

NLOPCTE MANUAL



NEWRON SYSTEM

25-27 Boulevard Victor HUGO
31770 COLOMIERS (France)

T: +33 (0)5 61 15 18 45

F: +33 (0)5 61 15 16 44

SUMMARY

Introduction	5
NL220.....	5
NLFacilities.....	5
NLOPC-MIP.....	5
NLopcTE.....	5
Installation of programs	6
Configuration requirements.....	6
Installation.....	6
Installation of program.....	6
Network interface	9
Valid interfaces.....	9
Verifying interface function.....	9
Verifying PCC10, PCLTA21.....	9
Verifying SLTA10.....	10
Verifying LPP.....	11
Verifying Loytec Interface.....	11
Features of NLOPCTE	13
Introduction.....	13
LNS compliant.....	13
Normal features.....	13
Configuration.....	14
Advanced features.....	14
Network Variables Monitoring.....	14
Devices Monitoring.....	15
Configuration Properties Monitoring.....	15
Formatted values:.....	15
Limited version:.....	16
Non Standard OPC Errors.....	16
NLOPCTE configuration	17
Introduction.....	17
Configuring NLOpTE (Mandatory).....	17
Configuring the Modes.....	17
Configuring the LNS Databases to monitor.....	18
Configuring the Tuning options.....	18

Configuring the Fast Polling engine (used only if Fast Polling mode is active).....	19
Saving your configuration.....	20
Other options	20
Advanced Configuration.....	21
Advanced configuration : Tuning and Options	21
Advanced configuration : Advanced settings	23
Advanced configuration : Fast polling	27
NLOPCTE items ADDRESS	29
About OPC Addresses.....	29
Address for Network Variables.....	29
Formatting	29
Format of the address of a network variable	30
Address for Configuration Properties.....	32
Formatting	32
Format of the address of a configuration property.....	32
Address for Devices (application or router).....	35
Formatting	35
Format of the address of a device	35
MultiField Items.....	36
What is a MultiField item ?	36
How to address a MultiField item ?.....	36
What is the format of the value of a MultiField item ?.....	36
Using aliases for item addresses.....	37
What are aliases ?	37
How to define aliases ?	37
Using an alias as a full address.....	37
Using an alias as a part of an address.....	37
Special Items.....	38
NLOPCTE User interface	40
Human interface	40
The toolbar	40
The connected OPC clients and OPC groups.....	42
The OPC items of an OPC group	42
Configuring the OPC items grid.....	43
Changing value in Simulation mode	45
The traces	47
Configuring the traces.....	47
First use with an OPC client	52
Softing OPC client.....	52
STEP 1: Configuring NLOpcte.....	52
STEP 2: Starting the Softing OPC client.....	52
STEP 3: Browsing items and adding one item	54
STEP 4 : More operation in Softing OPC client.....	54
STEP 5: Close OPC Client.....	55

PICTURES

Picture 1 Step 1	7
Picture 2 Step 2	7
Picture 3 Step 3	7
Picture 4 Step 4	7
Picture 5 Step 5	7
Picture 6 Installation Step 6.....	8
Picture 7 Install Step 7	8
Picture 8 Installation Step 8.....	8
Picture 9 Install Step 9	8
Picture 11 SLTA10 connected.....	11
Picture 12 SLTA10 unconnected.....	11
Picture 13 Valid LPP windows.....	11
Picture 14 Human interface of NLopcTE server.....	40
Picture 15 Connected OPC clients and existing OPC groups.....	42
Picture 16 Configuring the OPC items grid	43
Picture 17 Simulate the value of a device	45
Picture 18 Simulate a raw value	45
Picture 19 Simulate a formatted value	46
Picture 20 Traces.....	47
Picture 21 Configuring the traces	48
Picture 22 LNS Database for example.....	52
Picture 23 Softing OPC Client User Interface	53
Picture 24 Browsing the OPC servers	53
Picture 25 Softing Client Connected to NLopcTE	54
Picture 26 Browsing items.....	54
Picture 27 Item added to the group	54

INTRODUCTION

Thank you for choosing **NLopcTE** software member of NLSuite.

We are happy to help you in your LonWorks integration. All software of NLSuite are often updated to correct bugs and improve performances. Please check version on Web site www.newron-system.com.

NL220TE

This is a LonWorks LNS Manager tool.

NLFacilitiesTE

This is a graphical tool for managing your living spaces.

NLOPC-MIP

This is a very fast OPC server with embedded tool to tune your Scada.

NLopcTE

It is a OPC server for LNS Turbo Edition. It can manage directly iLon interfaces.

NLUti

This is a window node utility. It is used before installation for checking channel and other LonWorks products.

INSTALLATION OF PROGRAMS

This section explains how to install the NLOPC-VNI program

Configuration requirements

The table below shows the minimum configuration and the recommended configuration for the installation and correct functioning of the program.

Equipment	Minimum	Recommended
Operating system	Windows 2000, Xp, 2003 server	Windows 2000, Xp, 2003 server
Computer	Pentium IV 1Ghz 800 x 600 screen	Pentium IV 2Ghz 1024 x 768 screen
Memory	256 M octets	512 M octets
Hard disk	1Go	1Go
CD ROM	Required for installation	Required for installation
Software	LNS Turbo Edition 3.21 or greater	LNS Turbo Edition 3.21 or greater
Interface network	Type NSI or VNI card	Type NSI or VNI card

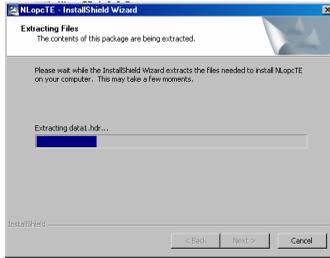
Table 1 The equipment

Installation

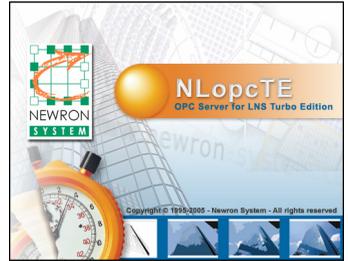
A setup program will guide you through the installation procedure and will ask you for any information necessary.

Installation of program

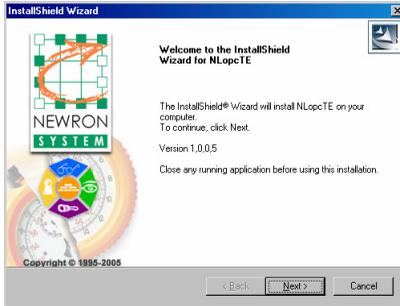
1. Insert the CDROM in the CD reader
2. If no window appears on the screen open: X:\MaquetteCD01.exe where X is the driver letter of your CD reader
3. After a couple of second, Picture 1 will appear.
4. Select **Software family** NLSUITE on main Menu
5. Picture 2 will appear on the screen.
6. Select a product logo or  for complete menu
7. The complete menu is show in Picture 4.
8. When you select a product, the Description information will appear like in Picture 5.
9. Select , for launch installation product.



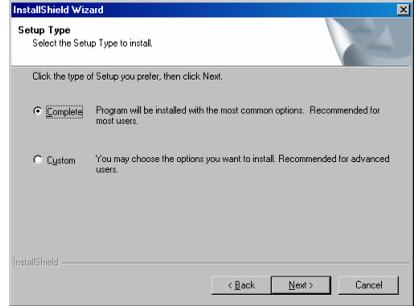
Picture 6 Installation Step 6



Picture 7 Install Step 7



Picture 8 Installation Step 8

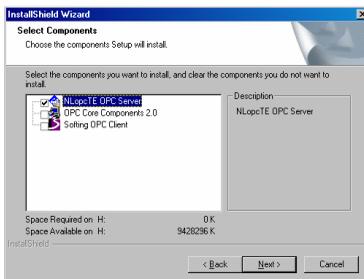


Picture 9 Install Step 9

You have the choice between the following installations:

Installation	Details
Complete	Complete installation of NLocpTE but WITHOUT generic Softing OPC client
Custom	You can select to install the generic Softing OPC client

Table 2 Type of installation



You should restart your PC at the end of the installation, according to the instructions

NETWORK INTERFACE

The network interface allows a physical link to be created between the PC and LonWorks network.

Valid interfaces

Type of interface	Maker	Connection	NSI	VNI
PCC10	Echelon	Slot PCMCIA	X	X
PCLTA21	Echelon	Slot PCI	X	X
ILON100	Echelon	Remote IP		X
U10	Echelon	USB	X	X
SLTA10	Echelon	Port RS232	X	
NIC709-PCI	Loytec	Slot PCI	X	X
NIC709-USB	Loytec	USB	X	X
NIC709-IP	Loytec	Remote IP	X	X
LPP	Gesytec	Slot ISA	X	
LPC	Gesytec	Slot PCI	X	

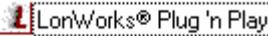
Table 3 Type of PC interface

To work, **NLopTE** needs a Firmware NSI or VNI interface

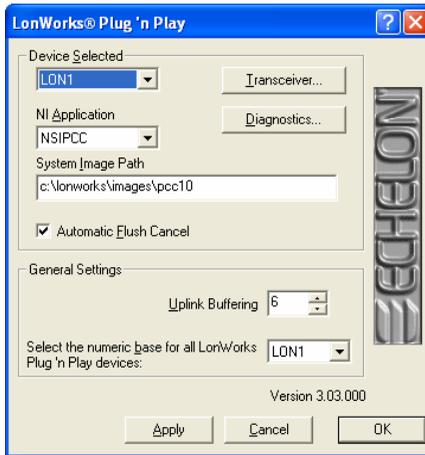
Verifying interface function

To function correctly, **NLopTE** must have a type NSI or VNI interface as shown in Table 3.

