time telegram, synchronizes the operating system time according the received sync It is recommended to use NV Time method only when the other synchronization methods do not work, or when the exact time is not needed because of the inaccuracy on high channel load on LON line with minute pulse.		

5.1.5. Station definitions

This section describes the definitions for the station (unit). An example station definition can be seen below. All the station definitions are listed in Table 5.1.5-1, where the attributes are divided into basic attributes and advanced attributes.

An example of how to define a station

```
[Station2]
Linenr=1
Stationnr=2
Devicetype=17
NodeName=STA2
NN=9
SN=1
UN=9
RM=7
```

Table 5.1.5-1 Station definitions

Basic attributes			
Name	**Data type**	**Value**	**Description**
Linenr	Linenr has to get a value between 1...4 (where 1...4 stand for the configured lines in the system)		
Stationnr	Stationnr is the number of the station		
Devicetype	Devicetype is the type of the device. The value is always 17.		
NodeName	NodeName is the name the station has in the OPC namespace		
NN (Node Number)	Integer	1...127	Node number of the station.
SN (Subnet Number)	Integer	1...127	Subnet number of the station

UN (Unit Number)	Integer	0...65535	Unit number used in transparent SPA messages (both messages resulting from SPA commands and messages generated with the Transparent SPA attribute)
RM (Running Mode)	Integer	Default: 7	Determines the running mode of the station communication. Should always be set to 7, which means event session based communication.
EF (Event Filter Number)	Integer	0...127 Default: 0	Filter number for event sessions.
RT (Reply Timeout)	Integer	0...655 Default: 20 (seconds)	The maximum time in seconds that communication software waits for reply from the REX unit when sending commands and transparent SPA messages.
Advanced attributes			
Name	**Data type**	**Value**	**Description**
SC (Session Nack Timeout)	Integer	1...60000 Default: 750 (ms)	The timer (Terr) for controlling the cyclic sending of NACK after a message sequence error. This timer is active only when the network congestion occur, and should be a bit less than the retransmit timer (Retr).
SI (Session Idle Timeout)	Integer	1 ... 60000 Default: 10000 (ms)	The idle ACK message interval timer (Tidle) is used to keep channel alive. It also re-transmits ACK messages in case of ACK loss. In that situation the flow will be driven by the retransmission timer. The Session Idle Timeout needs to be smaller than the Session Keepalive Timeout (SK).
SK (Session Keepalive Timeout)	Integer	1...60000 Default: 60000 (ms) Suggested: 10000...60000	The connection timer (Tconn) that supervises the operation of the remote node. On the idle channel both of the transmission partners send frequently so called keepalive messages. This transmission should happen in the range of 1 minute. Otherwise the connection timeouts.
SR (Session Retransmit Timeout)	Integer	1...60000 Default: 5000 (ms)	The retransmit timer (Tretr) is used to trigger a retransmission of the unacknowledged message if the message or ACK / NACK was lost. The Session Retransmit Timeout should be greater than the time to send a full window (max Credit).
SS (Session in Sequence Response Delay)	Integer	1...60000 Default: 300 (ms)	The time that the receiver of the message waits before responding. The timer is activated after every received message. If the channel is idle the timer will timeout. During obstruct of traffic the sender will lose the Credit and flag the message for immediate ACK (TranAck flag). In such circumstances the Tseq timer will not expire.
RQ (Receive Quota)	Integer	1...10 Default: 10	Receive quota for the station. Generally, the default value is suitable.
TQ (Transmit Quota)	Integer	1...10 Default: 10	Transmit quota for this device. Generally, the default value is suitable.

5.1.6. Data object definitions

Data objects are used to define the process data of LON device, which will be accessible via the LON OPC Server.

The LON OPC Server supports different data types, e.g. single/double indications, digital/analog inputs, pulse counters, commands and SPA parameters.

The definition of the data object consists of the item name, data type and LON/SPA data address. The item name is the name that is visible in the OPC server namespace

for the data object. It can be freely set by the user to describe the data indentity. The item name must be unique within one unit.

The data type should be set according to the type of the process signal. The LON/SPA address defines the LON IEC or SPA address of the data. The addresses should be set according to the configuration of the LON device used.

