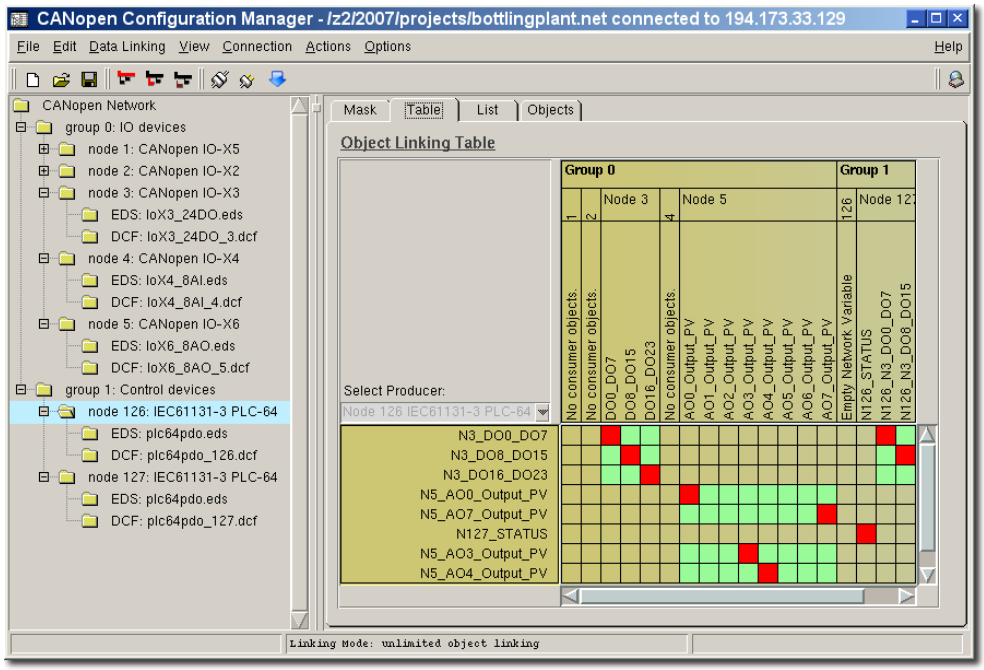
CANopen Configuration Manager

User Manual



Disclaimer
All rights reserved

The programs, boards and documentations supplied by *port* GmbH are created with due diligence, checked carefully and tested on several applications.

Nevertheless, *port* GmbH can not take over no guarantee and no assume del credere liability that the program, the hardware board and the documentation are error-free respective are suitable to serve the special purpose.

In particular performance characteristics and technical data given in this document may not be constituted to be guaranteed product features in any legal sense.

For consequential damages, which are emerged on the strength of use the program and the hardware boards therefore, every legal responsibility or liability is excluded.

port has the right to modify the products described or their documentation at any time without prior warning, as long as these changes are made for reasons of reliability or technical improvement.

We are thankful for hints of possible errors and may ask around for an information.

We will go all the way to verify such hints fastest

Copyright

Regensburger Straße 7b
D-06132 Halle
Tel. +49 345 - 777 55 0
Fax. +49 345 - 777 55 20
E-Mail service@port.de
Internet http://www.port.de

Table of Contents

1. Introduction

1.1. Product Overview

The CANopen Configuration Manager of *port* is a tool for the configuration of CANopen networks. Inputs and outputs of different devices can be linked together in a comfortable way. Based on these links the PDO configurations for each device are created automatically. The configurations are written into DCF-Files or can be send directly to the devices, if a CAN hardware interface is available.

1.2. Specialities at network access

Versions of the CANopen Configuration Manager which have the abbility of a network access perform this access via a so called CANopen-Server. This CANopen-Server may run on the local computer, on a remote computer or in an embedded device. The CANopen Configuration Manager uses TCP/IP to communicate with this server (m4d) using the protocol CiA 309-3 standardized by CAN in Automation.