Verifying PCC10, PCLTA21

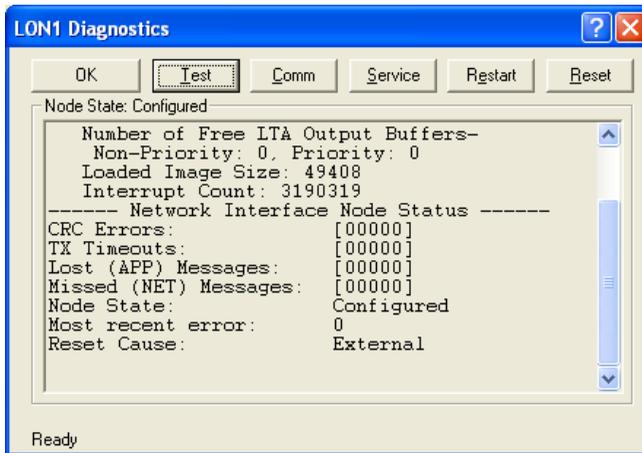
1. Open the configuration panel and launch the icon 

2.



Select NSI or VNI application.
Click on Diagnostics

3.



Click on Test to test your interface.

Verifying SLTA10

1. Activate: *Start/Programs/LonWorks SLTA10/SLTALink Manager*
2. The SLTA Link Manager menu bar must be as in Picture 11 with a green warning light, if it is red, or if the menu bar is like Picture 12, the interface is not working. Repeat the configuration steps, following the instructions carefully



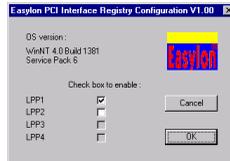
Picture 11 SLTA10 connected



Picture 12 SLTA10 unconnected

Verifying LPP

1. Open the configuration panel and launch the icon  Easylon PCI Interface
2. If a check box is valid, the interface is show by the system like in Picture 13



Picture 13 Valid LPP windows

Verifying Loytec Interface

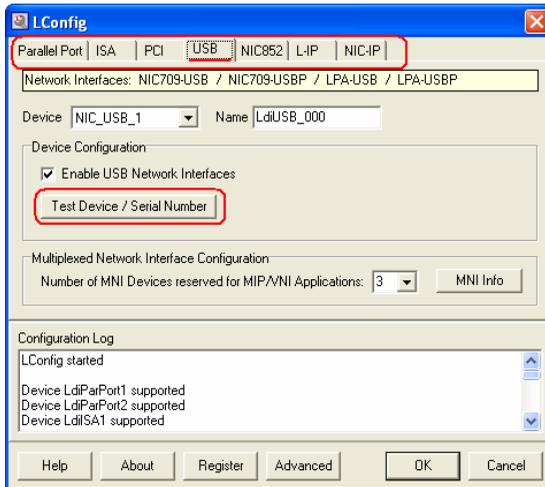
1. When Loytec Network Interfaces driver is installed you must have the Loytec tray icon.



2. Right click on the icon and select **LConfig** option.



3.



Select the tabulation depending on the type of your interface (ISA, PCI, USB, ...)
Click on Test Device/Serial number to test your interface.

FEATURES OF NLOPCTE

Introduction

NLopcTE is a LNS native OPC server. You need to have a LNS Turbo Edition 3.21 or greater installed on your machine.

NLopcTE is OPC 1.0, 2.05 and 3.0 compliant.



NLopcTE successfully pass OPC Compliance Tests.

You can consult **NLopcTE** web page on OPC Foundation Site : <http://www.opcfoundation.org/Products/ProductDetails.aspx?CM=1&RI=8476&CU=11>

This OPC server only needs to be configured. It can work on one single or several LNS databases.

You can also work in simulation mode to test your OPC client without being connected to the network.

LNS compliant

NLopcTE is LNS native. It can work simultaneously with any other LNS application, for example with NL220, the LNS manager tool.

NLopcTE fully supports iLon and any EIA852 compliant IP router.

NLopcTE fully supports Configuration properties.

NLopcTE fully translates SNVT or SCPT, even unique field from a structure.

NLopcTE fully translates UNVT or UCPT, even unique field from a structure. This is available if you included the User type definition in the catalog file (Ressource file)

Normal features

- ✓ Designed to be compliant to OPC 1.0, 2.05 and 3.0 specification.
- ✓ Connects LonWorks networks to process control-, visualisation-, SCADA, and front office software
- ✓ Allows to tag LonWorks devices: nodes and routers
- ✓ Allows to tag LonWorks Network Variables
- ✓ Allows to tag LonWorks Configuration properties

- ✓ Runs under Windows 2000, XP and 2003 server

Configuration

- ✓ **NLopcTE** Configuration Utility can be used to fully configure **NLopcTE** server
- ✓ Feature to import/export a configuration (xml format)
- ✓ Advanced/Normal configuration mode
- ✓ Tune your OPC server with advanced options
- ✓ Feature to change the special characters used in OPC item addresses when some characters are not supported by your OPC client
- ✓ Configure the contents of the OPC browser
- ✓ Configure the Fast Polling engine

Advanced features

- ✓ Operate as client/server architecture:
- ✓ Supports COM and DCOM mode
- ✓ Multi-client compatible
- ✓ Fast Polling feature to increase normal LNS polling performances
- ✓ Bad devices detection to avoid polling a device when absent on the network
- ✓ Can run in high level simulation mode in order to work offnet
- ✓ Can run with high-level trace mode to help you debugging your supervision.
- ✓ Supports OPC standard browser interface to access all items of its database.
- ✓ Supports several projects monitoring in a single OPC interface.
- ✓ Fully compliant with NLopcVNI server OPC addresses
- ✓ Supports NLopcVNI NLB/NLA files

Network Variables Monitoring

NLopcTE uses different ways to monitor network variables :

1. Bound to host network variable : The OPC server polls the network variable only one time for a start value and then waits for network updates.

2. Normal polling : The OPC server polls the network variable with the update rate of the OPC group the network variable belongs to.
3. Fast polling : The OPC server uses an internal engine to poll the network variables faster than normal LNS polling. The frequency of the fast polling depends on the update rate of the OPC groups the network variables belongs to.
4. Synchronous/Asynchronous read: **NLopcTE** polls the network variable on the network when an OPC client launches a read request.

Devices Monitoring

NLopcTE allows to monitor the state of a device (application device or router device) on the network.

NLopcTE uses query status messages to monitor a device.

When a device is considered as absent the network variables of the device are no more polled to avoid loss of performances.

Configuration Properties Monitoring

NLopcTE allows to monitor any configuration property.

You can monitor the database value or the network value of a configuration property.

NLopcTE uses different ways to monitor configuration properties :

1. Normal polling : The OPC server polls the configuration property with the update rate of the OPC group the configuration property belongs to. This behaviour must be explicitly requested by the OPC client because configuration properties monitoring can be very slow and decrease OPC server performances.
2. Synchronous/Asynchronous read: **NLopcTE** polls the configuration properties on the network when an OPC client launches a read request.

Formatted values:

NLopcTE can format any SNVTs, SCPTs, UNVTs, UCPTs with a network variable or a configuration property

NLopcTE uses the standard LonMark Resources Catalog to format the values.

It assumes all packaging, converting and updating tasks required for these objects. You can choose to get them in raw or in formatted value.

NLopcTE allows to access directly to any field of a structured type.

NLopcTE supports too MultiFields item to be able to write several fields of a structured type in one operation.

Limited version:

NLopcTE is available for evaluation in limited version.

NLopcTE has no limited function: you can check its entire functionality.

In evaluation version **NLopcTE** is limited to one hour running.

When the time expires (60 minutes) **NLopcTE** stops all network polling.

To unlock **NLopcTE** you need an hardware key (dongle USB or parallel).

Contact your distributor to get a hardware key.

Non Standard OPC Errors

NLopcTE uses only standard OPC errors.

NLOPCTE CONFIGURATION

Introduction

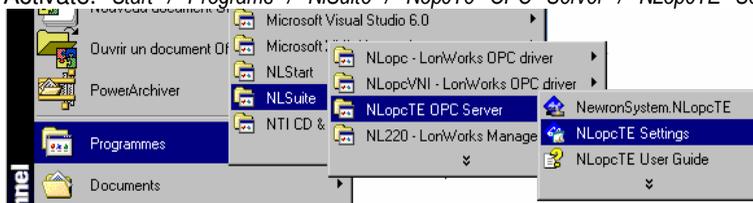
NLopcte needs to be configured at least one time.

NLopcte Configuration Utility implements a normal mode (with the minimum parameters to set) and an advanced mode with complete options to tune your OPC server.