Data objects (item tags) are defined in a section with the name ItemDefs_Station(n), where n is the section number of the unit that the items are defined for. The syntax for defining a data object in the configuration file can be seen below:

itemname = Data Type, LON/SPA Address

5.1.6.1. Single/Double Indication

Single and double indications are defined with data type 0. The following LON DMCD types can be used for updating the value:

- Single Indication DMCD=129
- Single with Time Tag Indication DMCD=130
- Double Indication DMCD=131 (reversed)
- Double Indication with Time Tag DMCD=132 (reversed)
- Double Indication DMCD=143
- Double Indication with Time Tag DMCD=144

Single Indications have value 0 for OFF and 1 for ON. Double Indications have value 0 = Intermediate position, 1 = OFF, 2 = ON, 3 = faulty position. The open and closed values can be reversed with UseReversedDoublePositions attribute in the server section of the configuration file.

An example of Single/Double Indication

Bay1_Breaker_Q0_Position = 0, 14100

5.1.6.2. Digital Input

Digital input values are defined with data type 1. The following LON DMCD types can be used for updating the value:

- Measured Value Signed Integer 16 bit DMCD=135
- Measured Value Signed Integer 16 bit with Time Tag DMCD=136
- Measured Value Signed Integer 32 bit DMCD=147
- Measured Value Signed Integer 32 bit with Time Tag DMCD=148
- Step Position Information DMCD = 141
- Step Position Information with Time Tag DMCD = 142

An example of Digital Input

T1_Tap_Changer_Position = 1, 4700

5.1.6.3. Analog Input

Analog input values are defined with data type 2. The following LON DMCD types can be used for updating the value:

- Measured Value, Short Floating Point DMCD=137

- Measured Value, Short Floating Point with Time Tag DMCD=138
- Measured Value Signed Integer 16 bit DMCD=135
- Measured Value Signed Integer 16 bit with Time Tag DMCD=136
- Measured Value Signed Integer 32 bit DMCD=147
- Measured Value Signed Integer 32 bit with Time Tag DMCD=148

An example of Analog Input

Current_L1 = 2, 22001

5.1.6.4. Pulse Counter

Pulse counter values are defined with data type 6. The following LON DMCD types can be used for updating the value:

- PCV with Time Tag DMCD=139
- Integrated Totals with Time Tag DMCD=140

An example of Pulse Counter

Energy_Pulses_L1 = 6, 3506

5.1.6.5. SPA Commands

Binary output SPA commands are defined with data type 10. These are used, for example, to control a breaker with REF543 relay. The SPA address is given in a format:

SPA channel 1, SPA channel 2, Data Category, Data Number 1, Data Number 2, Data Format

SPA channels 1 and 2 define the SPA channel number where the command is given, normally channel 1 and channel 2 are the same. Data Category defines the SPA data category (I,S,V,O); normally V is used for commands. Data Number defines the SPA data number; normally data number 1 and 2 are the same.

Data format defines the format of the data value, 3 is equal to binary value (0 or 1). This is the only format supported at the moment. The syntax for defining SPA commands in the configuration file can be seen below:

itemName = Type, Ch1, Ch2, Cat, Da1, Da2, Format

An example of SPA Commands

Q0_Breaker_OpenSelect = 10, 120, 120, V, 6, 6, 3

Q0_Breaker_CloseSelect = 10, 120, 120, V, 7, 7, 3

Q0_Breaker_Execute = 10, 120, 120, V, 11, 11, 3

Q0_Breaker_Cancel = 10, 120, 120, V, 10, 10, 3

5.1.6.6. IEC Commands

IEC style of commands are defined with data type 45 (Single Command DMCD=45) or 46 (Double Command DMCD=46).

IEC commands are somewhat different from the other item tags. Instead of a tag with the name in the definition, a node with the same name is created. This node in

turn contains tags to control the IEC command. The tags correspond to the commands Select off, Select on, Operate off, Operate on and Cancel. The names of the tags are as follows:

ctlSelOff
ctlSelOn
ctlOperOff
ctlOperOn
ctlCan

An example of IEC Commands

Q0_Breaker_Control_SC = 45, 1005

Q1_Breaker_Control_DC = 46, 1006

5.1.6.7. SPA Parameters

It is possible to configure item tags for SPA parameters. These tags can be used to read and write parameters in the relay which are only accesible via transparent SPA.

