

# NLMODELER 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

---

<b>Introduction.....</b>	<b>6</b>
General information.....	6
<b>Installation of programs .....</b>	<b>7</b>
Configuration requirements.....	7
Installation.....	7
Installation of program .....	7
<b>Start with NLModeler .....</b>	<b>10</b>
Starting NLModeler .....	10
<b>NLModeler Overview .....</b>	<b>11</b>
General view .....	11
Tree view .....	12
Edit view.....	13
Output traces view .....	13
NLModeler menu .....	13
NLModeler Project menu .....	14
NLModeler Edit menu .....	14
NLModeler Tools menu .....	15
NLModeler ? (Help) menu .....	15
NLModeler Toolbars .....	15
<b>NLModeler Database .....</b>	<b>16</b>
Objects.....	16
Categories .....	16
Network object .....	16
Network object profile .....	18
Connections .....	25
Objects grouping rules .....	25
Objects interaction rules .....	28
Host command.....	31
Binding to host .....	32
Life areas .....	34
Office .....	34
Common area.....	39
Links .....	40
<b>Create first NLModeler Database .....</b>	<b>43</b>
Before start .....	43

Limited version .....43  
LNS Database .....43  
Relation between LonMark object .....46  
Step 1 start database creation .....47  
Step 2 create categories and profiles.....48  
Step 3 Configuring grouping rules .....52  
Step 4 Configuring interaction rules .....55  
Step 5 Configuration of the bindings with the host .....56  
Step 6 Configuration of an office .....60

---

# PICTURES

---

Picture 1 Software and Database organization .....	6
Picture 2 Install Step 1.....	8
Picture 3 Install Step 2.....	8
Picture 4 Install Step 3.....	8
Picture 5 Install Step 4.....	8
Picture 6 Install Step 5.....	9
Picture 7 Install Step 6.....	9
Picture 8 General view of NLModeler .....	11
Picture 9 Tree view .....	12
Picture 10 Network object profile in Edit view .....	13
Picture 11 Output trace view.....	13
Picture 12 NLModeler Project menu .....	14
Picture 13 NLModeler Edit menu.....	14
Picture 14 NLModeler Tools menu .....	15
Picture 15 NLModeler Help menu.....	15
Picture 16 Category properties window.....	16
Picture 17 Network Object properties window .....	17
Picture 18 Network Object profile properties window general folder .....	19
Picture 19 Network Object profile properties window plug-in folder .....	20
Picture 20 Network Object profile properties window Browser folder.....	21
Picture 21 Setting dialog box for network variable browser.....	22
Picture 22 Network Object profile properties window Browser folder.....	23
Picture 23 Example 1 of Browser .....	24
Picture 24 Example 2 of Browser .....	24
Picture 25 Network Object profile properties window Master selection folder .....	24
Picture 26 Object grouping rules properties window .....	26
Picture 27 Objects interaction rules properties window .....	28
Picture 28 Interactions Master to Master.....	29
Picture 29 Interactions Master to all.....	29
Picture 30 Interactions Master to slaves .....	29
Picture 31 Interactions All to master .....	29
Picture 32 Interactions All to all.....	30
Picture 33 Interactions All to slaves .....	30
Picture 34 Interactions Slaves to master .....	30
Picture 35 Interactions Slaves to slaves .....	30
Picture 36 Interactions Slaves to all.....	31
Picture 37 Host command properties window.....	31

Picture 38 Bind to host properties window .....	33
Picture 39 Office profile properties windows general folder .....	35
Picture 40 Office profile properties windows Objects grouping rules folder....	36
Picture 41 Office profile properties windows Objects interaction rules folder..	37
Picture 42 Office profile properties windows Configurations folder .....	38
Picture 43 Office profile properties windows Host commands folder .....	39
Picture 44 Common area profile properties windows General folder .....	40
Picture 45 Office link profile properties windows General folder .....	41
Picture 46 Office link profile properties windows Object interaction rules.....	41
Picture 47 NLModeler in limited version.....	43
Picture 48 Dialog box in complete version .....	43
Picture 49 Light and Fan Coil module .....	44
Picture 50 LonMark objects about Light and Fan coil module .....	44
Picture 51 Blind module .....	45
Picture 52 LonMark objects about blind and light module .....	45
Picture 53 Sensor module .....	46
Picture 54 LonMark objects about sensor module .....	46
Picture 55 Relation between LonMark objects.....	47
Picture 56 Setting category properties .....	48
Picture 57 All categories are created .....	49
Picture 58 Dialog box for network object definition .....	49
Picture 59 Device template selection.....	50
Picture 60 LonMark object selection.....	50
Picture 61 Object profiles in database .....	51
Picture 62 Edit profile of Light .....	51
Picture 63 Geometric form setting for Light profile.....	51
Picture 64 Setting membership kind of areas .....	52
Picture 65 Contextual menu for grouping rule definition.....	52
Picture 66 Object's grouping rules window.....	53
Picture 67 Network variable selection about binding master to slave .....	53
Picture 68 dialog box Binding settings .....	54
Picture 69 Network variable selection about binding slave to slave.....	54
Picture 70 Light links in Object grouping rules tree .....	54
Picture 71 Object's grouping rules tree .....	54
Picture 72 Object's interaction rules window .....	55
Picture 73 Schema of binding All to Master.....	56
Picture 74 Network variable selection about binding All to master .....	56
Picture 75 Object's interaction rules tree .....	56
Picture 76 Schema of Host command .....	57
Picture 77 Schema of Bound to Host.....	57
Picture 78 Host command window .....	58
Picture 79 Bind to host window .....	59
Picture 80 Office profile setting window .....	61
Picture 81 dialog box of setting network configuration value .....	64

# INTRODUCTION

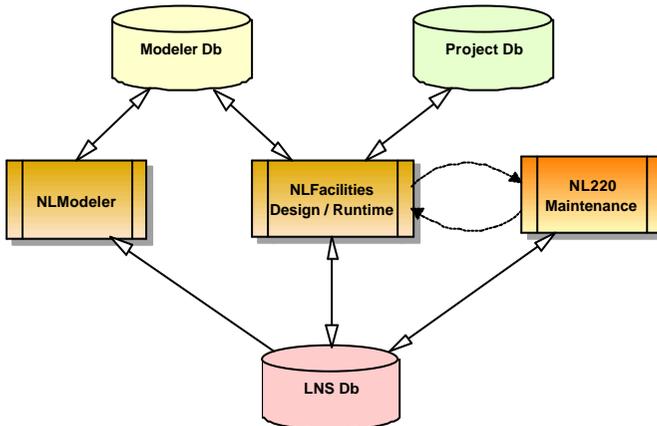
Thank you for choosing NLFacilities software member of NLSuite.

We are happy to help you in your LonWorks integration job. All softwares of NLSuite are often updated for correcting bugs and improve performances. We propose to you to check version on Web site [www.newron-system.com](http://www.newron-system.com).

## General information

---

In this package you have different tools describe follow.



Picture 1 Software and Database organization

**NLModeler** is a tool for managing relation between Nodes or LonMark objects in LNS Database. It defines network objects, space template for NLFacilities.

**NLFacilities** is a tool in two versions for managing map, network object and space template. In designer mode you draw a map and you put network object in it. In Runtime mode you draw office and other living space and it apply a space template in LNS Database and on Network.

**NL220** is a complement for maintaining your LonWorks network.

# INSTALLATION OF PROGRAMS

This section explains how to install the NLModeler included in NLFacilities installation.

## 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 NT, 2000, Xp	Windows NT, 2000, Xp
Computer	Pentium III 350 Mhz, 800 x 600 screen	Pentium III 750 Mhz, 1024 x 768 screen
Memory	64 M octets	128 M octets
Hard disk	100 Mo - 50 Mo for program - 0,5 to 8 Mo per project	200 M octets
CD ROM	Required for installation	Required for installation
Software	- LNS 3.0 or greater - NL220	- LNS 3.0 or greater - NL220
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: D:\index.htm
3. Picture 2 will appear on the screen.
4. Select **Software** on main Menu
5. Picture 3 will appear on the screen.
6. Select **NLFacilities** on center of screen.

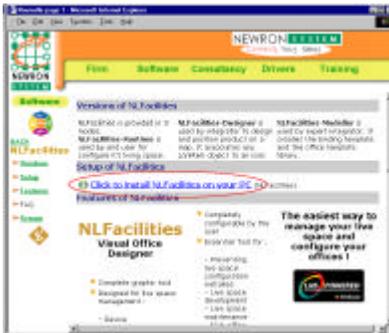
7. Picture 4 will appear on the screen.
8. Select **Click to install NLFacilities on your PC**, Picture 5 will appear on the screen.
9. Select **run this program fro its current location** and click on **OK** button.



Picture 2 Install Step 1



Picture 3 Install Step 2

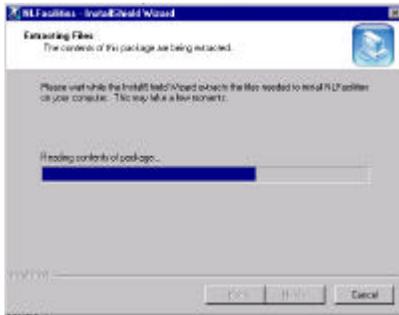


Picture 4 Install Step 3

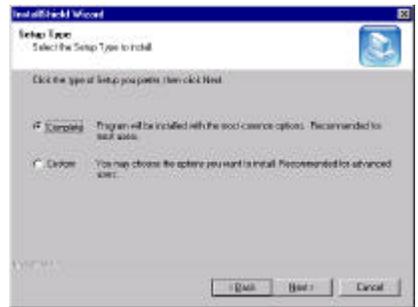


Picture 5 Install Step 4

The installation program will now be readied and Picture 6 will appear on the screen. Follow the instructions until you arrive at type of installation choice on Picture 7.



Picture 6 Install Step 5



Picture 7 Install Step 6

You have the choice between the following installations:

<b>Installation</b>	<b>Details</b>
Complete	Complete installation of NLFacilities and NLModeler.
Custom	You can choose the module to be installed

Table 2 Type of installation

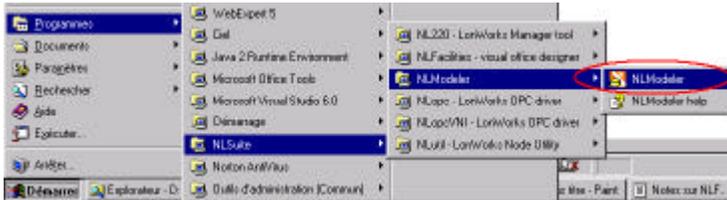
You should restart your PC at the end of the installation, according to the instructions

# START WITH NLModeler

## Starting NLModeler

---

To Start NLModeler, go to the **NLModeler** folder from the Start menu of windows.

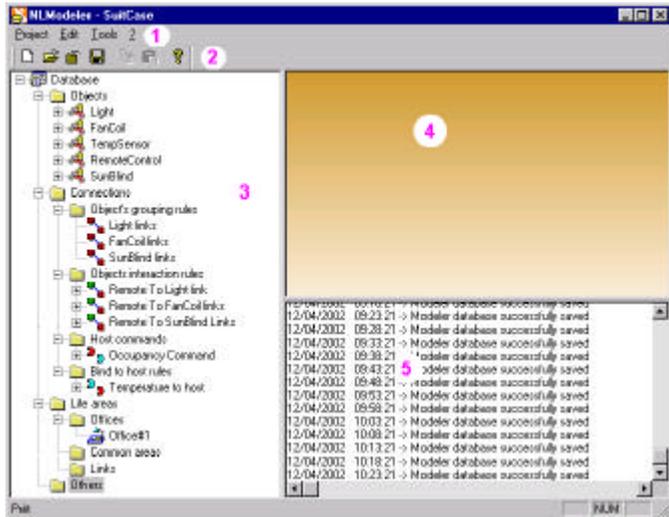


Picture 1 Start NLModeler

# NLMODELER OVERVIEW

## General view

---



Picture 8 General view of NLModeler

### 1 NLModeler Menu

Give access to any function.