Configuring NLopcte (Mandatory)

Execute **NLopcte** Configuration Utility:

1. Activate: *Start / Programs / NISuite / Nopcte OPC Server / NLopcte Settings*



Configuring the Modes



1. Click on **Modes** in the left menu.

2. Select the **LNS mode** :

Local: To monitor a local LNS database

Remote: To monitor a remote LNS database (over IP)

3. Select the **Simulation** mode.
Inactive to work in full network mode.
Active to work in simulation mode.

Simulation mode can be used to test your OPC client without being connected to the network.

In this mode you will be able to directly change the monitored items in **NLopcte** user interface to simulate a network value change.

4. Select the **Fast Polling** mode
Inactive to work without using the Fast Polling.
Active to work using the Fast Polling engine.

Fast Polling engine is an exclusive engine of **NLopcTE** to increase standard LNS monitoring performances.

If **Fast Polling** is used you must configure it (see later).

We highly recommend to use **Fast Polling** engine. This engine can increase from 2 to 10 the polling performances on the network.

Configuring the LNS Databases to monitor



1. Click on **LNS Databases** in the left menu.
2. Check the LNS databases you want to monitor with **NLopcTE**.

Database	Used	Network Interface	Retries	Repeats	TxTimer	RptTimer
BORNIER5	<input type="checkbox"/>					
CPPB	<input type="checkbox"/>					
MCLT53	<input type="checkbox"/>					
<input checked="" type="checkbox"/> NLopcTE	<input checked="" type="checkbox"/>	LON1	Default	Default	Default	Default
NLopcTE2	<input type="checkbox"/>					

3. For each selected database, select the used Network Interface.
Click on  to open the list with available Network Interfaces.
<Default> can be selected to use the last Network Interface used by this database.
<None> can be selected to use no Network Interface. Use this only in simulation mode.
4. For each selected database select the network timers and retries.
Default is the recommended value

Configuring the Tuning options



1. Click on **Tuning and options** in the left menu.
2. **Invisible when running as standalone**
If active then **NLopcTE** main window will be not visible when **NLopcTE** is manually launched.
In this case only the tray icon will be visible.
3. **Invisible when running as embedded**
If active then **NLopcTE** main window will be not visible when **NLopcTE** is automatically launched from an OPC client (embedded).
In this case only the tray icon will be visible.

4. **Run minimized**
If active then **NLopcTE** will be automatically minimized as a tray icon when launched.
5. **User cannot close**
If active then the interactive user will not be able to close **NLopcTE**
6. **Period to verify bad devices**
This is the frequency (in seconds) to verify absent devices on the network. When a device is considered as absent, **NLopcTE** will stop to poll any network variables of this device and this to increase network performances. When a device is verified and detected as present again then **NLopcTE** will start again the polling on this device.
7. **Fast add items**
Feature of **NLopcTE** to increase the initialization time when an OPC client adds the OPC item.
If active then **NLopcTE** will start the polling only when the OPC client stops to add OPC items.
It is highly recommended to activate this option.

Configuring the Fast Polling engine (used only if Fast Polling mode is active)



1. Click on **Fast Polling** in the left menu.
2. You must configure for each monitored LNS database the device templates that will support the Fast Polling.

Templates supporting fast polling	
Database : NLopcTE	
TEMPLATE1 [900A0E000300A001]	Normal
TEMPLATE2 [900A0E0002003004]	Normal
LTM [9000590000000000]	None
PANNEAU1 [9FFFFFF080A040400]	None
PANNEAU2 [9FFFFFF3D00040400]	None
PANNEAU3 [9FFFFFF1E20040400]	None

For each template you can select :

None : Fast polling not supported for this template

Normal : Normal Fast Polling used for this template

Host Based : Host Based Fast Polling for this template

Select **None** if you have if the Fast Polling causes problem with this type of devices.

Select **Normal** for any standard Neuron Chip based devices. In most cases a standard Neuron Chip based device supports Fast Polling without any problem.

Select **Host Based** if you device is a MIP device with the Fast Polling engine implemented in the device. In most cases if your devices is a MIP device prefer the option **None**.

Saving your configuration

Click on  or  to save your configuration.

Other options

Click on  to cancel any changes you have done on the configuration.

Click on  to set all the configuration to the default values.

Click on  to swap to advanced/normal mode.

Click on  to export your configuration into a XML file.

Click on  to import a configuration from a XML file.

Advanced Configuration

In advanced mode you have more options to tune your OPC server.

To swap between advanced/normal mode click on



Advanced configuration : Tuning and Options

Files\Browser file (NLB)

Full path name of the NLB file to use for the **NLopcTE** browser.

The NLB file can be used to customize the OPC browser.

For information about the NLB file format consult the user guide of **NLopcTE**.

Files\Alias file (NLA)

Full path name of the NLA file to use for the **NLopcTE** aliases.

Aliases can be used to customize OPC item addresses.

Aliases are discussed later in this document.

Appearance\Invisible when running as standalone

If active the user interface of **NLopcTE** will not be available when **NLopcTE** is launched manually.

Note : **NLopcTE** tray icon is available in all cases.

Appearance\Invisible when running as embedded

If active the user interface of **NLopcTE** will not be available when **NLopcTE** is automatically launched by an OPC client.

Note : **NLopcTE** tray icon is available in all cases.

Appearance\Run minimized

If active **NLopcTE** is automatically minimized as a tray icon when launched.

Appearance\User cannot close

If active the user cannot close **NLopcTE**.

Performances\Consecutive errors for bad devices

Enter here the number of consecutive network errors to consider the device as an absent device.

When the device is considered as absent all polling is stopped and the device is checked by a background task.

Performances\Period to verify bad devices (sec)

The interval in second to verify the state of devices considered as absent.

Performances\Unmark nv of bad devices

If active the network variables of a bad device are removed from the LNS monitoring engine.

Performances\Start polling after init

If active then **NLopcTE** will begin the network polling only when items are no more added by the OPC client.

It is highly recommended to activate this option to decrease the initialization time of your OPC client.

Performances\Initialization timer (millisecond)

This timer will be used as the time between the last item added and the polling on the network.

The timer is rearmed each time an item is added.

Not used if [**Start polling after init**] is Inactive.

Performances\Quality during initialization

The quality value of the items during the time of the creation of the item and the first polling of the item on the network.

Can be :

- **OPC_QUALITY_UNCERTAIN**

Uncertain

- **OPC_QUALITY_BAD with <No Initial Data> status**

Bad with the <No Initial Data> status of OPC 3.0 standard

- **No DataChange event before first reading**

If this option is used then your OPC client will receive no DataChange event for an item until a first read is done.

This means that between the time the item is created and the time it is polled for the first time, your client will have no value for the item.

Value format\Localized values format

Default format for localization.

Can be :

- Use regional settings

The format (SI or US) will be defined depending on the localization settings of your machine

- SI

The default format is SI

- US

The default format is US

Value format \Native format for enumeration

The default OPC format for an enumeration item.

Can be **string** or **integer**.

Note : In all cases, an OPC client can force an enumeration item type to string or integer depend less of this native format.

Value format\Enumeration format in multifield

The format of the enumeration fields in a multifield item.

Can be **string** or **integer**.

Multifield items are discussed later in this document.

Value format\Multified value separator

The character used in a mutifield string value to separate the value of each field.

Multifield items are discussed later in this document.

Value format\Raw value separator

The character used in a raw string value to separate the value of each byte.

Network\Do not write nv of bad devices

If active then when a write is send by an OPC client on a network variable belonging to an absent device then the write is not send on the network and an error is immediately returned.

If inactive then the write is send on the network in any case.

Network\Refuse write out of range

If active and if an OPC client writes an item with a value out of the range then the write is not send on the network and an error is returned.

If inactive then the write is accepted and send on the network in all cases.

Note : The ranges of an item are defined from the LonMark catalog or from the options **/min** and **/max**.

Write HeartBeat\Continue heartbeat on error

If active then **NLopcTE** continues to write an heartbeat even if an error occurs.

If inactive then **NLopcTE** stops an heartbeat when an error occurs.

Write HeartBeat is discussed later in this document.

Write HeartBeat\Continue heartbeat if item is removed

If active then **NLopcTE** continues to write an heartbeat even if an error occurs.

If inactive then **NLopcTE** stops an heartbeat when an error occurs.

Write HeartBeat is discussed later in this document.

LNS\Server for remote TCP clients

If checked then **NLopcTE** will be a TCP server for each monitored LNS database.

In this case a remote TCP machine can remotely access the databases.

LNS\Deactivate LNS network check

LNS engine continuously verifies the devices status on the network.

If this option is active then **NLopcTE** will automatically deactivate this LNS feature to decrease network bandwidth usage.

LNS\LNS Remote Event Interval

This interval is the minimum time between two LNS events send by the LNS server to the LNS Lightweight client.

Beware that the lower the value is the more traffic you will have on your TCP network.

Advanced configuration : Advanced settings

OPC item's address format\Address marker

Character used to separate each part of an OPC item address.

Addresses format is discussed later in this document.

OPC item's address format\Address' option marker

Character used to separate each option of an OPC item address.

Addresses format is discussed later in this document.

OPC item's address format\Type format marker

Character used as prefix for the format of an item (SI, US, ...)

Addresses format is discussed later in this document.