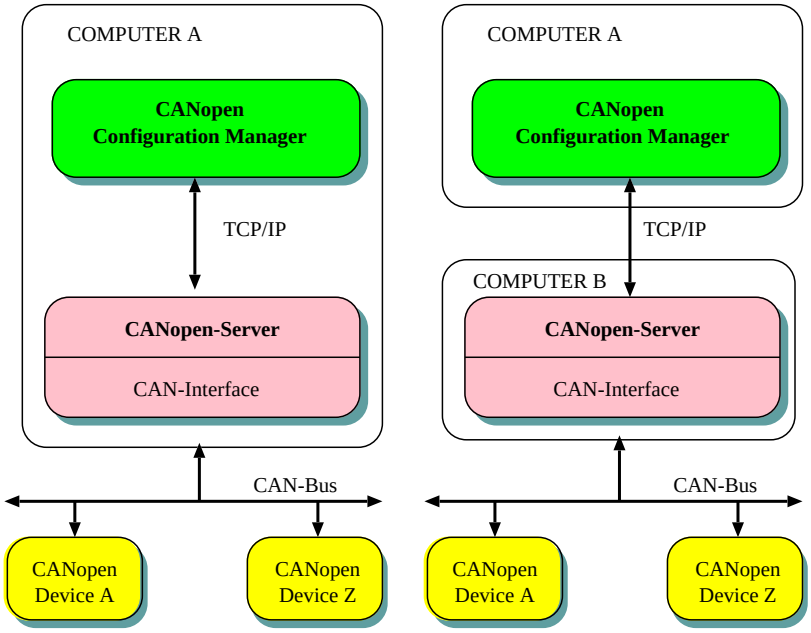


Figure 1: Structure of the CANopen Configuration Manager

1.3. Product Delivery

The scope of delivery of the CANopen Configuration Manager basic:

- User Manual

- software for CANopen Configuration Manager

The CANopen Configuration Manager contains additionally:

- CANopen-Server (m4d)
- a CAN hardware interface

All software components are available for download.

1.4. System Requirements

The CANopen Configuration Manager runs on PC's with Microsoft Windows 2000, Windows XP, Windows Vista or Linux.

Operating System:	Windows 2000, Windows XP, Vista, Linux
Processor:	Pentium IV or later
RAM:	256 MByte
Hard-disk Space:	15 MByte

1.5. Support by *port*

The senior engineers at port support the user by a telephone hot-line and by training courses. Additionally the user can have consultations in the whole field of CAN e.g. network planning, network configuration, message distribution, selection of devices and CANopen Profile implementations.

Please ask at:

email:	**service@port.de**
phone:	**"+49 345 777 55 - 0"**
Fax:	**"+49 345 777 55 - 20"**

2. Installation

The installation includes:

- the CANopen Configuration Manager
- the m4d server and
- a layer 2 driver for the CAN interface.

For the installation the following steps are necessary:

1. Maybe preparing installation steps are necessary depending on the used CAN-Interface.
2. Execute setup.exe.

   • Full Installation: The installation of all software components is happened automatically and menu driven. This includes copying of all manuals.

   • Customized Installation: The selection of software components is possible, which should be installed. For the installation of the CANopen Configuration Manager the following components are necessary: CANopen Configuration Manager, m4d and layer 2 driver.
3. For the icon on the desktop set the options for the call of the m4d server depending on your application. An overview of the options are given by the help: m4d -h

   Alternatively the CANopen Configuration Manager is able to start the CANopen server (m4d). For additional hints please see section "Hardware Configuration"
4. Define the working directory for the shortcuts on your desktop depending on your application.

3. Hardware Configuration

As alternative to the previous described method the CANopen Configuration Manager is also capable to start the CANopen-Server by itself. Thereby the CANopen Configuration Manager handles the start, the connection establishment, and the closing of the CANopen-Server (m4d) automatically.

At the first start of the CANopen Configuration Manager after the installation the configuration dialog opens automatically. After that the hardware configuration can be opened by Connection → CAN-Interface Configuration

The other method with a separate start of the CANopen-Server is still possible, so that its advantages like a remote-control of device via a TCP/IP network can be used, too.