### 2 NLModeler toolbars

Give quick access some functions.

### 3 Tree view

Displays in a tree the contents of database sorted by types of object and names.

### 4 Edit view

Displays detail of item showed in tree view. For editing you can use drag and drop functionality.

### 5 Output traces view

Displays history of commands since the start session.

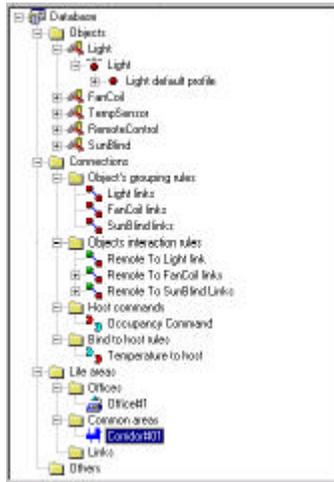
You can see launched command by user or made automatically by NLModeler and also error and warning.

Displays values of net points to be observed.

A second browser window is also available from the menus.

## Tree view

This view provides information on each item of a NLModeler database.



Picture 9 Tree view

Different icons are available in this view.

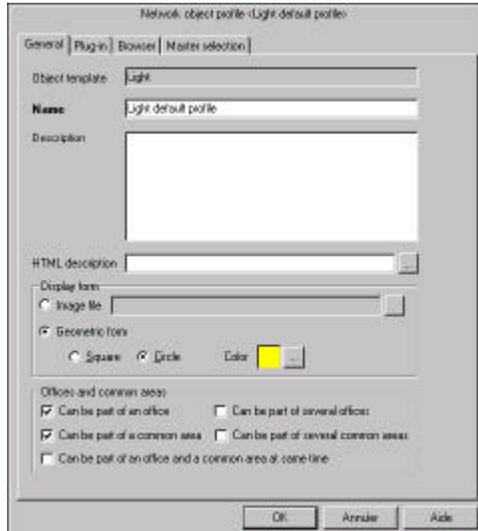
Each icon is followed by the name for explanation of item.

Icons	Representation
	Database
	Folder
	Category of product
	Network object
	Object profile
	Object's grouping rules
	Object interaction rules
	Host commands rules
	Binding to host rules
	Office profile
	Common area profile
	Links

## Edit view

---

This view depends of contextual item you edit.

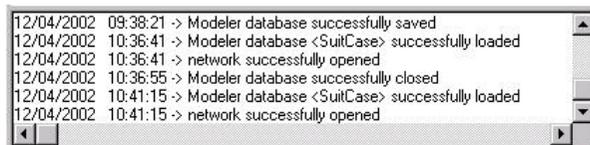


Picture 10 Network object profile in Edit view

## Output traces view

---

This view displays all the traces generated by NLModeler.



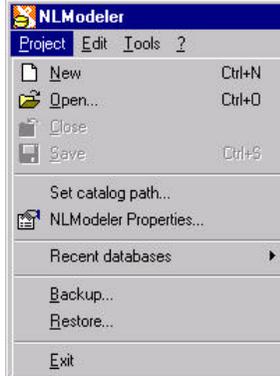
Picture 11 Output trace view

## NLModeler menu

---



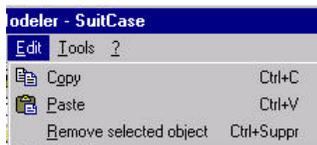
## NLModeler Project menu



Picture 12 NLModeler Project menu

Menu option	Icon	Description
New		Create new NLModeler database.
Open		Open an existing database
Close		Close current database
Save		Save current database
Set catalog path		Set SNVT catalog path.
NLModeler properties		Set general properties of NLModeler
Recent databases		List of database previously opened
Backup...		Backup database
Restore...		Restore database previously backedup
Exit		Quit NLModeler

## NLModeler Edit menu



Picture 13 NLModeler Edit menu

Menu option	Icon	Description
Copy		Copy object selected.
Paste		Paste previously copied object
Remove selected object		Remove selected object from database.

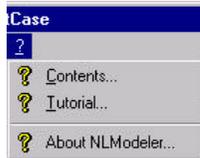
## NLModeler Tools menu



Picture 14 NLModeler Tools menu

Menu option	Icon	Description
Create new device template		Add new device template from XIF file into LNS database linked with NLModeler database.

## NLModeler ? (Help) menu



Picture 15 NLModeler Help menu

Menu option	Icon	Description
Contents		General Help about NLModeler
Tutorial		NLModeler tutorial in 8 steps
About NLModeler..		Dialog box about.

## NLModeler Toolbars



This toolbar provides general tools for database.

Icon	Short cut	Description
	Ctrl + N	Create a new NLModeler database.
	Ctrl + O	Open an existing NLModeler database.
	None	Close the current opened database.
	Ctrl + S	Save current NLModeler database.
	Ctrl + C	Copy object selected.
	Ctrl + V	Paste previously copied object.
	None	Dialog box about.

# NLMODELER DATABASE

## Objects

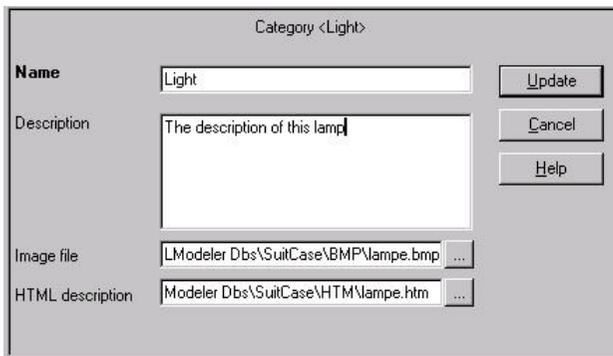
---

### Categories

Categories are used to organize **network objects** together.

In tree view  represent the category.

You must at least create one category to create network objects.



The screenshot shows a dialog box titled "Category <Light>". It contains four fields: "Name" with the value "Light", "Description" with the text "The description of this lamp", "Image file" with the path "LModeler Dbs\SuitCase\BMP\lampe.bmp", and "HTML description" with the path "Modeler Dbs\SuitCase\HTML\lampe.htm". To the right of the fields are three buttons: "Update", "Cancel", and "Help".

Picture 16 Category properties window

**Name** Name of the category

**Description** Description of the category (for information only).

**Image file** This image is used to display the category in **NLFacilities** tree and as the default for any **network object** created in the category. Any bitmap size is supported and will be converted to a 16\*16 bitmap. Optional information.

**HTML description** HTML file used to display information in **NLFacilities**. Optional information.

### Network object

The network object is the based item to create from a device template.

In tree view  represent the category.

A network object can be a full device template or one or several LonMark objects of a device template.

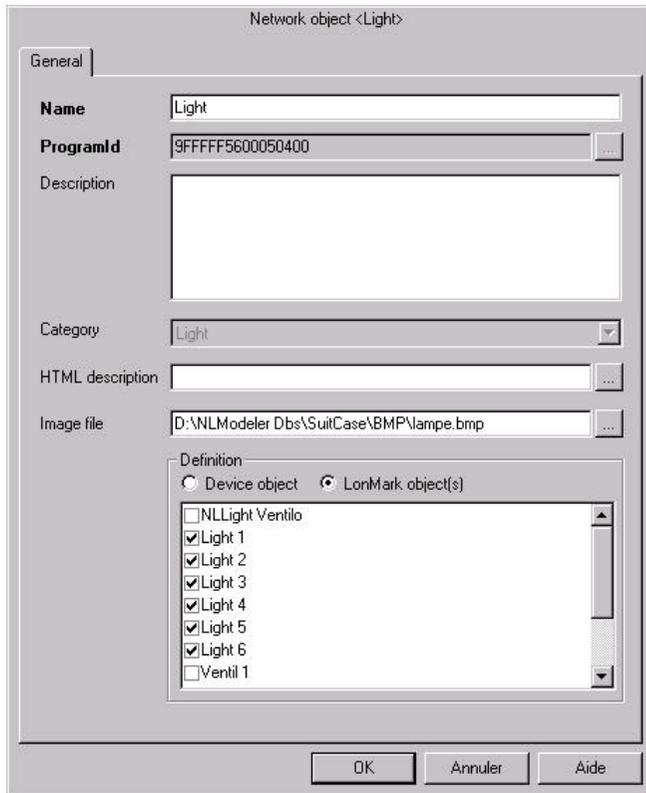
For a full device template all network variables will be available for binding configurations.

For a singleton LonMark object only the network variables of the object are available for binding configurations.

For a multi LonMark objects then compliant LonMark objects can be selected (same type, same network variables and same configurations). Only the network variables on the first LonMark object are used for bindings configuration. NLFacilities will use the correct network variable of the correct LonMark object when making bindings.



Note that it is not the network object used for binding configurations but one of its profile.



Picture 17 Network Object properties window

**Name** The name of the network object. Must be unique.

**ProgramId** ProgramID of the device template of the network object. You cannot change it after creation.

**Description** The description of the network object. Optional information.

**Category** Category the network object belongs to.

**HTML description** HTML file used to display information in **NLFacilities**. Optional information.

**Image file** Image file used as default for any profile of this network object or used as bitmap in tree when object has several profiles. Optional information.

**Definition** The type and items of the network object from a device template. You cannot change it after creation.

## **Network object profile**

The profile is derived from a network object.

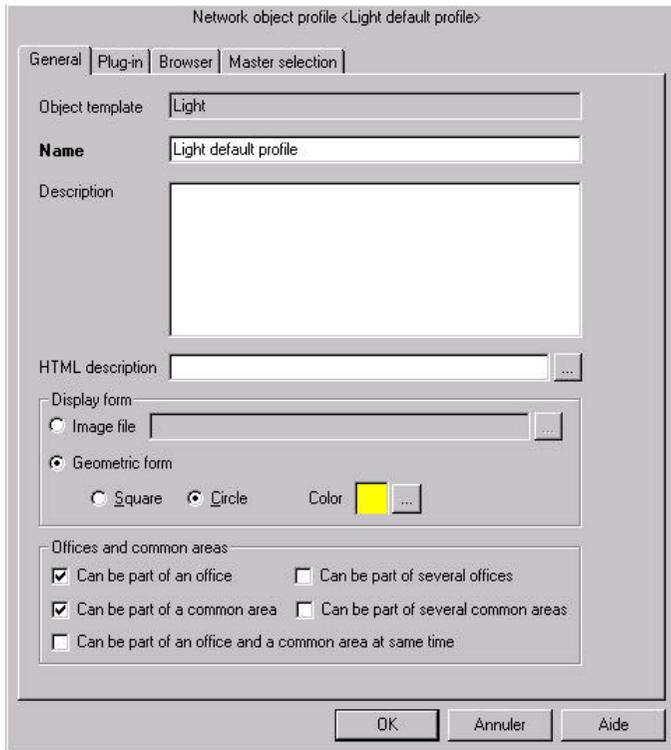
You can define in a profile its bitmap, the way it works with offices, its plug-ins and its browser configuration.

You can create one or several profiles for each network object.

Network object profile is the based item used for configuring bindings so you need at least one profile for each network object you want to use.

Note that when creating a network object a default profile is automatically created.

## General folder



Picture 18 Network Object profile properties window general folder

**Object template** Name of the profile's network object. You can't modify this information.

**Name** The Name of the profile

**Description** The description of the profile.

**HTML description** HTML file used by **NLFacilities** to display information about profile. Optional information.

**Display form** Type of display when the object is added in a view in **NLFacilities**. This can be a bitmap path or a form. For a bitmap you must select any compliant bitmap file. For the form you must select square or circle and the color of the form.