OPC item's address format\Property marker

Character used as prefix for a property item.

Addresses format is discussed later in this document.

OPC item's address format\Field separator

Character used to separate each field of a structured type.

Addresses format is discussed later in this document.

OPC item's address format\Multifield separator

Character used to separate each field of a multifield item address.

Multifield Items is discussed later in this document.

OPC Standard Interface\Address validation

Define the way **NLopcTE** will verify the address of an item.

Can be :

- **Complete**

The format of the address is verified and the LNS objects are verified in the LNS database

- **Format**

Only the format of the address is verified

Note : **Complete** is the recommended value. In all cases use **Complete** if you have any problem with your OPC client.

OPC Standard Interface\Node validation

Define the way **NLopcTE** will verify the format of a node path.

Can be :

- **Complete**

The format of the address is verified and the LNS objects are verified in the LNS database

- **Format**

Only the format of the address is verified

- **None**

No verification is done.

Note : **None** is the recommended value because it can greatly increase the performance of your OPC client and it will work with most clients. In all cases use **Complete** if you have any problem with your OPC client.

OPC Standard Interface\Supports EU range

If active then the min and max value (if any) of an OPC item will be available on the **NLopcTE** OPC standard interface.

OPC Standard Interface\Supports EU enumerated

If active then the enumeration values (for enumerated items only) of an OPC item will be available on the **NLopcTE** OPC standard interface.

NLopcVNI Backward Compatibility\Support old address format (NLopcVNI)

If active then the old address format of **NLopcVNI** will be supported.

This is recommended for backward compatibility with **NLopcVNI** clients.

It is HIGHLY recommended to use the new address format for new clients.

NLopcVNI Backward Compatibility\SNVT_switch automatic update

If active then when writing the field value or the field state of a SNVT_switch will cause the other field to change accordingly.

value<>0 -> state=1, value=0 -> state=0, state=1 -> value=100, state=0 -> value=0.

Recommended is [Active] for backward compatibility with NLopcVNI.

NLopcVNI Backward Compatibility\SNVT_setting automatic update

If active then when writing the field setting or rotation of a SNVT_setting will cause the field function to receive SET_STATE (5).

Recommended is [Active] for backward compatibility with NLopcVNI.

NLopcVNI Backward Compatibility\Raw is decimal

If active then a raw value when using old NLopcVNI address format will be a decimal raw value, if inactive a raw value will be an hexadecimal raw value.

Recommended is [Active] for backward compatibility with NLopcVNI.

OPC browser contents\Include device state item

If active then the state item of each device will be present in the browser.

Note : This option is not used if you use a NLB file.

OPC browser contents\Include device presence item

If active then the presence item of each device will be present in the browser.

Note : This option is not used if you use a NLB file.

OPC browser contents\Include dynamic interfaces

If active then the dynamic interface (for device supporting dynamic interfaces) will be present in the browser.

Note : This option is not used if you use a NLB file.

OPC browser contents\Include LonMark objects

If active then LonMark objects will be present in the browser.

Note : This option is not used if you use a NLB file.

OPC browser contents\Include Network Variables by name

If active then the network variables (by name) will be present in the browser.

Note : This option is not used if you use a NLB file.

OPC browser contents\Include Network Variables by index

If active then the network variables (by index) will be present in the browser.

Note : This option is not used if you use a NLB file.

OPC browser contents\Include nv raw value

If active then the raw value item of each network variable will be present in the browser.

Note : The raw value is always present in the browser for a network variable without any recognized type (SNVT or UNVT).

Note : This option is not used if you use a NLB file.

OPC browser contents\Use LNS format spec for nv type

If active then **NLopcTE** will use the LNS format type of a network variable to define the default type to use.

Note : This option is not used if you use a NLB file.

OPC browser contents\Add compliant UNVT type

If active then all compliant UNVT type (UNVT type with the same length as the length of the network variable) will be added for any user network variable.

Note : This option is not used if you use a NLB file.

OPC browser contents\Include configuration properties by name

If active then the configuration properties (by name) will be present in the browser.

Note : This option is not used if you use a NLB file.

OPC browser contents\Include configuration properties by index

If active then the configuration properties (by index) will be present in the browser.

Note : This option is not used if you use a NLB file.

OPC browser contents\Include cp raw value

If active then the raw value item of each configuration property will be present in the browser.

Note : The raw value is always present in the browser for a configuration property without any recognized type (SCPT or UCPT).

Note : This option is not used if you use a NLB file.

OPC browser contents\Use LNS format spec for cp type

If active then **NLopcTE** will use the LNS format type of a configuration property to define the default type to use.

Note : This option is not used if you use a NLB file.

OPC browser contents\Add all formats

If active then all available formats of each type will be present in the browser.

If inactive then only the default format is present in the browser.

Note : This option is not used if you use a NLB file.

LNS objects cache\LNS cache active

Activate or inactivate **NLopcTE** LNS cache system.

It is HIGHLY recommended to activate this option to increase performances.

LNS objects cache\Subsystems cache size

Number of subsystem objects kept in cache.

LNS objects cache\Application devices cache size

Number of application device objects kept in cache.

LNS objects cache Router devices cache size

Number of router device objects kept in cache.

LNS objects cache\Profile cache size

Number of profile (LonMark object) objects kept in cache.

LNS objects cache\Network variables cache size

Number of network variable objects kept in cache.

LNS objects cache\Configuration properties cache size

Number of configuration property objects kept in cache.

LNS objects cache\General objects cache size

Number of general objects kept in cache.

Advanced configuration : Fast polling

General settings\Ignore bound to host nv

If active then **NLopcTE** will not use the Fast Polling for network variables bound to the local host.

It is highly recommended to activate this option.

General settings\Fast Polling granularity length (bytes)

When a network variable expires and must be read, **NLopcTE** will read this network variable and any other network variable of the device which will use a message with a length lesser or equal to this granularity value.

If **255** then any network variable of the device will be read with the network variable which expired.

General settings\Fast Polling granularity time (second)

When a network variable expires and must be read, **NLopcTE** will read this network variable and any other network variable of the device whose update rate expires in a time lesser or equal to this granularity value.

If **zero** then only the expired network variables will be read.

If **999** then any network variable of the device will be read with the network variable which expired.

General settings\Security Length

Some devices does not support Fast Polling messages using the maximum device buffer size.

This length is used to reduce the maximum size of the fast polling messages in such cases.

You can set this value to zero to disable security mode.

General settings\Do not use LNS monitoring engine

If checked then **NLopcTE** will use the internal LNS monitoring engine ONLY for BOUND nvs.

Other nvs will be fetched by **NLopcTE** engine even if fast polling is not used.

Note that internal LNS monitoring engine is always used when fast polling is inactive.

Settings for MIP devices\Request code

Request code message for MIP fast polling engine (used only with host based devices).

The recommended value is 1.

Settings for MIP devices\Response code

Response code message for MIP fast polling engine (used only with host based devices).

The recommended value is **2**.

Activating/Inactivating fast polling for individual devices

In advanced mode you can activate/inactivate the Fast Polling for individual devices.

NLOPCTE ITEMS ADDRESS

About OPC Addresses

An OPC client access can supervise network variables, configuration properties or devices through OPC items.

An OPC item is identified by its address.

OPC item address must follow the explicit rules of **NLOpcTE**.

Notes :

- Old address format of NLOpcVNI are supported.
- OPC item addresses can be identified too using the OPC browser interface of **NLOpcTE**

Address for Network Variables

Formatting

A network variable can be:

- A SNVT network variable.
- A known UNVT network variable (it means a UNVT declared in the catalog file)
- An unknown UNVT network variable (it means a UNVT not found in the catalog file).

The value of the network variable can be accessed with several formats :

- Raw decimal
- Raw hexadecimal
- Formatted type
- Field of a structured type

The native type of a network variable item is:

- Long (VT_I4) for an enumeration (can be changed in **NLOpcTE** Configuration Utility in advanced mode)
- Long (VT_I4) for any integer value
- Boolean (VT_BOOL) for any integer value with a minimum value of zero and a maximum value of 1
- Double real (VT_R8) for any real value
- String (VT_BSTR) for any raw value

Format of the address of a network variable

The address of an item for a network variable is :

```
<DeviceAddress>[.<interface>][.<profile>].<nv>.<Field>
```

<DeviceAddress> is the full address of the device the network variable belongs to.

It can be :

- **@P<projectname>.@S<SubsystemPath>.@D<DeviceName>**
IE : @PProject1.@SLocations.Building1.Floor1.@DNode1
- **@P<projectname>.@V<DeviceHandle>**
IE : @PProject1.@V1234
- **@H**
For the host

<interface> is optional.

It can be used to identify a dynamic or static interface of an application device supporting multiple interfaces.

<interface> can be :

- **@ISTATIC**
For the static interface
- **@I<interfacename>**
IE : @Iintfc1

<profile> is optional.

It can be used to identify a profile (LonMark object).

<profile> can be :

- **@O<ProfileName>**
IE : @O<LM1>
- **@Z<ProfileLonIndex>**
IE : @Z0

- @J<ProfileCollectionIndex>

IE : @J1

<nv> can be :

- @N<networkvariablename>

IE : @Nnv1

- @L<networkvariablelonindex>

IE : @L0

- @X<networkvariablecollectionindex>

IE : @X1

<Field> is used to identify the format of the value.