This is an alternative method for Transparent SPA communication, which uses a Transparent SPA tag in the unit (station) attributes section. This is presented in section 4.1.3. Configuring SPA parameters as separate item tags makes it possible to access the parameters from an OPC Client that cannot interpret transparent SPA messages.

The value of a SPA parameter item tag is read from the unit when the client reads the item first time. After that a cached value is returned unless the client requests a read or refresh from the unit. The syntax for defining SPA parameters in the configuration file can be seen below:

item = SPA_PARAM, vartype (see below), access (R=read, W=write, RW (=WR)), SPA parameter (string)

Example of SPA Parameters:

spa_rw_r4 = SPA_PARAM, 4, RW, 0S1

spa__w_r8 = SPA_PARAM, 5, W, 999V999

spa_r__ui4 = SPA_PARAM, 19,R, 0S999

spa_rw_i4 = SPA_PARAM, 3, WR, 124S1

spa_rw_bstr = SPA_PARAM, 8, RW, 12S4

VARTYPE values

Basically, any vartype could be used (the server does not check the value in any way) but it is **strongly recommended** to use one of the vartypes listed below:

VARTYPE	Numeric value	Comment
VT_I2	2	Signed Integer 2 bytes (16bit)
VT_I4	3	Signed Integer 4 bytes (32bit)
VT_R4	4	Real 4 bytes (32bit)
VT_R8	5	Real 8 bytes (64bit)

VT_BSTR	8	String
VT_BOOL	11	Boolean
VT_DECIMAL	14	16 byte fixed point.
VT_I1	16	Char 1 byte (8bit)
VT_UI1	17	Unsigned Char 1 byte (8bit)
VT_UI2	18	Unsigned Integer 2 bytes (16bit)
VT_UI4	19	Unsigned Integer 4 bytes (32bit)
VT_I8	20	Signed Integer 8 bytes (64bit)
VT_UI8	21	Unsigned Integer 8 bytes (64bit)
VT_INT	22	Signed Integer (machine dependent)
VT_UINT	23	Unsigned Integer (machine dependent)

Vartype VT_BSTR is a string, vartype VT_BOOL is a boolean and the rest of the vartypes are numbers. The SPA parameter in the relay, which the SPA paramater item tag is connected to, should be of an equivalent type. The most commonly used types are probably VT_I4, VT_UI4, VT_R4, VT_R8 and VT_BSTR.

5.1.7. Event configuration

As mentioned in section 4.2., condition type events must be configured before they can be used.

This configuration is done in two steps. Firstly, a condition class is defined. Then the data objects, which generate condition events of this condition class, are connected to it.

5.1.7.1. Event condition class definitions

The format of the condition class definition depends on the event category for the condition. In general, a condition class definition defines the name, description, severity and acknowledge policy for conditions or sub-conditions. The basetype (event category) is of course also included in the definition.

For Discrete (Indication) condition classes the definition specifies the number of sub-conditions. For each sub-condition the description (message), severity and name are specified. It must also be specified whether the sub-condition must be acknowledged or not. The name of the basetype is DiscInd.

Format of Discrete Condition Class definition:

ConditionClassName = DiscInd, NumberOfSubConditions,
ConditionDescription1, Severity1, ConditionName1, AckRequired1,
ConditionDescriptionN, SeverityN, ConditionNameN,AckRequiredN

Trip conditions are single-state. Thus, the only information required in the definition of the condition class is the description (message) and severity. It must also be defined whether an acknowledgement is required or not. The name of the basetype is Trip. An example can be seen below:

Format of Trip Condition Class definition:

ConditionClassName = Trip, Description, Severity, AckRequired

An example can be seen below:

```
[EventConditionClassDefs]
Position2 = DiscInd, 2, Open, 1, Open, FALSE, Closed, 1, Closed, FALSE
Position4 = DiscInd, 4, Faulty, 1, Faulty, FALSE, Open, 1, Open, FALSE,
            Closed, 1, Closed, FALSE, Intermediate, 1, Intermediate, FALSE
TripEvent = Trip, TRIPDesc, 100, TRUE
TripEventNoAck = Trip, TRIPNoAckDesc, 1, FALSE
```