**Office and common area** Methods supported by the profile with offices and common areas.

**Can be part of an office** If checked the object profile can be added in an office.

**Can be part of a common area** If checked the object profile can be add in a common area.

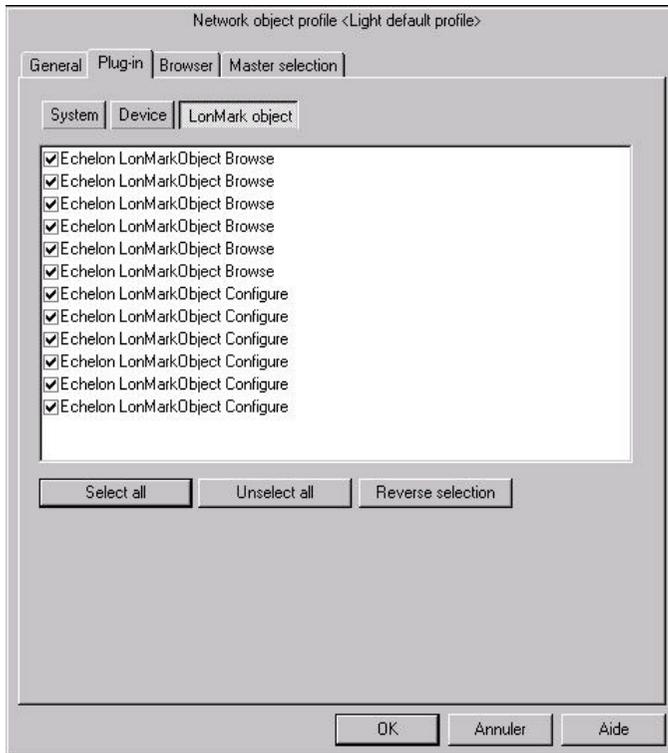
**Can be part of several offices** If checked the object profile can be shared by several offices.

**Can be part of several common areas** If checked the object profile can be shared by several common areas.

**Can be part of an office and a common area at same time** If checked the object profile can be shared by an office and by a common area.

### ***Plug-in folder***

In this part you can select the plug-ins supported and available in NLFacilities for this profile.



Picture 19 Network Object profile properties window plug-in folder

There are four types of plug-ins.

**System**            General project LNS plug-ins

**Device**            Specific or generic device plug-ins

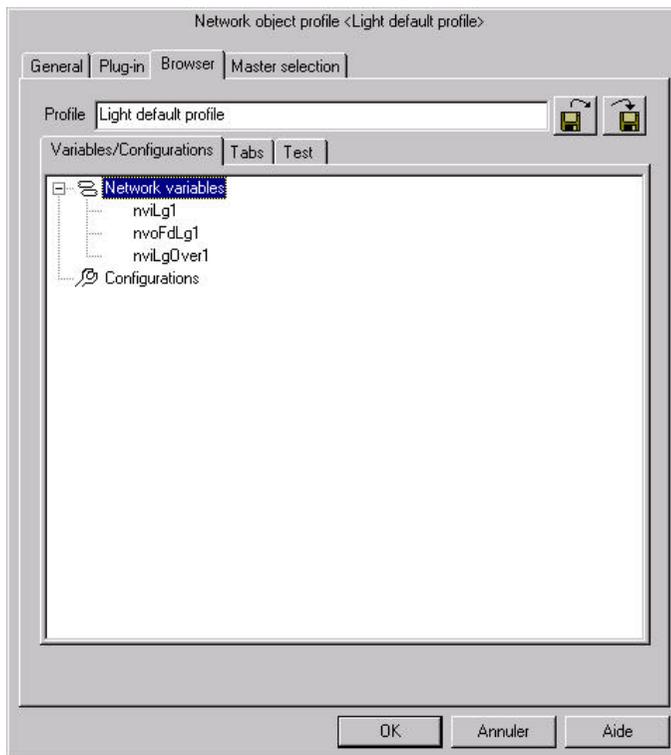
**LonMark object**    Specific or generic LonMark plug-ins

Each time you select a type NLModer display the available plug-ins register in the LNS database used by NLModer.

To enable a plug-in you must check it in the list.

### **Browser folder**

This part is used to configure the build-in browser of NLFacilities for this profile.



Picture 20 Network Object profile properties window Browser folder

**Profile** Name of the profile to be saved on disk.

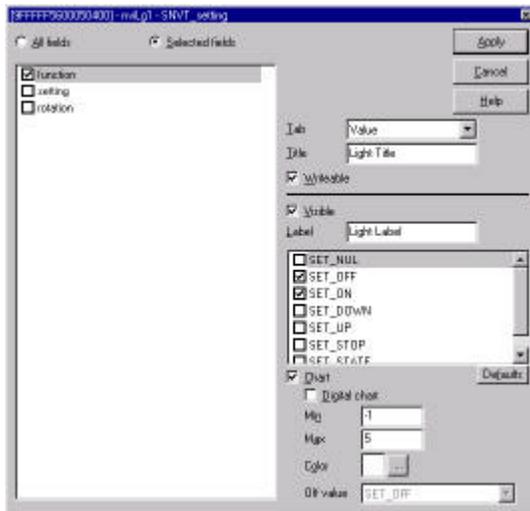


**Load** Click on this to reload the profile from disk.



**Save** Click on this to save the profile. This is optional because profile will be automatically saved when you validate the window.

If you double-click on network variable a dialog box appear.



Picture 21 Setting dialog box for network variable browser

**All fields** If checked all field of SNVT are show in NLFacilities browser.

**Selected fields** If checked, only field selected are showed in NLFacilities browser.

**Tab** Name of tab where fields selected in left list are visible in NLFacilities browser

**Title** Name of variable in Tab defined above.

**Writeable** Allow the modification of the value.

**Visible** If checked, this filed is visible in NLFacilities browser and it is checked in left list.

**Label** Name of variable in chart.

**Chart** Allow the chart with this field.

**Digital chart** Allow the digital chart with this field.

**Min** Minimum value for chart

**Max** Maximum value for chart

**Color** Color of the drawing value in graph.

### Variable/Configurations folder

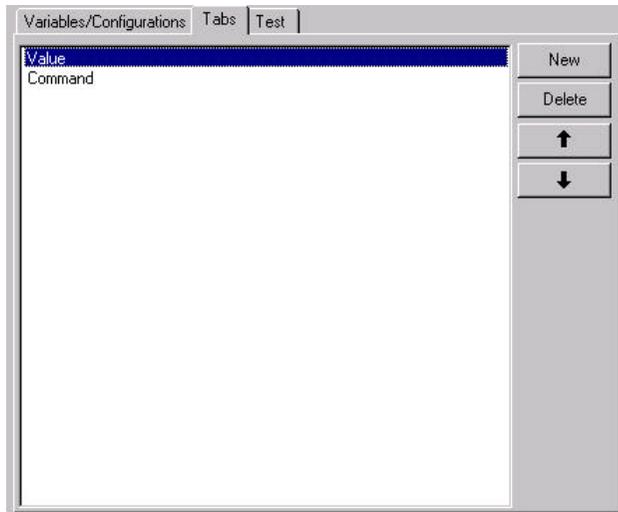
The list displays all available network variables and configurations of the profile. If the profile is a device profile then all network variables and configurations of the device template are available.



If the profile is a (or several) LonMark object then only network variables and configurations of the LonMark object are available in the list. Picture 20

### Tabs folder

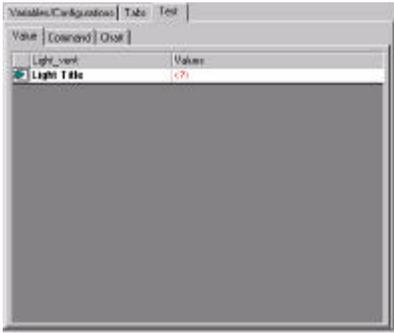
The Tabs folder is the list of tabs created in the browser. Tabs can be used to organize network variables and configurations in browser.



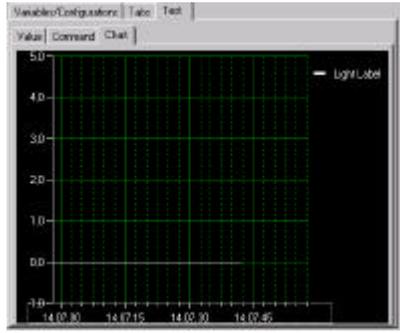
Picture 22 Network Object profile properties window Browser folder

### Test folder

Click on test to test the result in a fake browser.



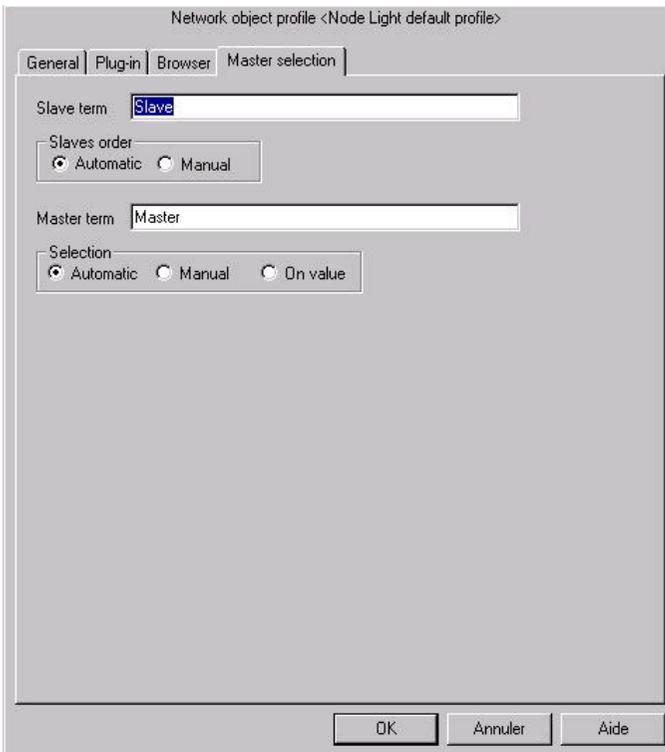
Picture 23 Example 1 of Browser



Picture 24 Example 2 of Browser

### ***Master selection folder***

In this part you can select the way to select the master and the order of the slaves for this profile.



Picture 25 Network Object profile properties window Master selection folder

**Slave term** Prefix used in **NLFacilities** to display slaves order.

**Slaves order** Way to define the order of the slaves.

**Automatic** The order is defined by **NLFacilities** without user interaction.

**Manual** The order is defined by the user under **NLFacilities** when making an area.

**Master term** String used in **NLFacilities** to display master selection.

**Selection** Way to select the master.

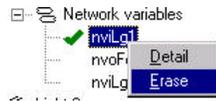
**Automatic** The master is defined by NLFacilities without user interaction

**Manual** The master is defined by user under NLFacilities when you define an area.

**On value** The master is defined depending of values of network variables and/or configurations.

**Conditional operator** The way to verify the value. Can be **equal** or **different**.

**Conditional value** The list displays the network variables and configurations of the profile. To define a value of a network variable or a configuration double click on the item in the list. To remove a value of a network variable or a configuration right click on the item in the tree and select option **Erase**.



## Connections

---

### Objects grouping rules

Object's grouping rules are used to configure bindings between same network object's profiles.

Object's grouping rules always work on a network object profile.

In tree view  represent the object grouping rules.



An example:

**NLModeler** have a network object profile Light.

Now under **NLFacilities** you can create an area (office or common area) including several objects of this profile.

The **object's grouping rules** are used here to make the bindings between objects with same profile.

It exists two types of object's grouping rules.

**Chain link** These bindings describe the links between objects of same profile in same area. It is used to "group" the objects in order to work together in same area.