It can be :

- @Frawdec
For a raw decimal value
- @Frawhex
For a raw hexadecimal value
- @F<SNVT_type>[#<format>][:<fieldpath>]
For a SNVT type.
<SNVT_type> is the name of the SNVT (ie : SNVT_lev_disc)
#<Format> is optional can be any supported format for the type (ie : #SI or #US)
<fieldpath> is the full path name of a field of the structured type
Each part of path is separated by a :
ie : struct1:struct2:field1
- @F<pid>::UNVT_type>[#<format>][:<fieldpath>]
For a UNVT type.
<pid> is the program ID of the user type (ie : 80010203040506007)
<UNVT_type> is the name of the UNVT (ie : UNVT_mytype)
#<Format> is optional and can be any supported format for the type
(ie : #SI or #US)
<fieldpath> is the full path name of a field of the structured type
Each part of path is separated by a :
ie : struct1:struct2:field1

Examples of full network variable addresses

@PPrj1.@H.@Nnv1.@Frawdec

@PPrj1.@SLocations.Building1.Floor1.@DNode1.@Nnv1.@Frawdec

@PPrj1.@SLocations.Building1.Floor1.@DNode1.@Nnv1.@FSNVT_lev_disc
@PPrj1.@SLocations.@DNode1.@Nnv1.@FSNVT_alarm:alarm_type
@PPrj1.@SLocations.Building1..@DNode1.@Nnv1.@F9001034356554401::
UNVT_mytype:st:st2:v1
@PPrj1.@V1234.@Nnv1.@FSNVT_lev_disc
@PPrj1.@SLocations.Building1.Floor1.@DNode1.@L0.@FSNVT_lev_disc
@PPrj1.@SLocations.@DNode1.@Intfc1.@Nnv1.@FSNVT_lev_disc
@PPrj1.@SLocations.Building1.@DNode1.@OLM1.@Nnv1.@FSNVT_lev_disc

Address for Configuration Properties

Formatting

A configuration property can be:

- A SCPT
- A known UCPT (it means a UCPT declared in the catalog file)
- An unknown UCPT (it means a UCPT not found in the catalog file).

The value of the configuration property can be accessed with several formats :

- Raw decimal
- Raw hexadecimal
- Formatted type
- Field of a structured type

The native type of a configuration property item is:

- Long (VT_I4) for an enumeration (can be changed in **NLopcTE** Configuration Utility in advanced mode)
- Long (VT_I4) for any integer value
- Boolean (VT_BOOL) for any integer value with a minimum value of zero and a maximum value of 1
- Double real (VT_R8) for any real value
- String (VT_BSTR) for any raw value

Format of the address of a configuration property

The address of an item for a configuration property is :

```
<DeviceAddress>[.<interface>][.<profile>][.<nv>].<cp[element]>.<Field>
```

<DeviceAddress> can be :

- **@P<projectname>.@S<SubsystemPath>.@D<DeviceName>**
IE : **@PProject1.@SLocations.Building1.Floor1.@DNode1**
- **@P<projectname>.@V<DeviceHandle>**
IE : **@PProject1.@V1234**

<interface> is optional.

It can be used to identify a dynamic or static interface of an application device supporting multiple interfaces.

<interface> can be :

- **@ISTATIC**
- **@I<interfacename>**
IE : **@Iintfc1**

<profile> is optional.

It can be used to identify a profile (LonMark object).

<profile> can be :

- **@O<ProfileName>**
IE : **@O<LM1>**
- **@Z<ProfileLonIndex>**
IE : **@Z0**
- **@J<ProfileCollectionIndex>**
IE : **@J1**

<nv> is optional

It can be used to access a configuration property of a network variable.

<nv> can be :

- **@N<networkvariablename>**
IE : **@Nnv1**
- **@L<networkvariablelonindex>**
IE : **@L0**
- **@X<networkvariablecollectionindex>**
IE : **@X1**

<cp[element]> can be :

[element] is used only if the configuration property is an array.

[element] is 1 based (first element=1).

- **@C<configurationname>[element]**
IE : **@Ccp1**
IE : **@Ccp1[1]**

- **@Y<configurationindex>[element]**

IE : @Y1

IE : @Y1[1]

<Field> can be :

- **@Frawdec**
For a raw decimal value
- **@Frawhex**
For a raw hexadecimal value
- **@F<SCPT_type>[#<format>][:<fieldpath>]**
For a SCPT type.
<SCPT_type> is the name of the SCPT (ie : SCPTMaxSendTime)
#<Format> is optional can be any supported format for the type (ie : #SI or #US)
<fieldpath> is the full path name of a field of the structured type
Each part of path is separated by a :
ie : struct1:struct2:field1
- **@F<pid>::UCPT_type>[#<format>][:<fieldpath>]**
For a UCPT type.
<pid> is the program ID of the user type (ie : 80010203040506007)
<UNVT_type> is the name of the UCPT (ie : UCPT_mytype)
#<Format> is optional and can be any supported format for the type (ie : #SI or #US)
<fieldpath> is the full path name of a field of the structured type
Each part of path is separated by a :
ie : struct1:struct2:field1

Examples of full addresses :

```

@Prj1.@SLocations.Building1.Floor1.@DNode1.@Ccp1.@Frawdec
@Prj1.@SLocations.Building1.Floor1.@DNode1.@Ccp1.@Frawhex
@Prj1.@SLocations.Building1.@DNode1.@Ccp1.@FSCPT_maxRcvTime
@Prj1.@SLocations.Building1.@DNode1.@Ccp1[1].@FSCPT_maxRcvTime
@Prj1.@SLocations.Building1.@DNode1.@Ccp1.@FSCPTToemType:ascii[0]
@Prj1.@SLocations.Building1.@DNode1.@Ccp1.@F9001034356554401::
UCPT_mytype:st1:st2:v1
@Prj1.@V1234.@Ncp1.@FSCPT_maxRcvTime
@Prj1.@SLocations.Building1.@DNode1.@Y1.@FSCPT_maxRcvTime
@Prj1.@SLocations..@DNode1.@OLM1.@Ccp1.FSCPT_maxRcvTime
@Prj1.@SLocations..@DNode1.@Nnv1.@Ccp1.FSCPT_maxRcvTime
@Prj1.@SLocations.Building1.@DNode1.@Intfc1.@OLM1.@Nnv1.@Ccp1.
FSCPT_maxRcvTime

```

Address for Devices (application or router)

Formatting

A device value can be :

- The device state
Native type : long (VT_I4)
- The device presence
Native type : long (VT_I4)

Format of the address of a device

<DeviceAddress>.<Field>

<DeviceAddress> can be :

- **@P<projectname>.@S<SubsystemPath>.@D<ApplicationDeviceName>**

IE : **@PProject1.@SLocations.Building1.Floor1.@DNode1**

- **@P<projectname>.@V<ApplicationDeviceHandle>**

IE : **@PProject1.@V1234**

- **@P<projectname>.@S<SubsystemPath>.@R<RouterDeviceName>**

IE : **@PProject1.@SLocations.Building1.Floor1.@RRouter1**

- **@P<projectname>.@E<RouterDeviceHandle>**

IE : **@PProject1.@E11**

<Field> can be :

- **@Fstate**
- **@Fpresence**

Examples of full addresses :

@PProject1.@SLocations.Building1.Floor1.@DNode1.@Fstate

@PProject1.@SLocations.Building1.Floor1.@DNode1.@Fpresence

@PProject1.@SLocations.Building1.Floor1.@RRt1.@Fstate

@PProject1.@SLocations.Building1.Floor1.@RRt1.@Fpresence

@PProject1.@V1234.@Fstate

@PProject1.@V1234.@Fpresence

@PProject1.@E11.@Fstate

@PProject1.@E11.@Fpresence

MultiField Items

What is a MultiField item ?

MultiField items can be used only for structured type (SNVT,SCPT,UNVT or UCPT).

The objective is to be able to read and write several fields of a structured type using only one item.

The main interest of **MultiField** items is to be able to write several fields of a structure in only one network write.

The most common example is the **SNVT_switch** type.

In most cases you may need to be able to write the fields **value** AND **state** in only one write.

How to address a MultiField item ?

Addressing a MultiField item is very similar than accessing a single field of a network variable or a configuration property.

For example to address the fields value and state of a SNVT_switch you may use :

```
@PProject1.@SLocations.@DNode1.@Nnv1.@FSNVT_switch:value  
@PProject1.@SLocations.@DNode1.@Nnv1.@FSNVT_switch:state
```

To access the TWO fields in one item you do :

```
@PProject1.@SLocations.@DNode1.@Nnv1.@FSNVT_switch:value|state
```

Each field names are separated by a |.

Note : The *multifield fields separator* | can be changed in **NLopcTE** Configuration Utility.

You can set as many fields of the structure you want in a single address.

MultiField items can be used with network variables and configuration properties

What is the format of the value of a MultiField item ?

The value is ALWAYS a string (VT_BSTR).

You cannot request a MultiField item with a different type than VT_BSTR.

The value of the item is the value of each field separated by a <space>.

The order of the value of each field in the string is the same order as the order of the fields in the address.