3.1. Configuration Dialog

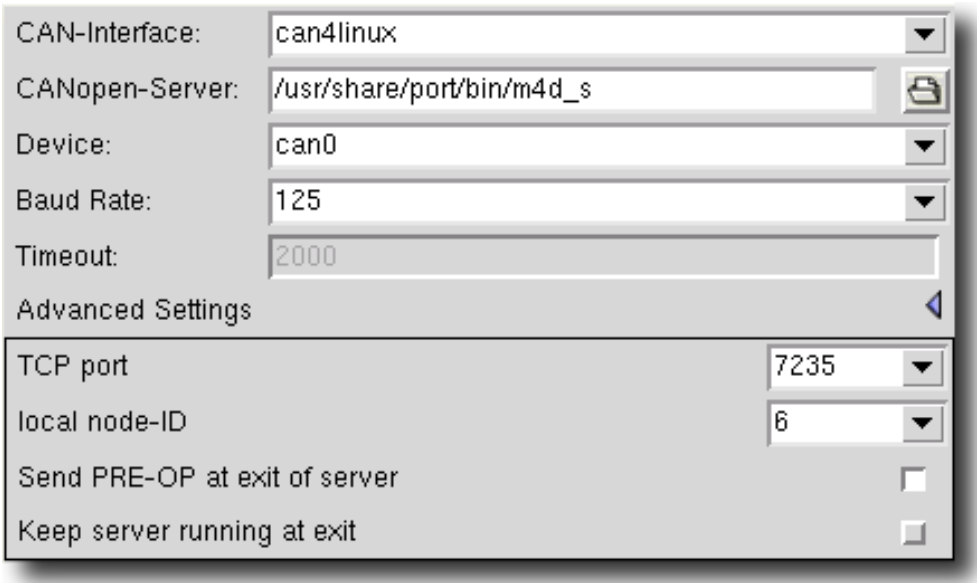


Figure 2: Hardware configuration dialog

Is TCP selected as "CAN-Interface", so the CANopen-Server has to be started separately and all other options in the dialog are disabled.

The following options are available for all supported hardware interfaces:

Option	Description
CANopen-Server	path to the CANopen-Server (m4d)
Baud Rate	CAN Baudrate [kbit/s]
TCP port	TCP-port used for the communication between the CCM and the CANopen-Server (m4d)
local node-ID	Node-ID of the CANopen-Server (m4d)
Send PREOP at exit	automatic transmission of the NMT command ENTER PREOPERATIONAL to all devices when the CANopen-Server is shut down
Keep server running at exit	the CANopen-Server will not be closed when the CCM is closed

Depending on the hardware interface some additional options like device, channel, board or unit can be present. These options distinguish the connected device or the desired communication channel.

The CANopen Configuration Manager searches for all installed drivers and CANopen-Server and offers only the installed drivers at the option CAN-Interface. Mostly it's only one driver plus TCP as default option.

Network access is not available in the version CCMbasic.

4. Overview

4.1. Main menu

4.1.1. File menu

Via the file menu the following functions are available:

New Project	Create a new project
Load EDS	Load a EDS file
Open Project	Open a project
Recently used projects	List of recently used projects
Import Project	Import an existing project
Save Project	Save the project
Save Project as	Save the project using a different name Hint: It is highly recommended to save the projects into different folders. The project outputs are generated automatically prior to saving the project.
Quit	Closes the CANopen Configuration Manager

4.1.2. Edit menu

Via the edit menu the following functions are available:

Delete Node	Delete the selected node

4.1.3. Data Linking menu

Via the data linking menu the following functions are available:

Generate PDO Linking	Generate the pdo configuration
Write DCF Files & Documentation	Write the configuration into DCF files and generates the documenation

4.1.4. View menu

EDS and DCF files can be viewed via the view command.