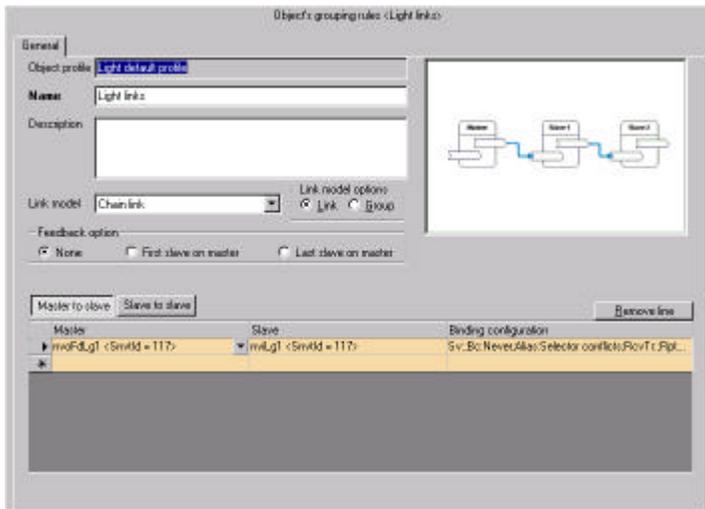
**Feedback link** These bindings are used to make feedback process between objects of same profile in same area.

Note that you not always need the both feature and if fact only **Chain link** is often used.

If you want to have a **Chain link** and a **Feedback link** for the same network object profile you must create TWO **Object's grouping rules** for this profile.

Numbered options are available to correctly configure your bindings between objects.

### **Editing Object grouping rules**



Picture 26 Object grouping rules properties window

**Name** Name of the object's grouping rules.

**Description** Description of the object's grouping rules.

**Link model** You can do a **Chain link** or a **Feedback link**.

**Link model options** (only available for **Chain link**)

**Link** In this mode the first object (**master**) send variables to the first slave.

**Group** In this mode the first object (**master**) send variables to all slaves in one shot.

**Feedback option** (only available for **Chain link**)

**None** No feedback.

**First slave on master** First slave send feedback values to master.

**Last slave on master** Last slave send feedback values to master. Determination of the first and last slaves are determined by the order of the slaves (see **Master selection**).

**Options** (only available for **Feedback link**)

**All to all** All objects (master and slaves) send feedback to all others.

**Slaves to all** All slaves (master excluded) send feedback to all others (master included).

**Slaves to slaves** All slaves (master excluded) send feedback to all others except the master.

**Slaves to master** All slaves (master excluded) send feedback to the master only.

**Last slave to master** Last slave send feedback to the master only.

**Turnaround** (only available for **Feedback link**) Each checked the network variable(s) are send from device to themselves as a LonWorks turnaround. Not available for a **Slaves to master** or **Last slave to master** mode.

**Bindings (grid)** In the grid you must select the source output network variable, the source inptu network variable and the binding parameters.

For a **Chain link** you can set the bindings settings for :

Master to slave

Set the bindings settings for the network variables send from master to slaves.

Slave to slave

Set the bindings settings for the network variables send from slave to slave. Set the bindings settings for the network variables send from slave to slave.

Feedback

Set the bindings settings for the feedback option (only if you select **First slave on master** or **Last slave on master**).

For a **Feedback link** you have no special options and you have to define the source and destination network variables of the binding.

For any type of binding you must set the source network variable, the destination network variable and the binding settings.

## Objects interaction rules

Objects interaction rules are used to configure bindings between two different [network object profile](#).



An example

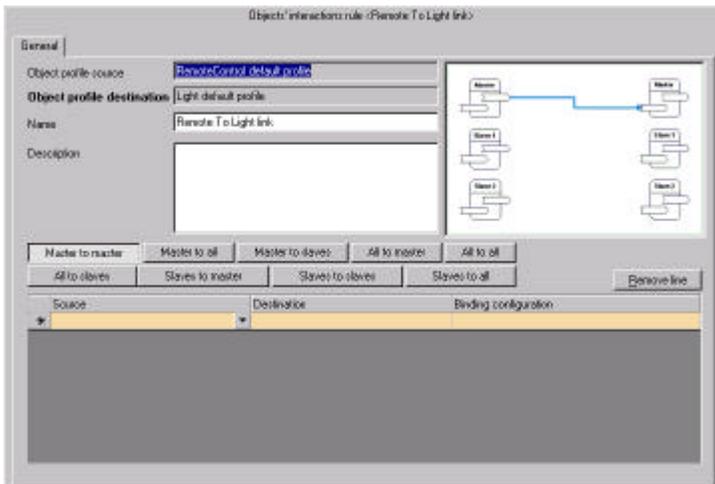
**NLModeler** have a network object profile Light and a network object profile Presence.

The second (Presence) must send the occupancy value to the first (light) when present in same area (office or common area).

The **objects interaction rules** are used here to make the bindings between the two different profiles.

### Editing Objects interaction rules

For each binding you must select the type:



Picture 27 Objects interaction rules properties window

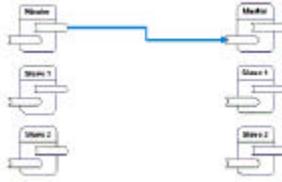
**Object profile source** The module provider of information.

**Object profile destination** The module receiver of information.

**Name** The name of the object's interaction rule.

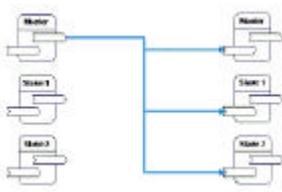
**Description** The description of the object's interaction rules. Optional information.

**Master to master** The network variable(s) is(are) send from the source master to the destination master.



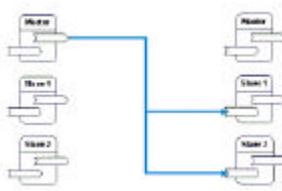
Picture 28 Interactions Master to Master

**Master to all** The network variable(s) is(are) send from the source master to all destination objects.



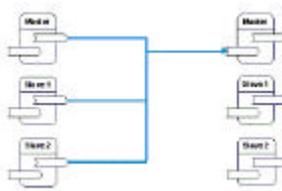
Picture 29 Interactions Master to all

**Master to slaves** The network variable(s) is(are) send from the source master to the destination slaves (destination master is excluded).



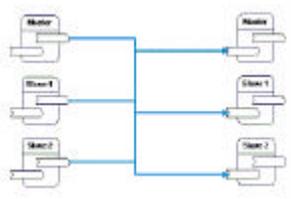
Picture 30 Interactions Master to slaves

**All to master** The network variable(s) is(are) send from all source objects to the destination master.



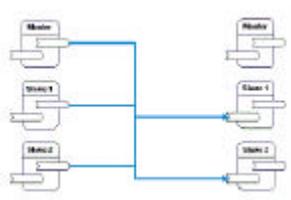
Picture 31 Interactions All to master

**All to all** The network variable(s) is(are) send from all source objects to the all destination objects.



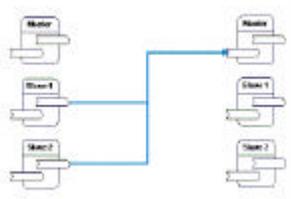
Picture 32 Interactions All to all

**All to slaves** The network variable(s) is(are) send from all source objects to the destination slaves (destination master is excluded).



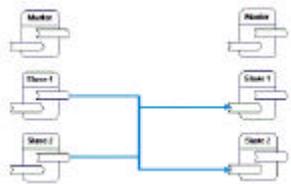
Picture 33 Interactions All to slaves

**Slaves to master** The network variable(s) is(are) send from all source slaves (source master is excluded) to the destination master.



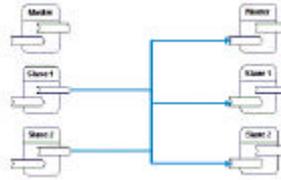
Picture 34 Interactions Slaves to master

**Slaves to slaves** The network variable(s) is(are) send from all source slaves (source master is excluded) to the destination slaves (destination master is excluded).



Picture 35 Interactions Slaves to slaves

**Slaves to all** The network variable(s) is(are) send from all source slaves (source master is excluded) to all destination objects.



Picture 36 Interactions Slaves to all



Note that you can combine any settings. You can configure for example a binding in **Slaves to all** mode and another in **Slaves to slaves** mode.

For each type required you must set the source network variable(s), destination network variable(s) and the binding configuration.

To change the Binding configuration select the cell in the Binding configuration column and click on .

## Host command

Host commands are output network variables send from the local host PC to one or several network devices.

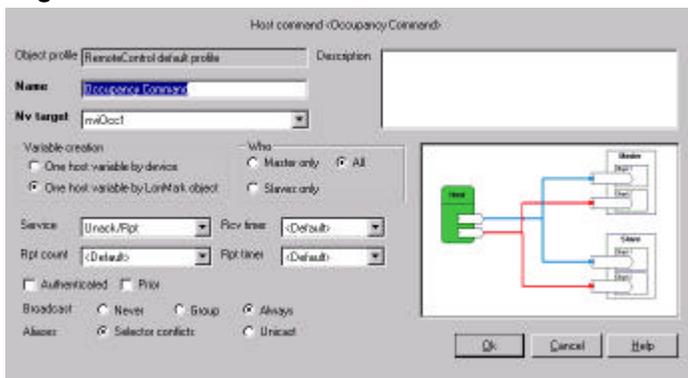
These host commands are use as derogate commands.

A host command is always create from a network object profile.



Note that in **NLModeler** user only configure settings of a host command. Under **NLFacilities** it is possible to create several host commands for only one host command created in the modeler.

### Editing host command



Picture 37 Host command properties window

**Name** The name of the host command.

**Nv target** The network variable destination for the command.

**Variable creation** This settings is used for [network object profile](#) using several LonMark objects. In this case it is possible that a host network variable must be send to several network variables on the same network device. This can cause aliases problem for devices not supporting aliases or without any free aliases.

**One host variable by device** This option force NLFacilities to create only ONE output network variable for the command. The destination devices have to support aliases if required.

**One host variable by LonMark object** This option force NLFacilities to create ONE output network variable FOR EACH LonMark object in the network object profile.

**Who** Objects which will receive the command.

**Master only** Only the master will receive the command.

**Slaves only** Only the slaves will receive the command.

**All** Master and slaves will receive the command.

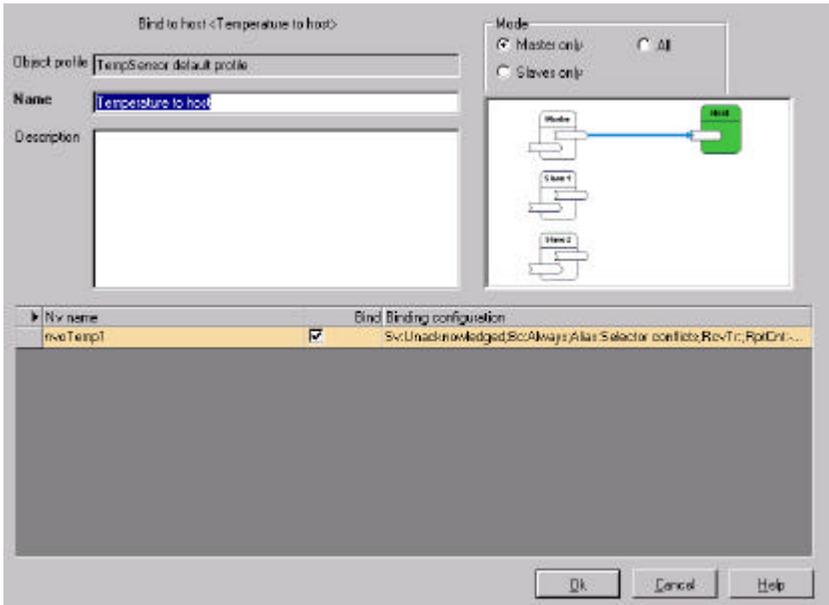
**Binding properties** Service, Rcv timer and so one are information set by expert for each binding.

## Binding to host

Binding to host can be used to automatically bind output network variables of network devices to the local host PC.

A binding to host is always created from a network object profile.