Note : The *<space> separator character* can be changed in **NLopcTE** Configuration Utility.

For example for

```
@PProject1.@SLocations.@DNode1.@Nnv1.@FSNVT_switch:value|state
```

The value may be :

"100 1" : 100 is for the value and 1 is for the state.

So for example if you write the same item with the value "0 0", **NLopcTE** will write the item with the value=0 and the state=0 IN ONE network write.

Using aliases for item addresses

What are aliases ?

Aliases are string you can use to replace part or a complete OPC Item Address. Using aliases is fully optional.

How to define aliases ?

Aliases are defined in a NLA file.
Consult user guide for more information about aliases file format.
You can define the NLA file path in **NLopcTE** Configuration Utility.

Each alias are defined by :
The alias
The string to replace

Using an alias as a full address

When you have an alias in the NLA file that points to a full address, then any OPC client can add an item by using the alias instead of the full address.

An example :
Alias : **NV1**
String to replace :
`@PPrj1.@SLocations.Building1.@DNode1.@Nnv1.@FSNVT_lev_disc`

Then if an OPC client adding an item with the address **NV1** is the same as adding an item with the full address
`@PPrj1.@SLocations.Building1.@DNode1.@Nnv1.@FSNVT_lev_disc.`

Using an alias as a part of an address

You can define an alias in the NLA file that points to a part of an address. This alias can be used in any complete address to replace the string the alias points to.

An example :
Alias : **DEVICE1**
String to replace : `@PProject1.@SLocations.Building1.@DNode1`

Then you can use the marker **@A** to use your alias like this :
`@ADEVICE1.@Nnv1.@FSNVT_lev_disc.`

@ADEVICE1.@Nnv2.@FSNVT_lev_disc.

is the same as :

@PProject1.@SLocations.Building1.@DNode1.@Nnv1.@FSNVT_lev_disc.

@PProject1.@SLocations.Building1.@DNode1.@Nnv2.@FSNVT_lev_disc.

Note :

You can use several aliases in the same address.

For example :

Alias : **DEVICE1**

String to replace : @PProject1.@SLocations.Building1.@DNode1

Alias : **LEVDISC**

String to replace : @FSNVT_lev_disc

You can do :

@ADEVICE1.@Nnv1.@ALEVDISC

@ADEVICE1.@Nnv2.@ALEVDISC

Special Items

Some OPC items are available as special items.

@\$LIFESIGNAL

This item is a long integer which increments at the same frequency as the update rate of the OPC group the item belongs to.

For example if the group update rate is 1000 ms then the item will increment each second.

This item is read-only.

Native type : Long integer (VT_I4)

Possible request type : String (VT_BSTR), Any numeric type

@\$SILENT

This item is a Boolean which indicates if **NLopcTE** works in silent mode or not.

The value is false if **NLopcTE** does not work in silent mode.

The value is true if **NLopcTE** works in silent mode.

In silent mode **NLopcTE** stops any polling on the network.

You can write this item to force the silent mode of **NLopcTE**.

This item is read/write.

Native type : Boolean (VT_BOOL)

Possible request type : String (VT_BSTR), Any numeric type

@\$NLTESTCHANNEL_CHANNELS

This item is the list of the channels state accessed from NLTestChannel.

The value is a XML string with all channels state.

This item is read only.

Native type : String (VT_BSTR)
Possible request type : String (VT_BSTR)

@\$NLTESTCHANNEL_ALARMS

This item is the list of the current alarms accessed from NLTestChannel.
The value is a XML string with all current alarms.

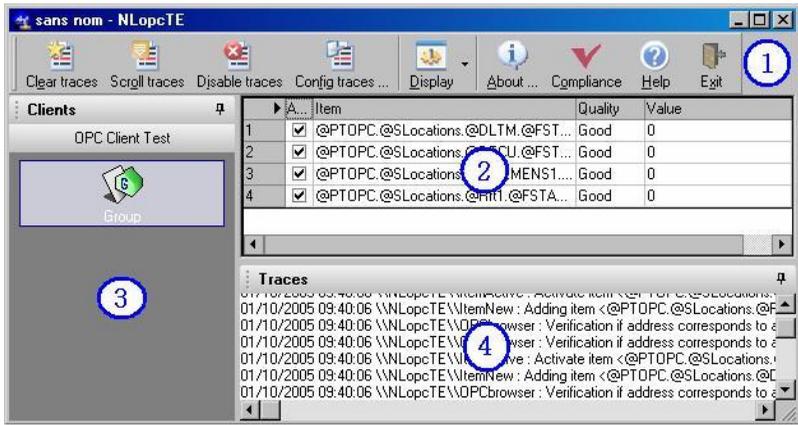
This item is read only.

Native type : String (VT_BSTR)
Possible request type : String (VT_BSTR)

NLOPCTE USER INTERFACE

Human interface

NLopcTE include a toolbar and three main views :



Picture 14
Human interface of **NLopcTE** server



The toolbar



The OPC items of an OPC group



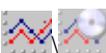
The connected OPC clients and the existing OPC groups



The traces

The toolbar



	Set the traces window to auto-scroll mode
	Disable traces (pressed if disabled)
	Configure the traces
	Configure the OPC items grid contents
	Change the OPC groups display mode (list/toolbox)
	Set NLOpcTE window always visible or not (pressed if always visible)
	Open the devices statistics window (described later)
	Activate/Inactivate the silent mode . In silent mode, NLOpcTE stops any polling on the network.
	Save the actual simulation scenes Note : Available only in simulation mode
	Remove all actual simulation scenes Note : Available only in simulation mode
	Display NLOpcTE about box
	Display NLOpcTE Compliance Tests Results
	Display NLOpcTE user guide
	Exit NLOpcTE . A warning will be displayed if at least one OPC client is still connected to the server.

The connected OPC clients and OPC groups



Picture 15
Connected OPC clients and existing OPC groups

This list displays the connected clients and the OPC groups of each client. To display the groups of a client, simply click on the client name. To view items of a group, simply click on the group.

You can display the properties of a group by double-clicking on the group.

You can reduce the list by clicking on the symbol . The list will be reduced to a vertical tabulation like this:

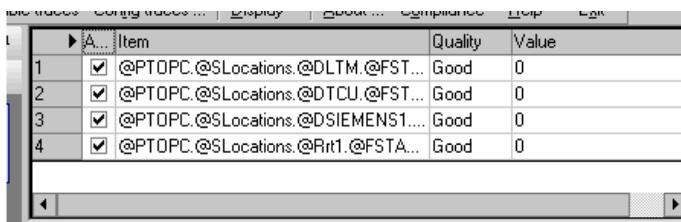


To show the list, simply click on the tabulation.

To restore the list visibility, click on the tabulation and then click on the symbol .

If required, you can scroll the group using the symbols  and .

The OPC items of an OPC group



	Item	Quality	Value
1	<input checked="" type="checkbox"/> @PTOPC.@SLocations.@DLTM.@FST...	Good	0
2	<input checked="" type="checkbox"/> @PTOPC.@SLocations.@DTCU.@FST...	Good	0
3	<input checked="" type="checkbox"/> @PTOPC.@SLocations.@DSIEMENS1...	Good	0
4	<input checked="" type="checkbox"/> @PTOPC.@SLocations.@Rrt1.@FSTA...	Good	0

This window displays the OPC items of the selected OPC group.

To display the items of a group simply select the group in the Clients and Groups list (left view).

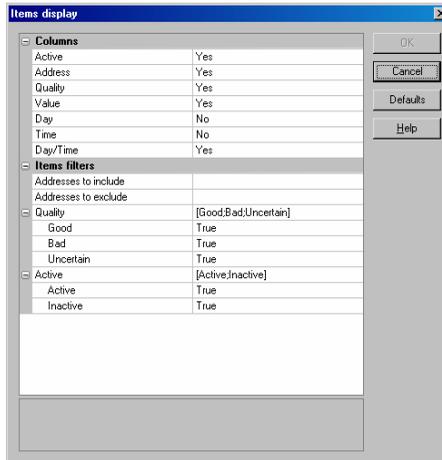
The available columns of the grid are :

- **Active**
Checked if the item is active.
Unchecked if the item is inactive.
- **Item**
The address of the item.
- **Quality**
The actual quality of the item.
- **Value**
The actual value of the item.
- **Day**
The day of the actual timestamp of the item.
- **Time**
The time of the actual timestamp of the item.
- **Day/Time**

The day and the time of the actual timestamp of the item.

Configuring the OPC items grid

You can configure the contents of the grid by clicking on   in the toolbar.



Picture 16
Configuring the OPC items grid

- **Columns/Active**
Show/Hide the active column
- **Columns/Address**
Show/Hide the address column

- **Columns\Quality**
Show/Hide the quality column
- **Columns\Value**
Show/Hide the value column
- **Columns\Day**
Show/Hide the day column
- **Columns\Time**
Show/Hide the time column
- **Columns\Day/Time**
Show/Hide the day/time column
- **Items filters\Addresses to include**
You can set a filter on the address of items you want to be displayed in the grid.
A filter can be :
<string>*
Any item the address begins with **<string>**
***<string>**
Any item the address ends by **<string>**
<string>
Any item the address contains **<string>**
<string>
Any item the address is **<string>**
You can set several filters by separating each to other with a semi colon (;).
- **Items filters\Addresses to exclude**
You can set a filter on the address of the items you want to be NOT displayed in the grid.
A filter can be :
<string>*
Any item the address begins with **<string>**
***<string>**
Any item the address ends by **<string>**
<string>
Any item the address contains **<string>**
<string>
Any item the address is **<string>**
You can set several filters by separating each to other with a semi colon (;).
- **Items filters\Quality**
You can set a filter on the quality of the items you want to be displayed in the grid.
- **Items filters\Quality**
You can set a filter on the active flag of the items you want to be displayed in the grid.