EDS file	Show the EDS file of the current node
DCF file	Show the DCF file of the current node
IEC61131 file	Show the IEC61131 variable file
Project documentation	Show the project documentation
Log file	Show the log file
CAN message log	Open the CAN message display

4.1.5. Connection menu

Via this menu connections to a CANopen-Server can be opened, configured and closed. It is only available if the currently used release of the CANopen Configuration Manager supports it.

Connect	Connect to a CANopen server
Disconnect	Disconnect
CAN-Interface	Configuration of the CANopen server

4.1.6. Action menu

The action menu allows access to the CANopen network. It is only available if the currently used release of the CANopen Configuration Manager supports it.

Stop Network	Send the NMT command STOP
Preop Network	Send the NMT command ENTER PREOPERATIONAL
Reset Communication	Send the NMT command Reset Communication
Reset Node	Send the NMT command Reset Communication
Start Network	Send the NMT command START
Send Configuration	Transmit the current configuration to the nodes Subsequent to the transmission the parameters can be stored in non-volatile memory if it is supported by the devices.

 Handle network access with caution.

4.1.7. Options menu

Different options can be configured using this menu. The number of the options and their availableness depend on the kind of the CANopen Configuration Manager and on the current linking mode.

Menu item	Description
EDS Repository	Configuration of the EDS repository
Update EDS Repository	Update repository information
Ask before network access	Toggle this option
COB-ID-Distribution	Configuration of the COB-ID-Distribution settings
Auto-Connect	Enable or disable the auto-connect feature
Verbose status information	Enable or disable verbose status information
Copy EDS files to project	If this is activated, the EDS files are copied from the EDS repository to the project folder.
Reset default values	If this is activated, the default settings of the device are restored by writing to the object 0x1011 before the device is configured. Additionally, only value that differ from their default values are written into the DCF or concise DCF file.
Keep active tab	Keeps the active tab when the node id changed.
Ignore errors at download	With this option, errors at the download of the configuration to the devices can be ignored (SDO abort or missing devices).
Font Settings	Configure the font settings

4.1.8. Help menu

Context Help	Show specific help in the standard browser
About	Show information about the program
Install Licence File	The way to install a (new) licence file
Latest release info	Retrieve information about new releases

4.2. Tool bar

The tool bar provides access to the following functions (from left to right):

- File -> New Project
- File -> Save Project
- File -> Open Project
- Add new group
- Add new node
- Delete active node

- Connect
- Disconnect

5. Network tree

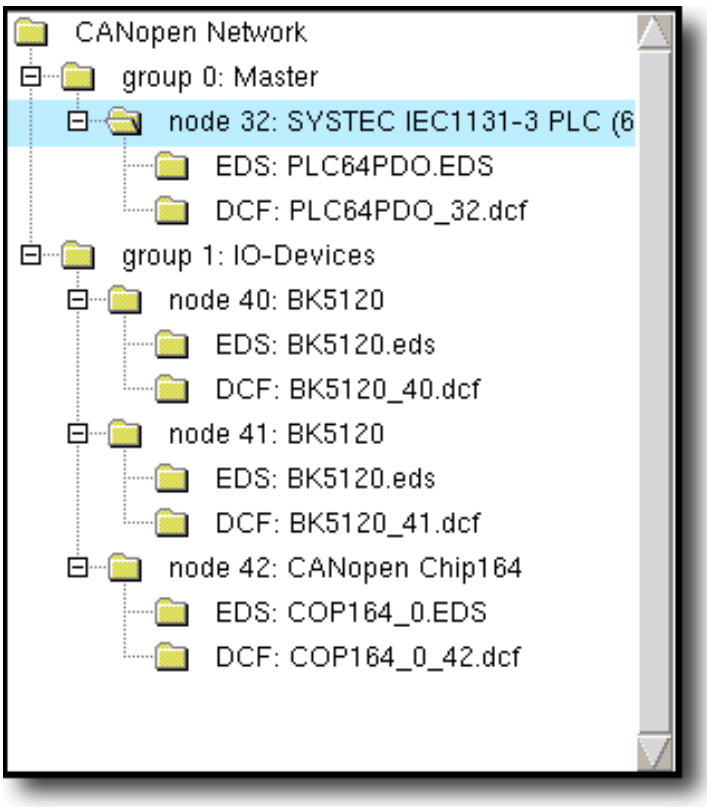


Figure 3, Network tree

The network tree contains all nodes of a network in groups. A group is a set of nodes with a common attribute. E.g. it is possible to arrange all input devices and all outputs in different groups or to arrange them according to their topology.