## Editing bind to host



Picture 38 Bind to host properties window

**Name** The name of binding.

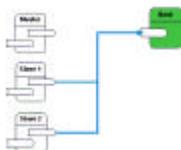
**Description** The description of the binding to host. Optional information.

**Mode** You can define here which objects will be bound to the host.

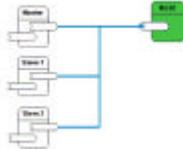
**Master only** Only the network variables of the master are bound.



**Slave only** Only the network variables of the slaves are bound.



**All** The network variables of master and slaves are bound.



In the grid are displayed all output network variables you can bind.

Simply check the network variable (Bind column) you want to bind to the host and enter the bindings settings.

## Life areas

---

### Office

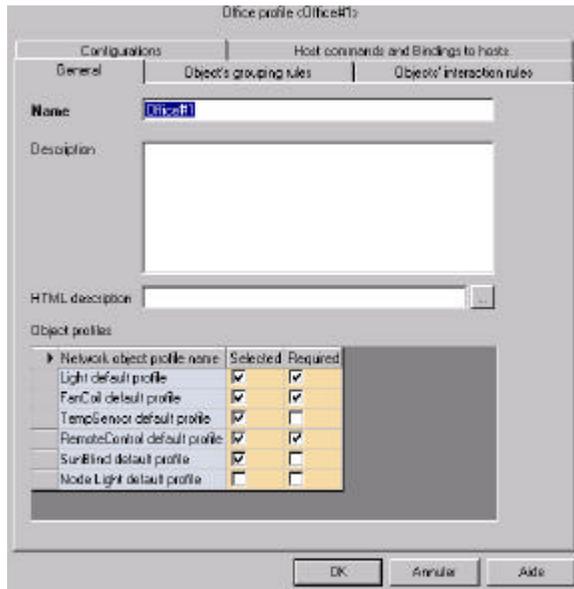
The office is the final unit used by **NLFacilities** to configure area.

When you set an office profile you will define what the office supports and uses.

This includes:

- The network object profiles
- The object's grouping rules
- The objects interaction rules
- The configurations
- The host commands
- The bindings to host

## Editing office profile



Picture 39 Office profile properties windows general folder

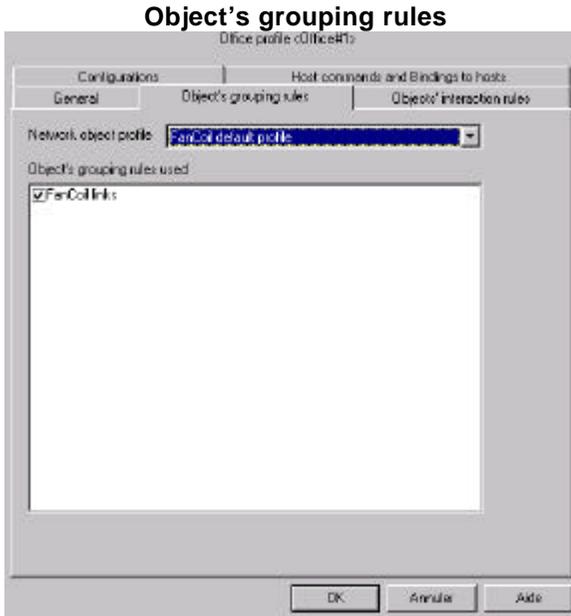
### General folder

**Name** The name of profile.

**Description** The description of profile. Optional information.

**HTML description** HTML file used by NLFacilities to display information.

**Object profiles** Here are selected the network object profiles supported by the office. After creation you cannot change the **Selected** column. Check the column **Required** if the office couldn't be create with at least one object of this profile.

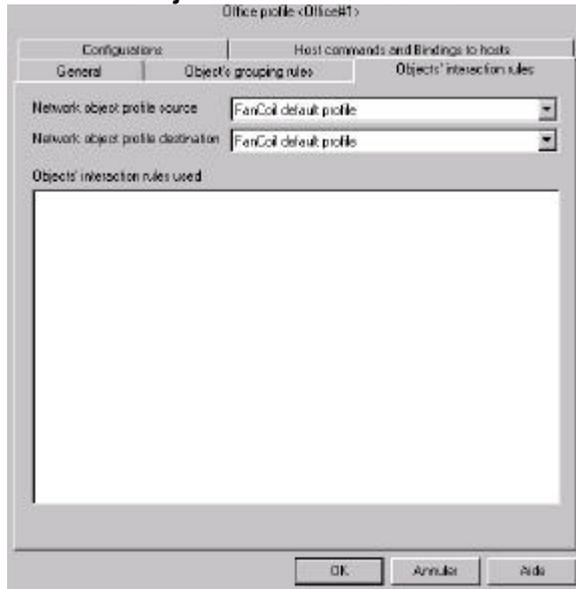


Picture 40 Office profile properties windows Objects grouping rules folder

You can select here the Object's grouping rules used by the office profile for each network object profile.

First select the Network object profile then the object's grouping rules used.

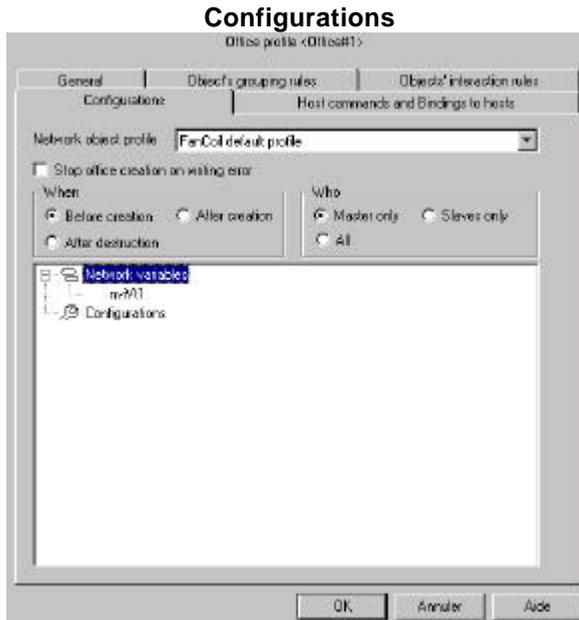
## Object's interaction rules



Picture 41 Office profile properties windows Objects interaction rules folder

You can select here the Objects interaction rules used by the office for each couple of network object profile.

Select first the **Network object profile source** then the **Network object profile destination**, then select the objects interaction rules used.



Picture 42 Office profile properties windows Configurations folder

You can define here input network variables and/or configurations to write on network device when creating or removing an office.

## When

**Before creation** Sets the values of network variables and/or configurations to write just before the office is created.

**After creation** Sets the values of network variables and/or configurations to write just after the office is created.

**After destruction** Sets the values of network variables and/or configurations to write just after the office is deleted.

## Who

**Master only** The values are only write on the master.

**Slaves only** The values are only write on the slaves.

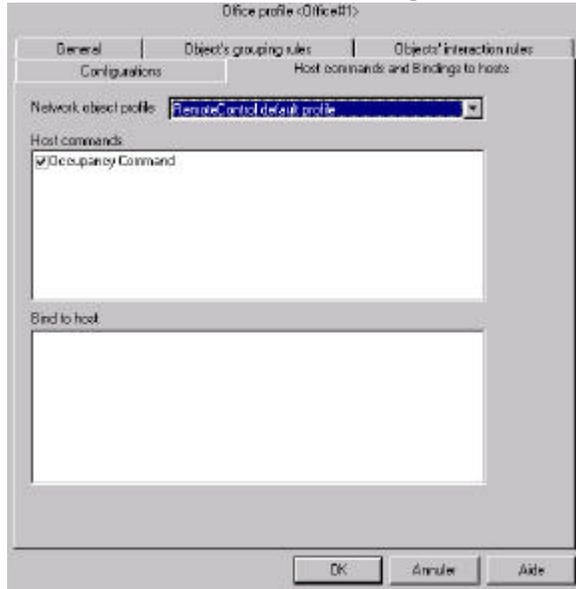
**All** The values are only write on master and slaves.

The list displays the input network variables and configurations you can change.

To set a value double click on the item in the list and enter the value.

To remove the value right click on the item and select the option **Erase**.

### Host commands and binding to hosts



Picture 43 Office profile properties windows Host commands folder

You can select here the host commands and bindings to host used by the office for each network object profile.

First select the Network object profile then check the bindings used by the office.

### Common area

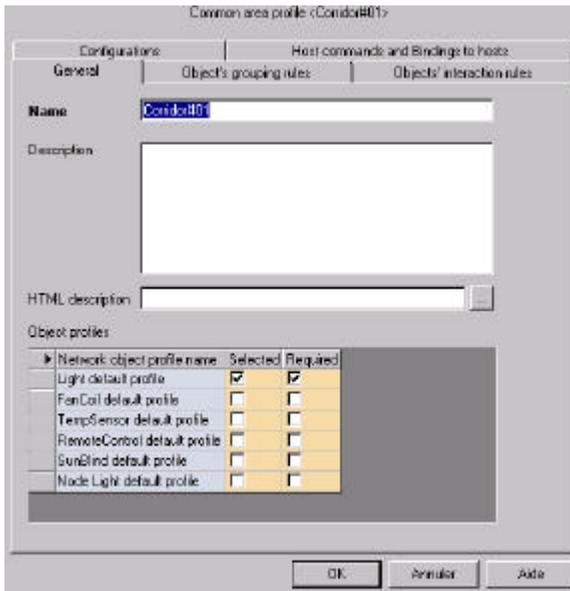
The common area is the final unit used by **NLFacilities** to configure common area.

When settings an common area profile you will define what the office supports and uses.

This includes:

- The network object profiles
- The object's grouping rules
- The objects interaction rules
- The configurations
- The host commands
- The bindings to host

## Editing common area profile



Picture 44 Common area profile properties windows General folder

All options and actions are same than office profile.

## Links

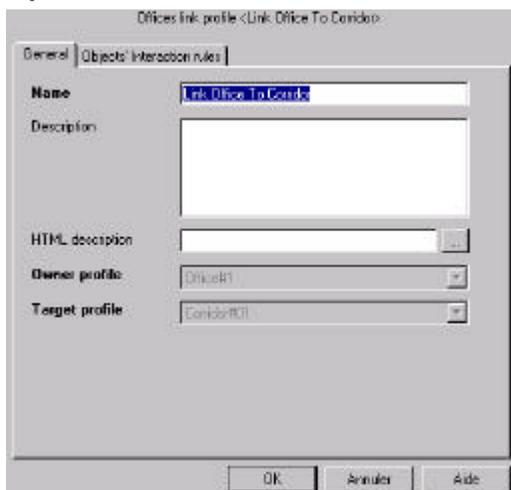
Under **NLFacilities** you can link any office to a common area.

In this case **NLFacilities** will use the bindings configuration sets as the links between the office and the common area.

The bindings configuration for this type link is based on a **Objects interaction rules**.

First create the **Objects interaction rules** you want to use and then create a link between the office and the common area using these rules.

## Editing link profile



Picture 45 Office link profile properties windows General folder

### General folder

**Name** The name of link.

**Description** The description of link. Optional information.

**HTML description** HTML file used by **NLFacilities** to display information

**Owner profile** Office used by the link.

**Target profile** Common area used by the link.

### Object's interaction rules



Picture 46 Office link profile properties windows Object interaction rules

You can select here the objects interaction rules used when an office is linked to a common area.

# CREATE FIRST NLModeler DATABASE

## Before start

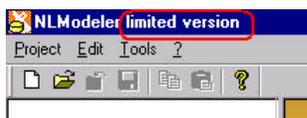
---

### Limited version

Before you start, check if you have a LNS 3.0 manager tool installed on your PC.