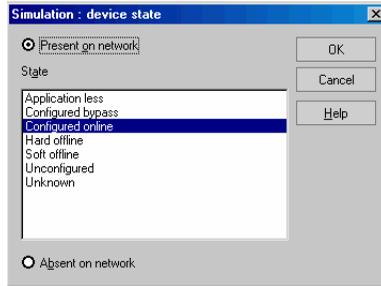
If you work in [simulation mode](#), you can double click on an item to edit and change its value.

Changing value in Simulation mode

Simulate the value of a device

To edit the simulate value of a device :

- Select the group of the OPC item in the Clients and Groups list
- Double click on the OPC item in the OPC items grid



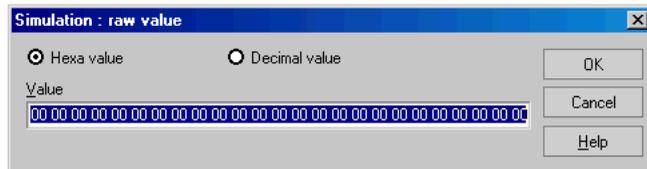
Picture 17
Simulate the value of a device

- **Present on network**
Check this to simulate the device as present.
- **State**
Select the simulate state of the device.
- **Absent on network**
Check this to simulate the device as absent.

Simulate a raw value

To edit the raw simulate value of an item (network variable or configuration property) :

- Select the group of the OPC item in the Clients and Groups list
- Double click on the OPC item in the OPC items grid



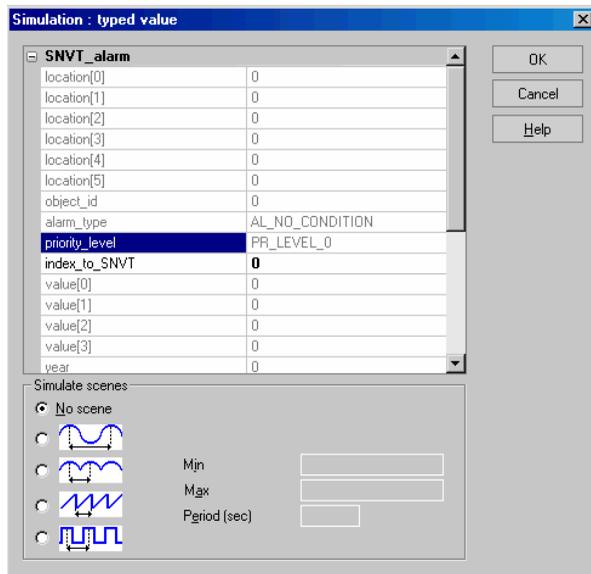
Picture 18
Simulate a raw value

- **Hexa value**
Checked if the raw value is an hexadecimal raw value.
- **Decimal value**
Checked if the raw value is a decimal raw value.
- **Value**
The raw value.
Each byte must be separated by a space.

Simulate a formatted value (SNVT,UNVT,SCPT,UCPT)

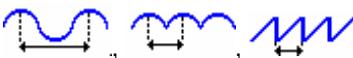
To edit the typed simulate value of an item (network variable or configuration property) :

- Select the group of the OPC item in the Clients and Groups list
- Double click on the OPC item in the OPC items grid



Picture 19
Simulate a formatted value

- **Value**
The grid contains all fields of the value (only one field is the value is not a structure).
You can change only the line in bold (corresponding to the item you simulate).
- **Simulate scenes**
A simulate scene can be used to force the value of the item to change following a scene.

There are four forms of scene :  and 

For a scene you must enter :

The **min** value
The **max** value
The **period** in second

The traces



```
Traces
01/10/2005 10:14:56 \\NLOpcTE\NETVerifDevice: Verify presence of application device
01/10/2005 10:14:56 \\NLOpcTE\NETVerifDevice: Verify presence of application device
01/10/2005 10:14:57 \\NLOpcTE\NETVerifDevice: Verify presence of application device
01/10/2005 10:14:57 \\NLOpcTE\NETread: Query status of router device: <TOPC.Loca
01/10/2005 10:14:59 \\NLOpcTE\NETread: Query status of router device: <TOPC.Loca
01/10/2005 10:15:01 \\NLOpcTE\NETread: Query status of router device: <TOPC.Loca
01/10/2005 10:15:02 \\NLOpcTE\NETread: Query status of router device: <TOPC.Loca
01/10/2005 10:15:02 \\NLOpcTE\NETreaderror: Error getting network status [no respons
01/10/2005 10:15:03 \\NLOpcTE\NETread: Query status of router device: <TOPC.Loca
```

Picture 20
Traces

The traces display all information, warnings and errors.

You can clear the traces by clicking on  in the toolbar.

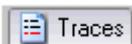
You can disable the traces by clicking on  in the toolbar.

You can active the auto-scroll mode by clicking on  in the toolbar.

You can configure the traces by clicking on  in the toolbar.

You can reduce the traces window by clicking on the symbol .

The traces window will be reduced as a horizontal tabulation like this :

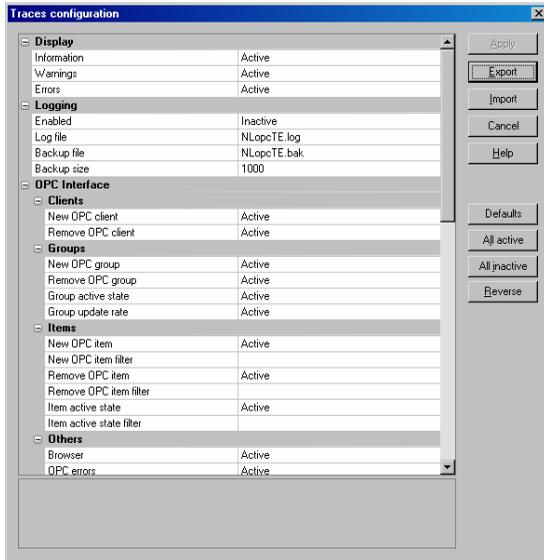


To show the traces simply click on the tabulation.

To restore the traces visibility click on the tabulation and then click on the symbol .

Configuring the traces

To configure the trace click on  in the toolbar.



Picture 21
Configuring the traces

IMPORTANT NOTE :

Using trace filters (filter on string) can greatly decrease **NLOpcTE** performances.

Use filters only when required. Remove them when no more needed.

Note about filters :

You can set several filter for the traces.

Each filter can be :

*<string>**

Begins with <string>

**<string>*

Ends by <string>

<string>

Contains <string>

<string>

Is <string>

You can set several filters by separating each to other with a semi colon (;).

- **DisplayInformation**
Activate/inactivate information traces.
- **DisplayWarnings**
Activate/inactivate warning traces.
- **DisplayErrors**
Activate/inactivate errors traces.
- **Logging\Enabled**
Activate\Inactivate traces logging.
If active all traces will be written in a log file.
IMPORTANT : Traces logging greatly decreases **NLOpcTE**

performances.

Remember to disable logging when you do not need it anymore.

- **Logging\Log file**
File used to log the traces.
If a file is set without path then the path of **NLopcTE** executable will be used.
- **Logging\Backup file**
File used as a backup for the log file.
When the log file size reaches a limit then the log file is backup and then deleted.
If a file is set without path then the path of **NLopcTE** executable will be used.
- **Logging\Backup size**
Limit size in kilo bytes to backup the log file.
- **OPC interface\Clients\New OPC client**
If active a trace is added each time a client connects to **NLopcTE**.
- **OPC interface\Clients\Remove OPC client**
If active a trace is added each time a client disconnects from **NLopcTE**.
- **OPC interface\Groups\New OPC group**
If active a trace is added each time a new OPC group is added by a client.
- **OPC interface\Groups\Remove OPC group**
If active a trace is added each time a new OPC group is removed by a client.
- **OPC interface\Groups\Group active state**
If active a trace is added each time an OPC group active state changes.
- **OPC interface\Groups\Group update rate**
If active a trace is added each time an OPC group update rate changes.
- **OPC interface\Items\New OPC item**
If active a trace is added each time a new OPC item is added by a client.
- **OPC interface\Items\New OPC item filter**
Filter used to display traces when new OPC items are added.
The filter works on the address of the item.
- **OPC interface\Items\Remove OPC item**
If active a trace is added each time a new OPC item is removed by a client.
- **OPC interface\Items\New OPC item filter**
Filter used to display traces when new OPC items are removed.
The filter works on the address of the item.
- **OPC interface\Items\Item active state**
If active a trace is added each time an OPC item active state changes.
- **OPC interface\Items\Item active state filter**
Filter used to display traces when OPC items active state changes.
The filter works on the address of the item.
- **OPC interface\Others\Browser**
If active a trace will be added each time a OPC client accesses to **NLopcTE** browser.
- **OPC interface\Others\OPC errors**
If active a trace will be added each time a generic OPC error occurs.