Popup menus, that are available via the right mouse button, are assigned to different tree elements of the tree.

These elements are

- CANopen Network
  - Groups
  - Nodes

The elements of these popup menus are:

CANopen Network	Add new Group	Adds a new group

group	Rename Group	Renames the group
	Delete Group	Deletes the group
	Add new node	Adds a new node

node	Change Node-ID	Changes the node ID
	Rename node	Renames the node
	Move node	Moves node to another group
	Delete node	Deletes the node
	Load EDS file	Loads an EDS file
	Change DCF name	Renames the DCF file of the node

! It is not possible to undo changes here.

6. EDS Settings

6.1. EDS Files

EDS files describe CANopen devices, their properties and objects in a standardized text format.

The CANopen Configuration Manager is able to read EDS files newer than EDS version 4.0. The main difference to earlier version is the declaration of the PDOs of a device. If you get the error message "No PDO found", your EDS is maybe out of date.

6.2. EDS Repository

The EDS Repository is a folder that stores all EDS files. Via Options → EDS Repository the folder for the repository can be chosen.

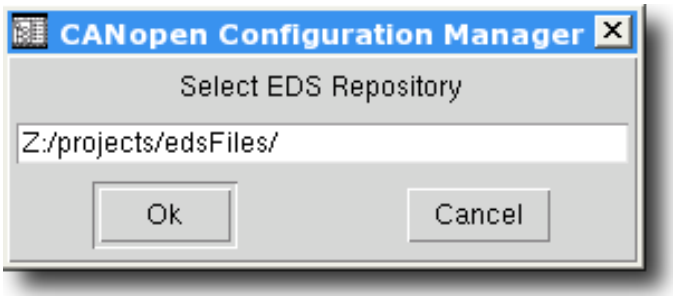
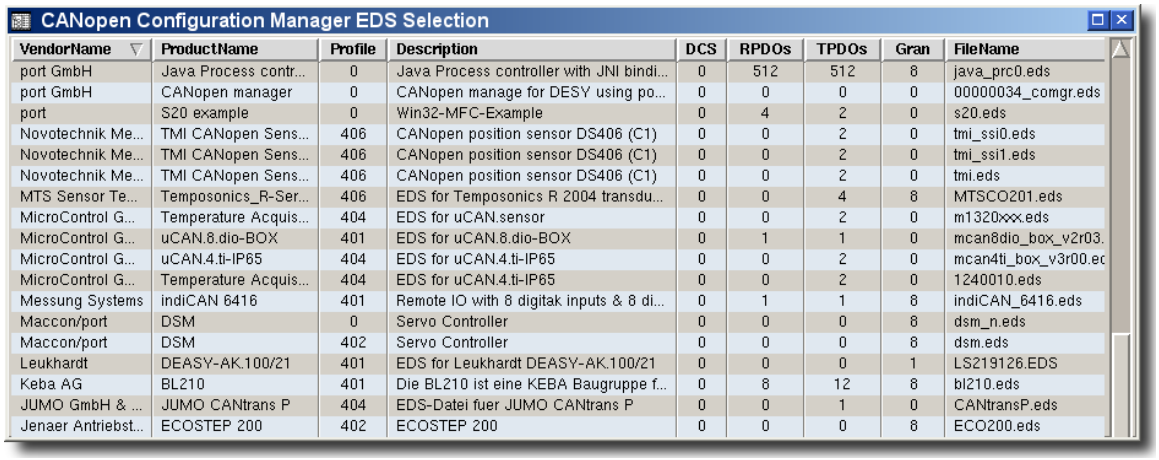


Figure 4, dialog to select the EDS directory

To add files to the EDS repository they have to be copied into the EDS repository folder. Additionally, the repository data have to be updated by Options → Update EDS Repository

The EDS selection dialog simplifies the selection of an EDS file for the device by showing device specific parameter.