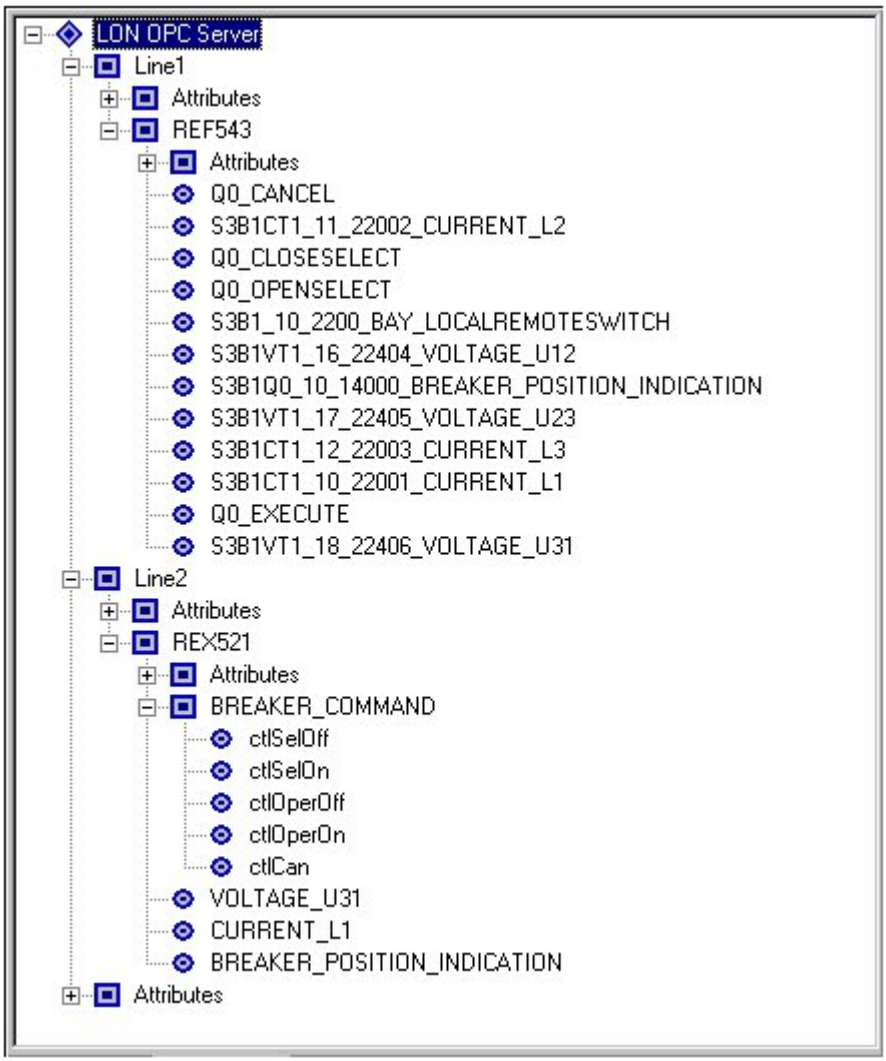
5.1.7.2. Event definitions

Condition classes are connected to data objects in the eventdefs section which corresponds to the itemdefs section where the data objects were defined in. The name of the section is the same as the itemdefs section, except that it starts with “EventDefs” instead of “ItemDefs”.

The content of the section simply connects item tags with a condition class (itemname = condition class). The format is shown below.

```
[EventDefs_Station2]
Ind1001  = Position2
Ind1002  = Position4
Trip534  = TripEvent
```

5.1.8. An example configuration of a server



Station.tif

Fig. 5.1.8.-1 An example configuration of a server

```
[options]
EnableOPCv2Optimizations=1
UseReversedDoublePositions=1
PCLTAConfChanged=1

[LON1]
SN=1
NN=127

[LON2]
SN=2
NN=127
```

```
[Line1]
Nodename = Line1
Linenr=1
Protocol=27
SD=LON1
LK=0
PS=250

[Line2]
Nodename = Line2
Linenr=2
Protocol=27
SD=LON2
LK=0
PS=250

[Station1]
Linenr=1
Stationnr=50
Nodename=REF543
Devicetype=17
NN=50
SN=1
UN=50
RM=7

[Station2]
Linenr=2
Stationnr=51
Nodename=REX521
Devicetype=17
NN=51
SN=2
UN=51
RM=7

[ItemDefs_Station1]
S3B1CT1_10_22001_Current_L1=2,22001
S3B1CT1_11_22002_Current_L2=2,22002
S3B1CT1_12_22003_Current_L3=2,22003
S3B1VT1_16_22404_Voltage_U12=2,22404
```