- **OPC reads/writes\Group\Group Events**
If active a trace will be added each time an OPC group event occurs.
An event can be : onDataChange, OnReadComplete, OnWriteComplete or OnCancelComplete.
- **OPC reads/writes\Group\Group Events filter**
Filter used to display traces when a group event occurs.
The filter works on the name of the group.
- **OPC reads/writes\Reads\OPC read (sync or async)**
If active a trace will be added each time an OPC client make a synchronous or asynchronous read.
Note : Cache reads are never traced.
- **OPC reads/writes\Group\OPC read item filter**
Filter used to display traces when an OPC read is received.
The filter works on the address of the item.
- **OPC reads/writes\Writes\OPC write (sync or async)**
If active a trace will be added each time an OPC client make a synchronous or asynchronous write.
- **OPC reads/writes\Group\OPC write item filter**
Filter used to display traces when an OPC write is received.
The filter works on the address of the item.
- **OPC reads/writes\Properties\Properties access**
If active a trace will be added each time an OPC client accesses to the properties of an item.
- **Network\LNS\LNS general**
If active a trace will be added for general LNS access.
- **Network\LNS\LNS change value event**
If active a trace will be added each time a LNS change value event occurs.
- **Network\LNS\LNS error event**
If active a trace will be added each time a LNS error event occurs.
- **Network\LNS\LNS events filter**
Filter used to display traces when a LNS event (change value or error) occurs.
The filter works on the name of the network variable.
- **Network\LNS\LNS mark**
If active a trace will be added each time a network variable is added/removed to/from the LNS monitoring engine.
- **Network\LNS\LNS events filter**
Filter used to display traces when a network variable is added/removed to/from the LNS monitoring engine.
The filter works on the name of the network variable.
- **Network\Network access\Network read**
If active a trace will be added each time a read is done on the network (a read can be a network variable fetch, a device query status, a configuration property read, a fast polling read).
- **Network\Network access\Network read error**
If active a trace will be added each time a read error occurs on the network (a read can be a network variable fetch, a device query status, a configuration property read, a fast polling read).

- **Network\Network access\Network read filter**
Filter used to display traces when a network read or a network read error occurs.
- The filter works on the name of the LNS object to read (device, network variable or configuration property).
- **Network\Network access\Network write**
If active a trace will be added each time a write is done on the network (a write can be a network variable or a configuration property update).
- **Network\Network access\Network write error**
If active a trace will be added each time a write error occurs on the network (a write can be a network variable or a configuration property update).
- **Network\Network access\Network write filter**
Filter used to display traces when a network write or a network write error occurs.
- The filter works on the name of the LNS object to write (network variable or configuration property).
- **Network\Device state\Device state verification**
If active a trace will be added each time **NLopcTE** verifies if a device is still absent on the network.
- **Network\Fast polling\Prepare fast polling**
If active a trace will be added each time the configuration of a device is checked to prepare fast polling.
- **Network\Fast polling\Fast polling read**
If active a trace will be added each time a fast polling read is done.
- **Network\Fast polling\Fast polling error**
If active a trace will be added each time a fast polling error occurs.
- **Network\Fast polling\Fast polling filter**
Filter used to display fast polling traces.
The filter works on the device name.
- **Network\Fast polling\Fast polling read details**
If active a trace is added for each network variable concerned by a fast polling.
- **Network\Fast polling\Fast polling details filter**
Filter used to fast polling read details.
The filter works on the network variables name.

FIRST USE WITH AN OPC CLIENT

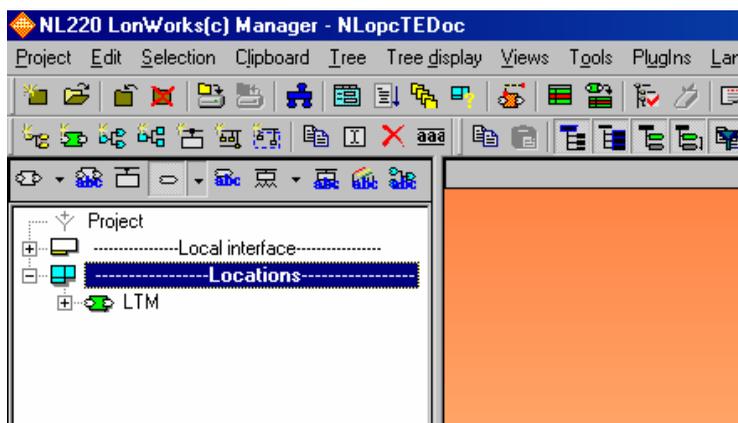
Softing OPC client

NLopcTE is provided with a the free Softing OPC client.



Note : To install this client you must use the custom installation of **NLopcTE**.

The example will used one LNS database with one device named LTM.



Picture 22
LNS Database for example

STEP 1: Configuring NLopcTE

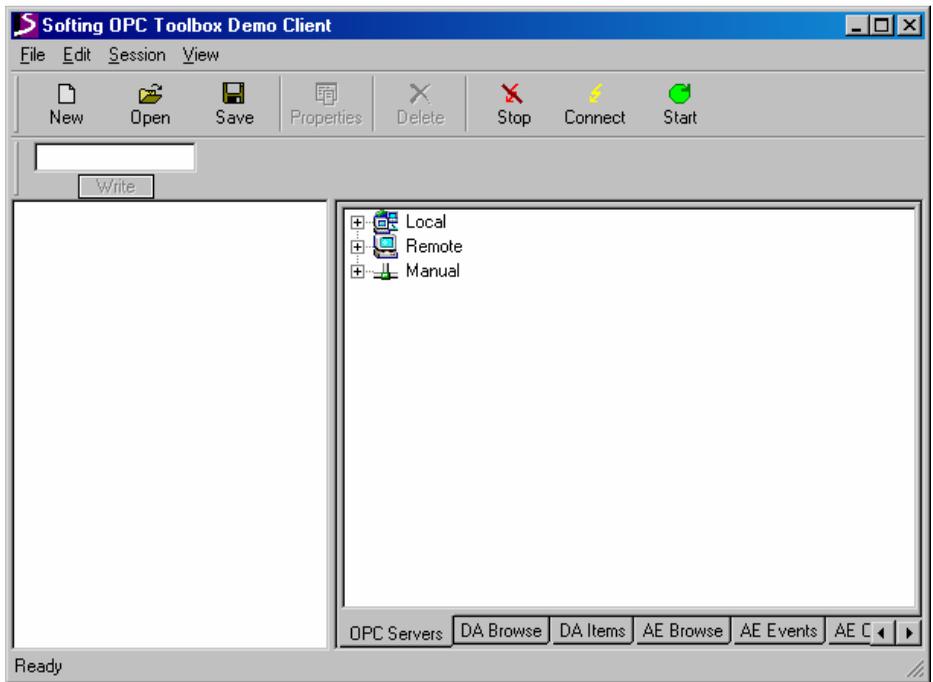
You need to configure **NLopcTE** to work with one or several LNS database (See Chapter NLOPCTE configuration p 17)

STEP 2: Starting the Softing OPC client

Execute the Softing OPC client.

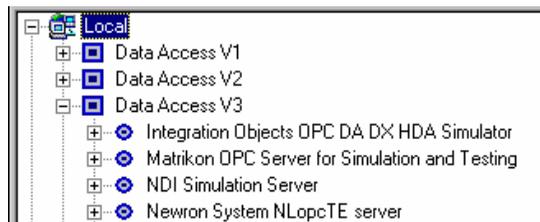
Note : You will find it in **Start\Programs\Softing OPC Toolbox Demo Client\Softing OPC Toolbox Demo Client**

The following windows appear:



Picture 23
Softing OPC Client User Interface

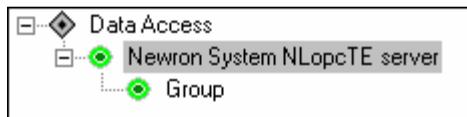
Open the branch Local, then the branch Data Access V3.



Picture 24
Browsing the OPC servers

Double click on **Newron System NLOpcTE server**.

In the left tree you will see the server with one OPC group created by the client.



STEP 3: Browsing items and adding one item

To browse the server click on the tabulation .

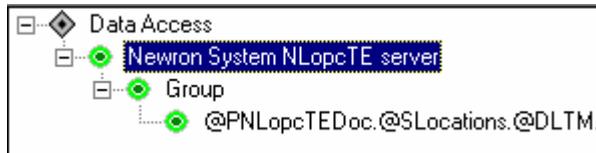
Open the branches of the browser to access the item you want to add :

Note : Items are preceded by a .



Picture 26
Browsing items

Double click on the item to add it in the group.



Picture 27
Item added to the group

STEP 4 : More operation in Softing OPC client

You can try to use the Softing OPC client for more operations like :

- Reading items
- Writing items
- Adding several items in one operation from the browser
- Saving/Loading a site
- Creating and configuring OPC groups

STEP 5: Close OPC Client

Stop the OPC client application.