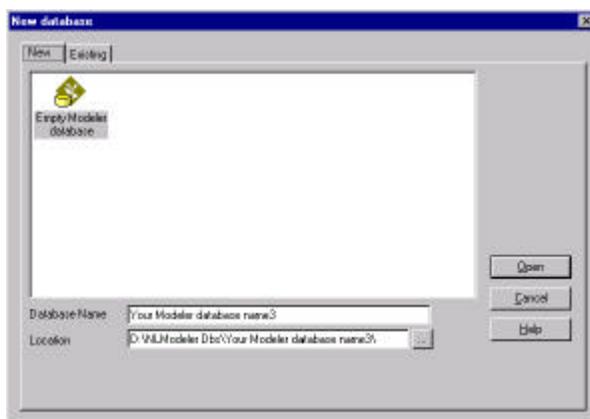


If you start NLModeler in demonstration mode for the first time, you can create and work **ONLY** on "DEMO" Database.



Picture 47 NLModeler in limited version

In complete version you can set the name of your NLModeler database see Picture 48.



Picture 48 Dialog box in complete version

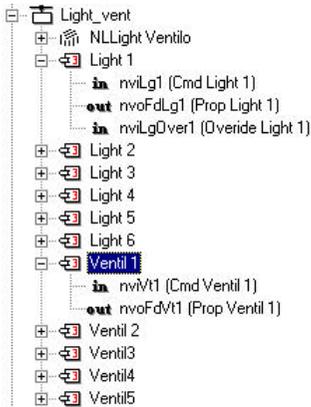
### LNS Database

For designing you database, you need products and definition of these products.

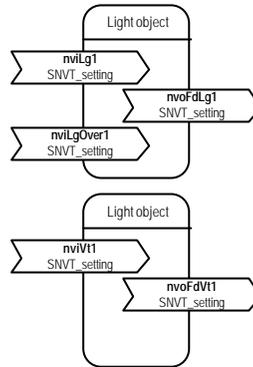
In demonstration case you have three nodes. In NL220 Database DEMO you have also three devices templates.

## Light and Fan coil module

The first device template handle light named **Light** and Fan coil named **Ventil** like ventilation.



Picture 49 Light and Fan Coil module

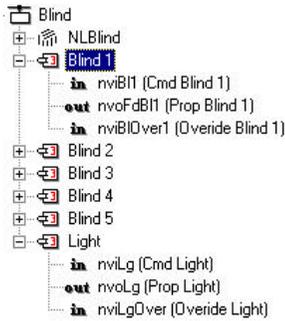


Picture 50 LonMark objects about Light and Fan coil module

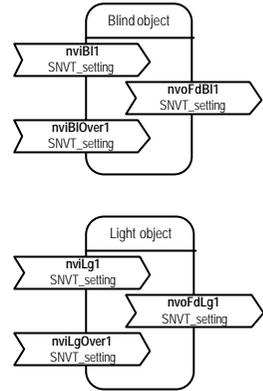
- NviLg1** Main input where module receives order for switch on or off the light.
- NvoFdLg1** Output sends for feed back or next light the current state of light received in main input.
- NviLgOver1** Secondary input for locally changes the state of light.
- NviVt1** Main input where module receives order for define speed of fan.
- NvoFdVt1** Output sends for feed back or next fan the current speed of fan received in main input.

## Blind module

The second module handles sunblind and one Light.



Picture 51 Blind module

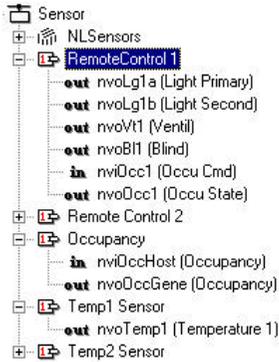


Picture 52 LonMark objects about blind and light module

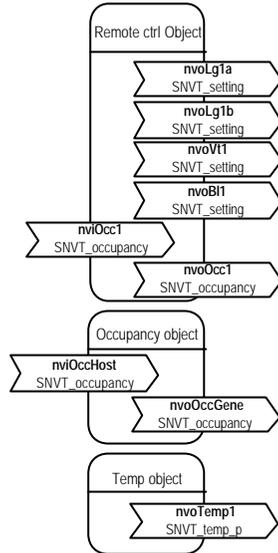
- NviBI1** Main input where module receives order up or down for sunblind.
- NvoFdBI1** Output sends for feed back or next sunblind the current action of sunblind received in main input.
- NviBIOver1** Secondary input for locally action on sunblind.
- NviLg1** Main input where module receives order for switch on or off the light.
- NvoFdLg1** Output sends for feed back or next light the current state of light received in main input.
- NviLgOver1** Secondary input for locally changes the state of light.

**Sensor module**

The third module handles remote control module and temperature sensor.



Picture 53 Sensor module

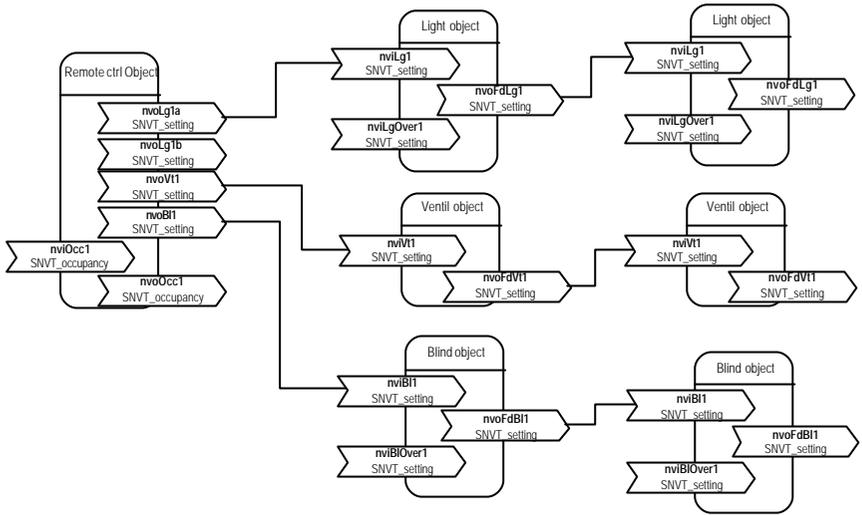


Picture 54 LonMark objects about sensor module

- NvoLg1a** Main output to control office light.
- NvoLg1b** Secondary output to change light state in accordance with occupancy.
- NvoVt1** Main output to control office fan speed.
- NvoBI1** Main output to control office sunblind.
- NviOcc1** Input for controlling occupation of building and it manages the state of all main output control in office.
- NvoOcc1** Output to define real state of the remote control occupancy.
- NviOccHost** Input from host for general control about occupancy.
- NvoOccGene** Output to define occupancy.
- NvoTemp1** Output to simulate the temperature variation.

## Relation between LonMark object

In Picture 55 you see an example of relations between a remote control and two LonMark object of each kind.



Picture 55 Relation between LonMark objects

We propose to design an NLModeler database for this project. In complement we add a binding to the host for temperature sensor.

## Step 1 start database creation

### Create project

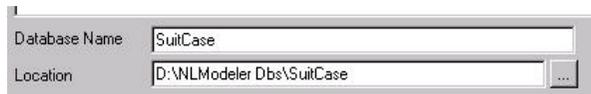
We will first create the project.

Select option **New** of menu **Project**.



### Set database name

Enter the name of the project and the path of the project database.



And then click on

Now **NLModeler** asks to select a LNS database.

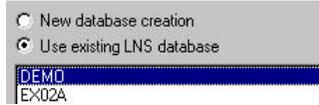
**NLModeler** needs LNS database to work on device templates.

Remember here that **NO** devices are required in the LNS database, only device templates are used.

You can either select an existing database or create a new one.

## Select a LNS database

Select the option **Use existing LNS database**.



And then click on .

The project is now ready to be configured.

## Step 2 create categories and profiles

We will now create the object profiles for Light, Fan Coil, Blind, Temperature and Remote Control devices.

### Create categories

We must first create categories.

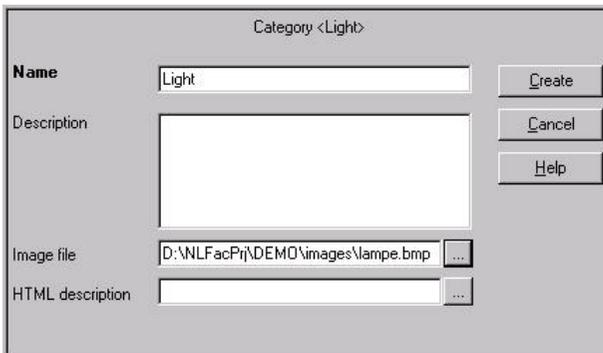
Categories are used to group objects and object profiles. Using categories is optional but it is required to create at least one.

We will create three categories: **Light**, **FanCoil**, **Blind**, **TempSensor** and **RemoteControl**.

Right click on **Objects** branch and select **New category ...** option.



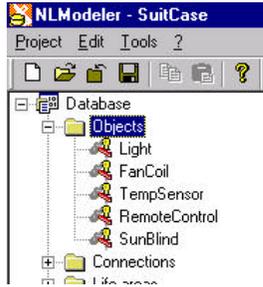
Enter **Lights** as in Name field, choose the image file for this categorie and then click on .



Picture 56 Setting category properties

Do the same for others categories.

You have now your five categories present in the tree.



Picture 57 All categories are created

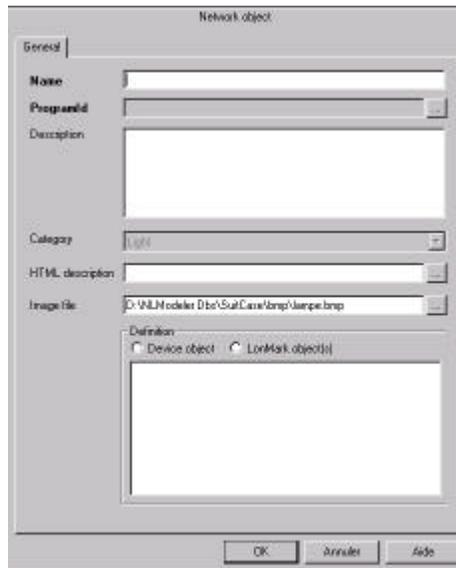
### ***Creating network objects***

We are now able to create the first object. It will be the light object.

Right click on category **Light** and select option **New network object ...**



A dialog box appear on right window like Picture 58.



Picture 58 Dialog box for network object definition

### ***Set name of object***

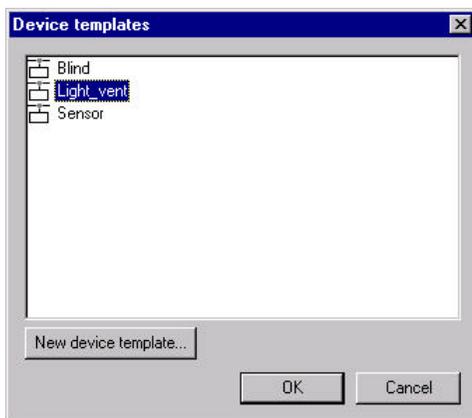
In **Name** field enter **Light**.



Click on the  button on the right side of the **ProgramID** field.



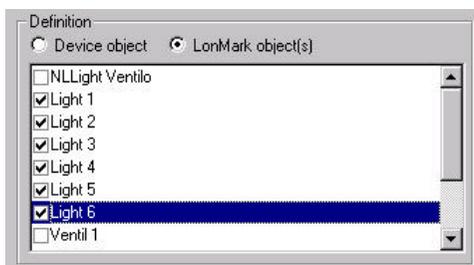
Select the device template **Light\_vent** and click on .



Picture 59 Device template selection

We must now select the LonMark objects supported by this object.

**NLModer** supports the device itself (all network variables) or any LonMark objects to use as a network object.



Picture 60 LonMark object selection

Now click on  to create the new object.

Now do the same for the SunBlind, FanCoil, Temperature sensor and Remote control device.

Click on  to save your project here.