S3B1VT1_17_22405_Voltage_U23=2,22405
S3B1VT1_18_22406_Voltage_U31=2,22406
S3B1_10_2200_Bay_localremoteswitch=0,2200
S3B1Q0_10_14000_Breaker_position_indication=0,14000
Q0_OpenSelect = 10,120,120,V,6,6,3
Q0_CloseSelect = 10,120,120,V,7,7,3
Q0_Cancel = 10,120,120,V,10,10,3
Q0_Execute = 10,120,120,V,11,11,3

[EventDefs_Station1]
S3B1_10_2200_Bay_localremoteswitch=Position4
S3B1Q0_10_14000_Breaker_position_indication=Position4

[ItemDefs_Station2]
Current_L1=2,22001
Current_L2=2,22002
Current_L3=2,22003
Voltage_U12=2,22404
Voltage_U23=2,22405
Voltage_U31=2,22406
Bay_localremoteswitch=0,2200
Breaker_position_indication=0,14000
Breaker_Command = 46, 14071

[EventDefs_Station2]
Bay_localremoteswitch=Position4
Breaker_position_indication=Position4

[EventConditionClassDefs]
Position4 = DiscInd, 4, Faulty, 1, Faulty, FALSE, Open, 1, Open, FALSE, Closed, 1, Closed, FALSE, Intermediate, 1, Intermediate, FALSE

In the example configuration above, two lines (Line1 and Line2) and one unit for each line are configured. Each line has to be connected to one physical communication interface. In this configuration, Line1 is connected to LON1 and Line2 to LON2, which can be seen by checking the value of the SD attribute in line section. REF543 is under Line1 and REX521 is under Line2. This can be seen by checking the Linenr attributes in the line and unit sections.

Because REF543 uses SPA commands, it has four command points (itemtype = 10), some measurements, e.g. analog inputs (itemtype=2) and few binary points, e.g. indications (itemtype=0).

Instead, REX521 uses IEC commands. In the example above, you can see how the command is configured with only one point (itemtype=46). The same command can

also be seen in Fig. 5.1.8.-1 (Breaker Command for REX521). Other points are as in REF543. Event mappings are defined in EventDefs_Stationx -sections (item=eventcondition), although it is not visible in Fig. 5.1.8.-1.

5.2. DCOM configuration

By default, the OPC Server is installed and used as a local server, i.e. both the server and the client run on the same computer. The server can also be run as a remote server on another computer than the client. The remote server is accessed in the same way as the local server (except that the computer where the server runs must be selected in the OPC client; how this is done is client-specific), but some DCOM configuration must be performed before accessing the remote server. The DCOM configuration must be done on both the server and the client computer.

The server must naturally be installed and registered on the server computer. It must also be registered on the client computer unless the client uses the OPC Server Browser (OPCENUM.EXE) provided by the OPC Foundation or unless the client can browse the registry on the server computer. The registration is done automatically during installation, therefore it may be easiest to install the server on the client computer as well if the client requires local registry entries for the server.

DCOM uses security settings to protect client and server from unauthorized access. The actual settings that are used depend on the security requirements, but the settings below should work in most cases. The settings can be changed with the program dcomcnfg. Administrator rights are required in order to use it.

You can launch the dcomcnfg program by choosing Start > Run. When prompted for the name of a program, type “dcomcnfg“ and click OK (see Fig. 5.2.-1).

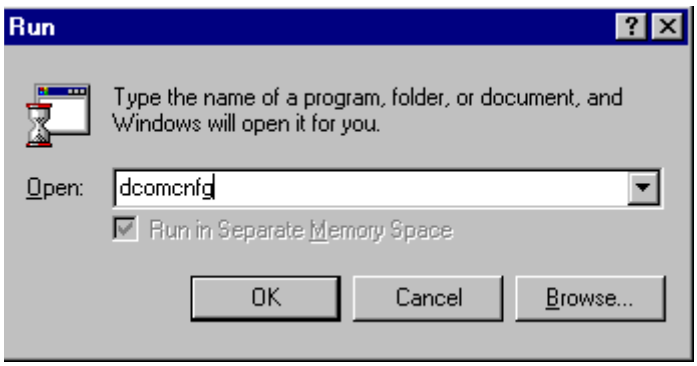
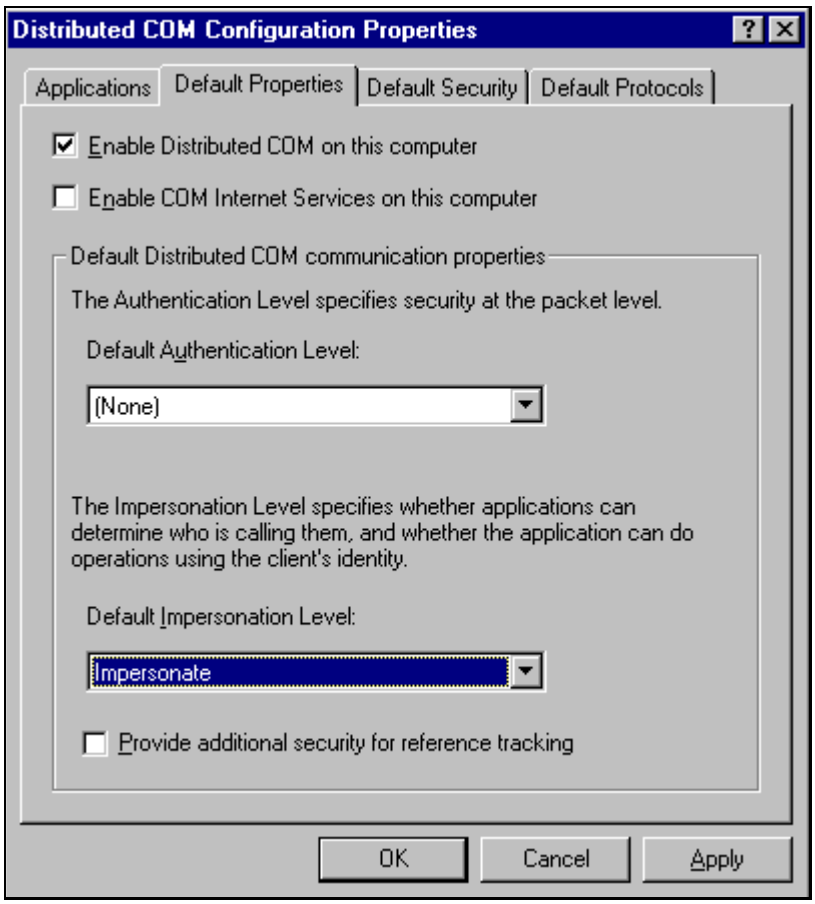


Fig. 5.2.-1 Launching the dcomcnfg



dcomconfig.tif

Fig. 5.2.-2 An example of DCOM configuration

More information on DCOM is available in Microsoft Developer Network (MSDN).

5.2.1. Server computer

Default properties for the server computer are listed below:

- Make sure DCOM is enabled
- Set default authentication level to "None"
- Set default impersonation level to "Impersonate"

Make sure that the Default Configuration Permissions include SYSTEM.

Specify custom security for OPCEnum (needed if remote OPC Server browsing is used in the OPC Client).

- Set custom access to NETWORK, SYSTEM, Everyone
- Set custom launch permissions to INTERACTIVE, SYSTEM, NETWORK, Everyone

Specify custom security for the OPC Server.

- Set custom access to NETWORK, SYSTEM, Everyone
- Set custom launch permissions to INTERACTIVE, SYSTEM, NETWORK, Everyone

Set the identify for the OPC Server to "Interactive User" or configure it to run in a specific user account (see Local DCOM configuration).

5.2.2. Client computer

Default properties for the client computer are listed below.

- Make sure DCOM is enabled
- Set default authentication level to "None"
- Set default impersonation level to "Impersonate"

Before starting a remote LON OPC Server, be sure that the LON OPC Server computer and the OPC Client computer can access each other on the network. The "Remote Procedure Call" service should be started on the LON OPC Server computer.