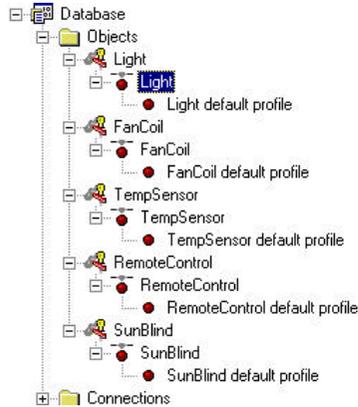
### **Configuring the profiles**

For each created object we need now to configure a profile.

**NLModer** allows to create several profiles for each object.

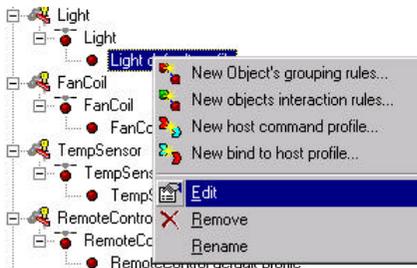
In our example we need only one profile by object and we will use the profiles automatically generated by NLModer.

Open in the tree each category and then each object in the categories.  
 You can see that three profiles are available named **Light default profile**, **FanCoil default profile** and so one.



Picture 61 Object profiles in database

Right click on the light profile (**Light default profile**) and click on **Edit** option.



Picture 62 Edit profile of Light

We must select the display form. This display settings are used by **NLFacilities** to display the object in a view.

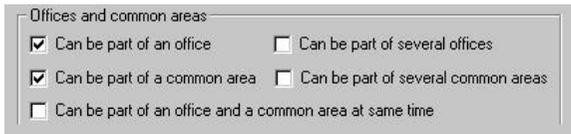
Because we do not have any bitmap we will use a geometric form.

Select **Circle** and select a yellow color.



Picture 63 Geometric form setting for Light profile

Now verify that **Can be part of an office** and **Can be part of a common area** are checked.



Picture 64 Setting membership kind of areas

Click on  to validate the profile.



You have three other tabs available in profile window:

**Plug-in** Configure the plug-ins available under NLFacilities for the profile

**Browser** Configure the build in browser of NLFacilities for the profile

**Master selection** Configure the mode to select the master and slaves order for the profile

We will keep the default values for all these settings.

Do the same for the **other default profile**.

Click on  to save your project here.

### Step 3 Configuring grouping rules

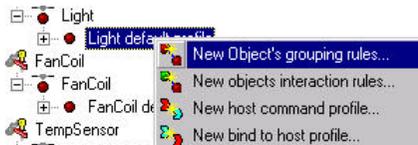
We have now create the five objects profiles **Light default profile**, **FanCoil default profile**, **TempSensor default profile**, **RemoteControl default profile** and **SunBlind default profile**.

Light, FanCoil and SunBlind profiles have bindings between themselves when belonging to an area. These kinds of bindings are called **Object's grouping rules**.

#### *Configuring Object's grouping rules for Light objects*

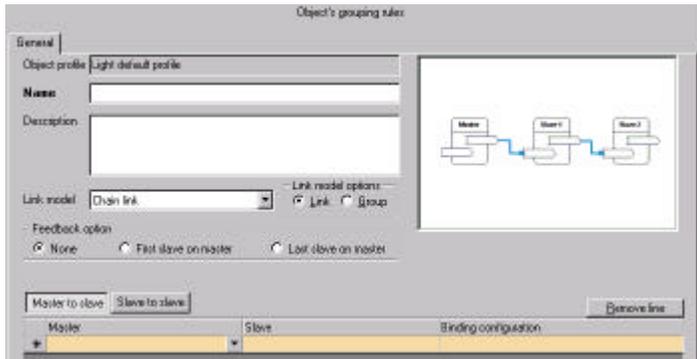
When light objects are added in an area they must be linked one to each other.

Right click on the **Light default profile** item and select **New Object's grouping rules** option.



Picture 65 Contextual menu for grouping rule definition

The start edition of grouping rule show you a windows to the right.  
Picture 66



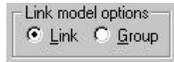
Picture 66 Object's grouping rules window

We will name these rules **Lights Links**

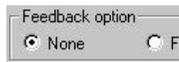
The Link Model must be **Chain link** (the other type of link is **Feedback link** but we does not have to create any feedback).



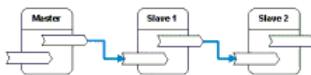
The **Link model options** will be selected as **Link** (the other option is **Group** but our lighting objects use master to slave scheme).



In **Feedback option** select **None** because we do not need any feedback.



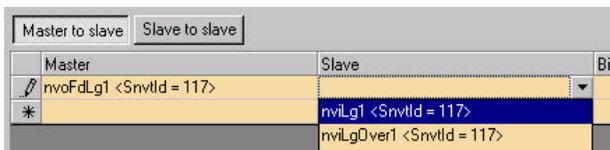
The result bitmap of our link is



Now we must select the network variables to be bound.

Click on **Master to slave**.

In **Master** column select **nvoFdLg1** network variable and in **Slave** column select **nviLg1** network variable like this:

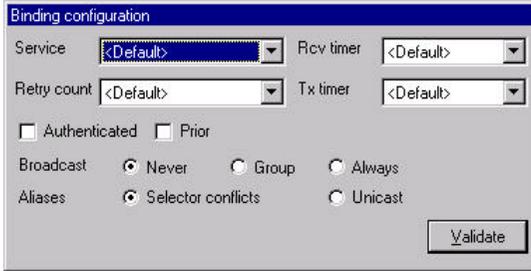


Picture 67 Network variable selection about binding master to slave

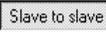
☞ The SNVT type must be the same for this binding. The `<SrvId = xx >` is showed for more convenience.

Keep the binding configuration as the default.

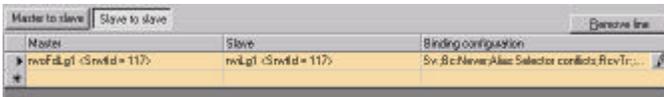
You can change the binding configuration by selecting the column Binding configuration and clicking on button .



Picture 68 dialog box Binding settings

Now click on .

Select the same network variables as for Master to slave, it means **nvoFdLg1** for the first column (Slave source) and **nviLg1** for second column (Slave destination).



Picture 69 Network variable selection about binding slave to slave

Click on  to validate the **Object's grouping rules**.

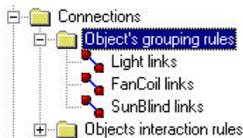


Picture 70 Light links in Object grouping rules tree

Click on  to save your project here.

Do the same for **FanCoil** and **SunBlind** profile.

All Object grouping rules are in tree.



Picture 71 Object's grouping rules tree

## Step 4 Configuring interaction rules

In step 3 we saw rules between object profiles of same type.

Now we will see binding rules between two objects profiles of different types.

This is the case for the RemoteControl and some other network objects.

RemoteControl object must be link to light object, FanCoil object and SunBlind object when belonging to same area.

This type of bindings are called **Objects interactions rules**.

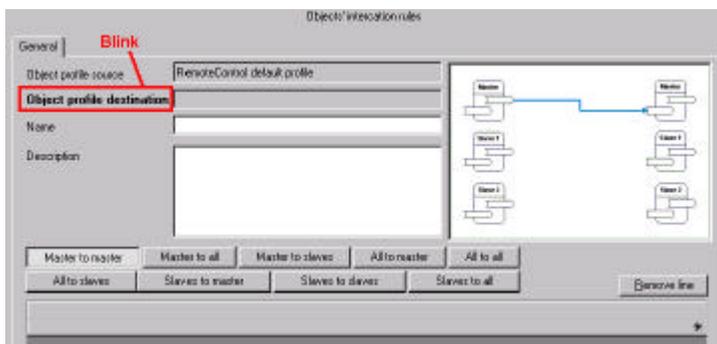
### **Object's interaction rules between RemoteControl and Light**

When RemoteControl objects and light objects are in same area then RemoteControl objects must send their **nvoLg1a** network variables to the **nviLg1** of the light object.

Right click on the **RemoteControl default profile** item and select **New Objects interaction rules ...** option.



To start edition of grouping rule show you a windows to the right.



Picture 72 Object's interaction rules window

**Object profile destination** is blinking because we do not set the other object's profile required for the interaction rules.

Drag the item **Light default profile** from the tree to the right window.

When you drop the item in the window **Object profile destination** stops to blink and **Light default profile** is now displayed.

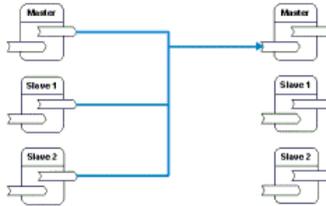
We will name theses rules as **Remote to light links**.



Now we have to define the bindings, and first the type of the binding.

We need to have all RemoteControl objects in an area to send order but only to the master of the lights object in area (because master will send lighting command using the **Grouping object's rules** we see in step 3).

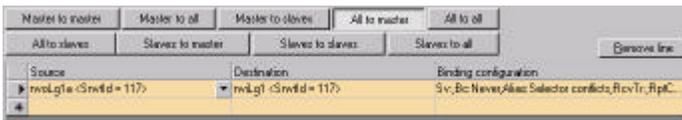
In this case we must click on **All to master** button.



Picture 73 Schema of binding All to Master

We now have to select the network variables to bound.

We will send **nvoFdLg1a** of the RemoteControl object to **nviLg1** of the light object.



Picture 74 Network variable selection about binding All to master

Click on  to validate the **Objects interactions rules**.

Click on  to save your project here.

### **Object's interaction rules between RemoteControl and all other**

You made the same operations for each other objects profiles that need interaction rules.



Picture 75 Object's interaction rules tree

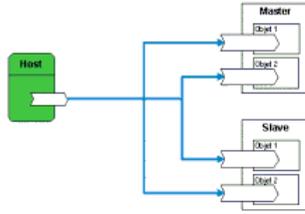
## **Step 5 Configuration of the bindings with the host**

In this step we will see how to configure bindings with the host.

The host is the local PC.

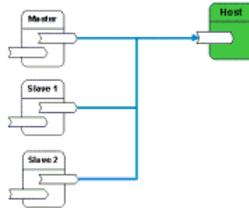
It exists two types of bindings with the host: **Host command** and **bind to host** rule.

The first binding is used to send command from host to devices.



Picture 76 Schema of Host command

The second binding is used to bound network variables of remote devices to the host.

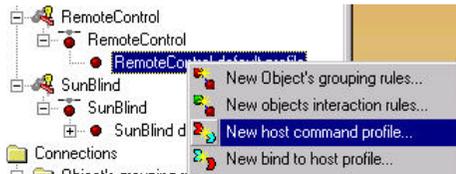


Picture 77 Schema of Bound to Host

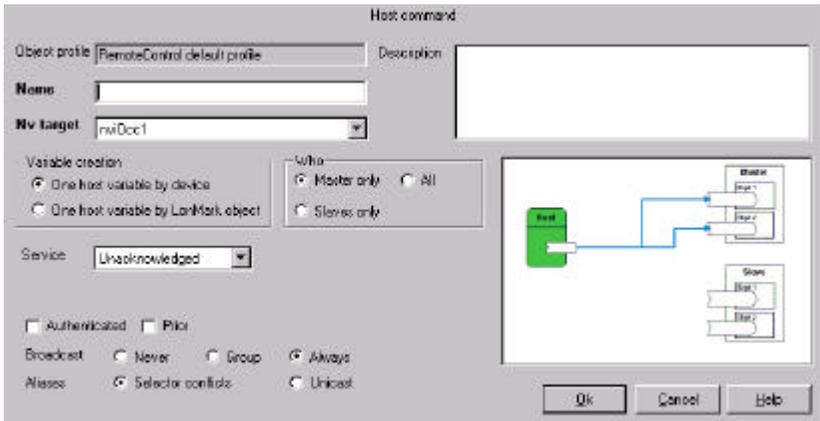
### Host commands

We will make here a host command for the input occupancy network variable (nviOcc1) of the RemoteControl devices.