5.2.3. Local DCOM configuration

Sometimes it is necessary to configure DCOM even when the server and client run on the same computer. This is the case when two clients that are running in different user accounts must both access the same server. For example, an alarm and event client running as a service and an interactive data access client.

If the server is configured to run as the "Launching User", only the client that connects first will succeed to get a connection. This is due to how COM works. The solution is to configure the server to run either as the "Interactive User" or in a specific user account ("This user:").

If the server is configured to run in a specific user account, its user interface will not be visible when it is activated by the COM runtime. This is because it is not started in the interactive desktop. The only place where it is visible is the Task Manager.

The server should not be started manually, if it is configured in this way. In that case, the clients that are running in other user accounts will not be able to access it although it is correctly configured.

If the server is configured to run as the "Interactive User" it will be terminated when the interactive user logs off. This is not desirable, for example, if the client is an alarm service that is always running.

Index

T

U

V

Appendix A

Status codes

0	DEVICE_OK
13213	INVALID_ATTRIBUTE_VALUE
13214	CHAR_TYPE_EXPECTED
13222	NO_ACKNOWLEDGE_RESPONSE
13225	ONLY_WRITE_IS_POSSIBLE
13226	NO_ACKNOWLEDGE_REPLY
13272	DEVICE_STOPPED
13273	DEVICE_STARTED
13274	DEVICE_HALTED
13275	OUT_OF_BUFFERS
13276	TIMEOUT_WHILE_WAITING_RESPONSE
13277	DEVICE_CONNECTING
13278	DISCONN_IND_RECEIVED
13279	CONN_CON_RECEIVED
13280	EVENT_SESSION_REJECTED
13281	EVENT_SESSION_SETUP_TIMEOUT
13282	RUNNING_MODE_OPERATION_CONFLICT
13283	DEVICE_WAITING_GO_COMMANDS
13284	SETUP_SESSION_REJECTED
13285	SETUP_SESSION_TIMEOUT
13311	SC_DATA_OVERFLOW
13312	LON_ADDRESS_NOT_CONFIGURED
13313	DEVICE_MUST_BE_ALLOCATED
13314	UNKNOWN_DIAGNOSTIC_COUNTER
13315	INVALID_INDEX_RANGE
13316	ILLEGAL_APPLICATION_FOR_OPERATION
13317	ATTRIBUTE_IS_WRITE_ONLY
13318	INVALID_ATTRIBUTE_VALUE
13319	CHAR_TYPE_EXPECTED
13320	INTERNAL_ERROR
13321	INVALID_POINT_DEFINITION
13322	TOO_MANY_ARGUMENTS
13323	ARGUMENT_EXPECTED
13324	UNABLE_TO_ALLOCATE_MEMORY
13325	POINT_DEFINITION_NOT_FOUND
13326	NO_ACKNOWLEDGE_RESPONSE
13327	UNEXPECTED_VALUE_TYPE
13328	ILLEGAL_OBJECT_TYPE
13329	ONLY_WRITE_IS_POSSIBLE
13330	NO_ACKNOWLEDGE_REPLY

13331	NO_SM_REPLY_AVAILABLE
13332	OUT_OF_BUFFERS
13333	SM_ATTR_ACCESS_VIOLATION
13334	PARAMETER_TRANSFER_RESERVED
13335	PARAMETER_TRANSFER_RELEASE_DENIED
13336	ILLEGAL_BROADCAST_ACTION
13337	ILLEGAL_IEC_EVENT_RECEIVED
13338	GI_ACTIVATION_CONF_RECEIVED
13339	GI_TERMINATION_RECEIVED
13340	UNABLE_TO_CONVERT_FROM_SC_COMMAND
13341	INVALID_RT_ATTRIBUTE_VALUE
13342	UNKNOWN_LON_ATTRIBUTE
13355	MAX_NACK_LIMIT_EXCEEDED
17550	OPEN_FAILED_WITH_INVALID_HANDLE
17551	CLOSE_FAILED
17552	INVALID_ATTR_VALUE
17553	INVALID_ATTR_INDEX
17554	INVALID_LINE_NUMBER
17555	LINE_DOES_NOT_EXIST
17556	NV_INDEX_NOT_CONFIGURED

Customer feedback

About this chapter

This chapter contains information on how to send customer feedback.

Customer feedback database

Customer Feedback is a Lotus Notes database which ABB companies can use to report errors, make improvement proposals and queries related to products manufactured by ABB Substation Automation Oy. The Customer Feedback database is connected to the change management system of ABB Substation Automation Oy, which handles all error corrections and improvements made to the products.

Please note that the Customer Feedback database is primarily intended for writing reports about released products. If you are using, for example, a beta release in a pilot project, this should be clearly stated.

Writing a customer feedback report

When writing a Customer Feedback report the following general instructions should be taken into consideration:

- Write the report in English.
- Write only one error report, query or improvement proposal in a Customer Feedback Report.
- If you are reporting an error, try to isolate the error as well as possible. Describe the sequence of events and actions causing the error. If any error messages or other debug information is provided by the system, please write it down. Include also information of the system, e.g. a system diagram, revision information and configuration data.
- If you are making an improvement proposal, try to describe how the improved function should work. Avoid providing solutions. Information about the importance of the improvement, e.g. number of projects that require the improvement, helps us to make the decision whether and when the improvement should be implemented.

To make a Customer Feedback Report, select Feedback Report from the Create menu. This opens an empty Customer Feedback document. Fill out the fields listed below. A question mark next to a field provides help for filling out the field.

1 Subject. This should contain a short description of the issue. A more detailed description can be given in the Description of Feedback field below.

2 The type of Feedback: Comment/Improvement, Query or Complaint/Error.

3 Customer Information.

4 Reporting Information. This should contain detailed information about the product in question.

5 The person who you want to send the feedback to and whether you want to get a reply from that person or not.

6 Information related to internal handling of the report (not obligatory).

7 Category.

You can issue the report by clicking the Issue Feedback button. This will send the report to the selected person and change its status to “in progress”.

Actions

When ABB Substation Automation Oy receives a Customer Feedback report it is analysed by a sales person or a representative of the technical support. The analyser may ask for additional information in order to complete the analysis. After the report has been analysed, the following actions will be taken:

- In case of a clear error the report will be moved to the change management system of ABB Substation Automation Oy. In this system the error will be analysed in detail and corrected in a future patch release or major release depending on the severity and impact of the error.
- In case of an improvement proposal the report will also be moved to the change management system where it will be considered as a requirement for future releases.
- In case of a query an answer will be provided.

When Customer Feedback reports are handled in the change management system, the outcome can be one of the following:

No Actions	This means that it is decided that the report requires no further action. For example, if the problem is caused by a configuration error, it belongs to this category.
Will be implemented in patch/current release	This means that the correction or a new feature will be available in the next official program release.
Moved to future release	This means that the new feature will be available in a new program release in the near future.

Ordering information

LON OPC Server

LON OPC Server version 1.0.0 software CD is ordered according to the price and conditions stated in the price list from ABB Substation Automation Oy.

The ordering number is 1MRS151021.

The LON OPC Server version 1.0.0 software CD has the following content:

- LON OPC Server v.1.0.0
- LON OPC Server Installation and Commissioning Guide
- Adobe Acrobat Reader

RER107

RER 107 can be supplied with glass or plastic fibre optic transceivers.

Type designation for RER 107:

RER 107 MM,	ST-type glass fibre optic transceivers	1MRS090702-MM
RER 107 BB,	snap-in type plastic fibre optic transceivers	1MRS090702-BB
RER 107 SS,	SMA-type plastic fibre optic transceivers	1MRS090702-SS

CAP 505 v1.0.2. Relay Product Engineering Tool Box
Installation and Commissioning Manual 1MRS750537-RUM

LNT v.1.0.1. Lon Network Tool
Installation and Commissioning Manual 1MRS750830-RUM

For more information, see:

RER 107	A	1MRS 750610-MUM
Transceiver module		
User's manual and Technical description		

PCLTA-20 LonTalk adapter

Please, note that the PCLTA-20 adapter is not available form ABB Substation Automation Oy. The adapter can be purchased from Echelon Corporation. More information is available on the web site http://www.echelon.com.

PCLTA-20 card features:

- 32 bit PCI, SMX card for LONWORKS networks (made by Echelon)
- Windows NT 4.0 or Windows 2000 driver from the manufacturer