Right click on **RemoteControl default profile** item in tree and select option **New host command profile ...**



To start edition of Host command profile show you a windows to the right.



Picture 78 Host command window

Set **Occupancy command** in Nam field.



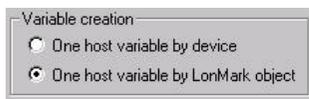
The **Nv Target** is **nviOcc1**.



The **variable creation** is used to avoid aliases problem.

It configures the number of network variable on host when destination device have several objects and cannot support aliases.

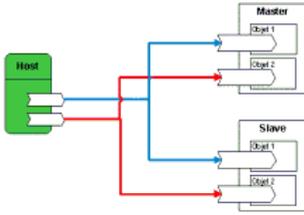
The RemoteControl device has two objects RemoteControl and to avoid aliases problem we will check the option **One host variable by LonMark object**.



In the **Who** field we will select **All** in order to have all presence devices to get the command.



The result bitmap for our command must be



Now we just have to configure the properties of the binding to have it as a broadcast command.

In **Service** select **Unack/Rpt**.



In **Broadcast** select **Always**.



Click on **OK** to validate the changes.

Click on to save your project here.

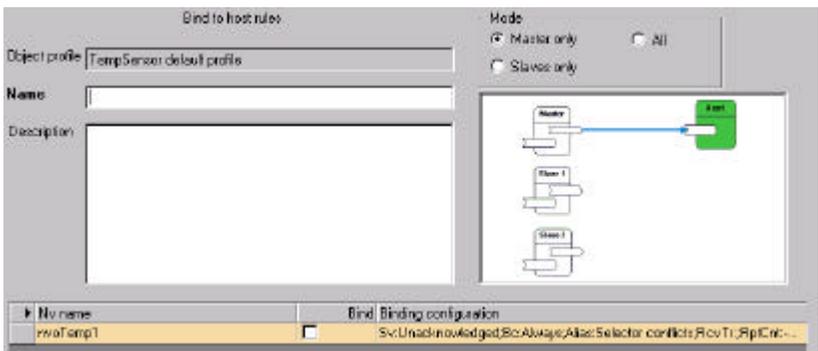
### ***Binding to host***

We will configure the network variable **nvoTemp1** of the TempSensor object to be bound to the host.

Right click on HVAC default profile item in tree and select option **New bind to host profile ...**



To start edition of grouping rule show you a windows to the right.



Picture 79 Bind to host window

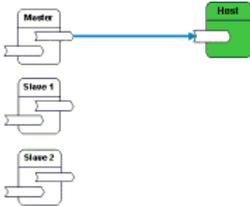
Set **Temperature to host** in **name** field.

Name

In **mode** select **Master device only** because in our example only the master HVAC object has a temperature sensor.

Mode  
 Master device only  All devices  
 Slaves devices only

The result bitmap for the binding must be



In the grid check the network variable **nvoTemp1**.

Nv name	Bind	Bind
nvoTemp1	<input checked="" type="checkbox"/>	Sv:L

Click on  to validate the changes.

Click on  to save your project here.

## Step 6 Configuration of an office

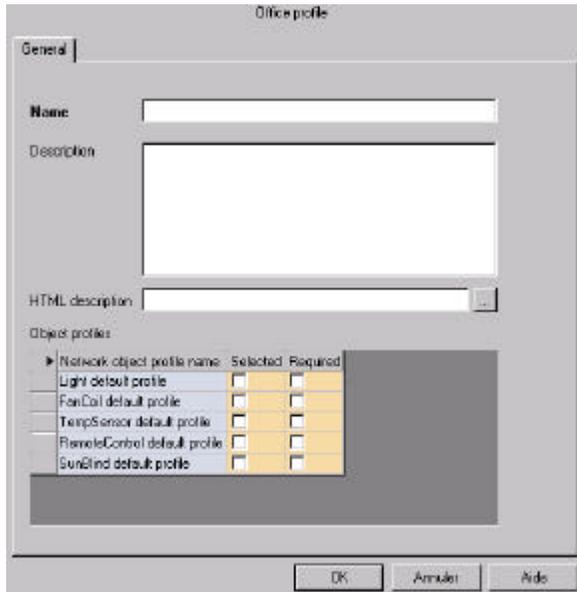
In this final step we will create an office profile.

### Create office

Right click on **Life areas** and select option **New office profile ...**



To start create office profile show you a windows to the right.



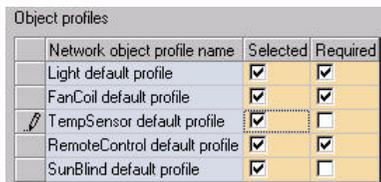
Picture 80 Office profile setting window

Set **Office#1** in name field.



In this office we want to use lights, FanCoil, SunBlind, TempSensor and RemoteControl objects. We will set too lights , FanCoil and RemoteControl as required.

The grid selection must look as this:



Click on  to office profile.

Now four new tabs are available.

Tabs are show on follow.



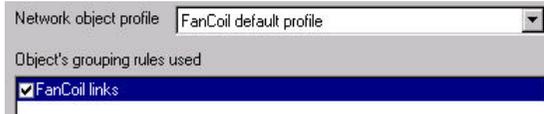
### ***Set Object's grouping rules for office***

Click on **Object's grouping rules** tab

In this screen we must select the **Object's grouping rules** used by the office.

In **Network object profile** select **FanCoil default profile**.

Then check **FanCoil Links** in the list.



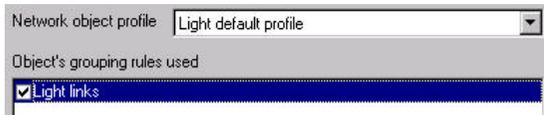
Network object profile: FanCoil default profile

Object's grouping rules used

- FanCoil links

Now in **Network object profile** select **Light default profile**.

Then check **Lights Links** in the list.



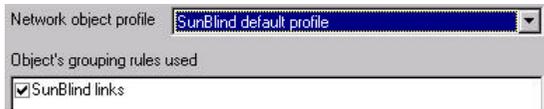
Network object profile: Light default profile

Object's grouping rules used

- Light links

Now in **Network object profile** select **SunBLind default profile**.

Then check **SunBLind Links** in the list.



Network object profile: SunBlind default profile

Object's grouping rules used

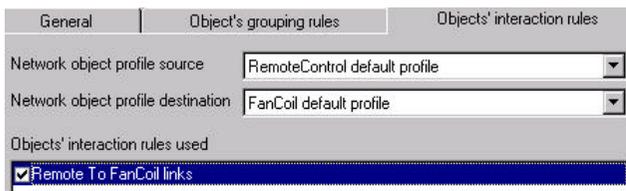
- SunBlind links

### ***Set Object's interaction rules for office***

Now click on **Object's interaction rules** tab.

In this screen we must select the **Objects interaction rules** used by the office.

In **Network object profile source** select **RemoteControl default profile**. In **Network object profile destination** select **FanCoil default profile**. In the list check **Remote to FanCoil links**.



General | Object's grouping rules | Objects' interaction rules

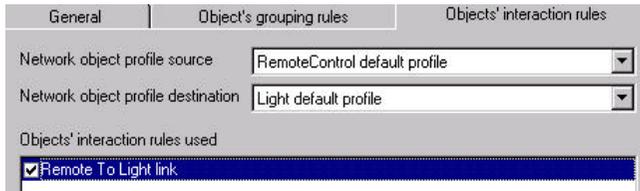
Network object profile source: RemoteControl default profile

Network object profile destination: FanCoil default profile

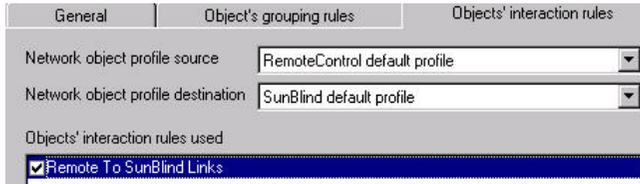
Objects' interaction rules used

- Remote To FanCoil links

Now in **Network object profile destination** select **Light default profile**. In the list check **Remote to light links**.



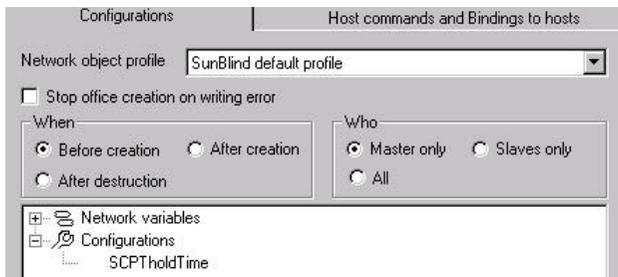
Now in **Network object profile destination** select **SunBlind default profile**. In the list check **Remote to SunBlind links**.



### **Set Configuration for office**

The tab **Configurations** is used to write network variables and configurations on object when office is created or deleted.

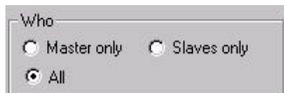
In **Network object profile destination** select **SunBlind default profile**. In the list open Configurations. You see SCPTHoldTime. This network configuration properties set the time to action is maintained (light green and red are ON in demonstration suit case).



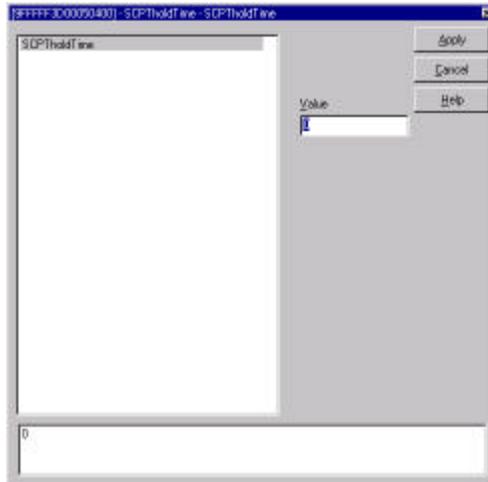
Check **Before creation** for set value before creation office in **When** options.



This set must be made for all network objects. Check **All** in **Who** options.



Double click on **SCPTHoldTime** variable. A dialog box will appear.



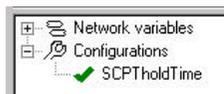
Picture 81 dialog box of setting network configuration value

Select the variable **SCPTHoldTime** in list set **0,5** in value field.



Click on  to apply configuration.

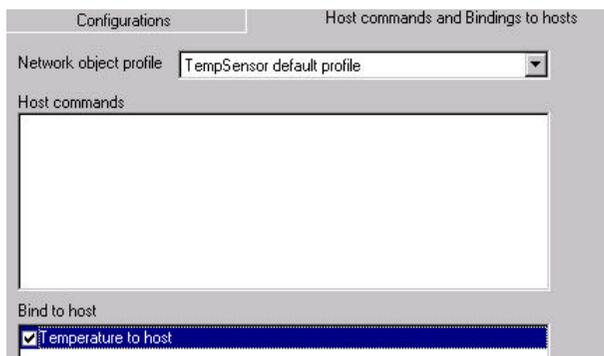
In the previous windows the **SCPTHoldTime** is checked for remember information.



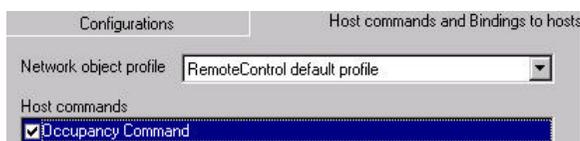
### ***Set Host command and binding to hosts for office***

Now in **Host commands and Binding to hosts** we must select the host binding to use for the office.

In Network object profile select **TempSensor default profile** then check **Temperature to host**.



In Network object profile select **RemoteControl default profile** then check **Occupancy command**.



Click on  to validate the changes.

Click on  to save your project here.