VendorName	ProductName	Profile	Description	DCS	RPDOs	TPDOs	Gran	FileName
port GmbH	Java Process contr...	0	Java Process controller with JNI bindi...	0	512	512	8	java_prc0.eds
port GmbH	CANopen manager	0	CANopen manage for DESY using po...	0	0	0	0	00000034_comgr.eds
port	S20 example	0	Win32-MFC-Example	0	4	2	0	s20.eds
Novotechnik Me...	TMI CANopen Sens...	406	CANopen position sensor DS406 (C1)	0	0	2	0	tmi_ssi0.eds
Novotechnik Me...	TMI CANopen Sens...	406	CANopen position sensor DS406 (C1)	0	0	2	0	tmi_ssi1.eds
Novotechnik Me...	TMI CANopen Sens...	406	CANopen position sensor DS406 (C1)	0	0	2	0	tmi.eds
MTS Sensor Te...	Temposonics_R-Ser...	406	EDS for Temposonics R 2004 transdu...	0	0	4	8	MTSCO201.eds
MicroControl G...	Temperature Acquis...	404	EDS for uCAN.sensor	0	0	2	0	m1320xxx.eds
MicroControl G...	uCAN.8.dio-BOX	401	EDS for uCAN.8.dio-BOX	0	1	1	0	mcan8dio_box_v2r03.
MicroControl G...	uCAN.4.ti-IP65	404	EDS for uCAN.4.ti-IP65	0	0	2	0	mcan4ti_box_v3r00.ec
MicroControl G...	Temperature Acquis...	404	EDS for uCAN.4.ti-IP65	0	0	2	0	1240010.eds
Messung Systems	indiCAN 6416	401	Remote IO with 8 digitak inputs & 8 di...	0	1	1	8	indiCAN_6416.eds
Maccon/port	DSM	0	Servo Controller	0	0	0	8	dsm_n.eds
Maccon/port	DSM	402	Servo Controller	0	0	0	8	dsm.eds
Leukhardt	DEASY-AK.100/21	401	EDS for Leukhardt DEASY-AK.100/21	0	0	0	1	LS219126.EDS
Keba AG	BL210	401	Die BL210 ist eine KEBA Baugruppe f...	0	8	12	8	bl210.eds
JUMO GmbH & ...	JUMO CANtrans P	404	EDS-Datei fuer JUMO CANtrans P	0	0	1	0	CANtransP.eds
Jenaer Antriebst...	ECOSTEP 200	402	ECOSTEP 200	0	0	0	8	ECO200.eds

Figure 5, EDS selection dialog

The following table explains the columns of the list:

Column header	Description
VendorName	Name of the device's vendor, read from DeviceInfo section
ProductName	Name of the device, read from DeviceInfo section
Profile	Used device profile, lower word of the value of object 0x1000
Description	Description of the device, read from DeviceInfo section
DCS	Value of the "DynamicChannelsSupported"-Entry
RPDOs	Number of Receive-PDOs, value of "NrOfRXPDO"-Entry
TPDOs	Number of Transmit-PDOs, value of "NrOfRXPDO"-Entry
Gran	Value of the "Granularity"-Entry

7. Configuration tabs

According to the selected element in the network special tabs are available at the right side. When dealing with large projects on slow computers the activation and updating of these tabs can take a noticeable time.
Depending on the active tree element the mask tabs provide access to node or network settings or present only general information.

7.1. Mask tab

If the CANopen network is selected, the mask tab provides access to network settings like baud rate, reset timeout, global SYNC parameters or the desired linking mode.

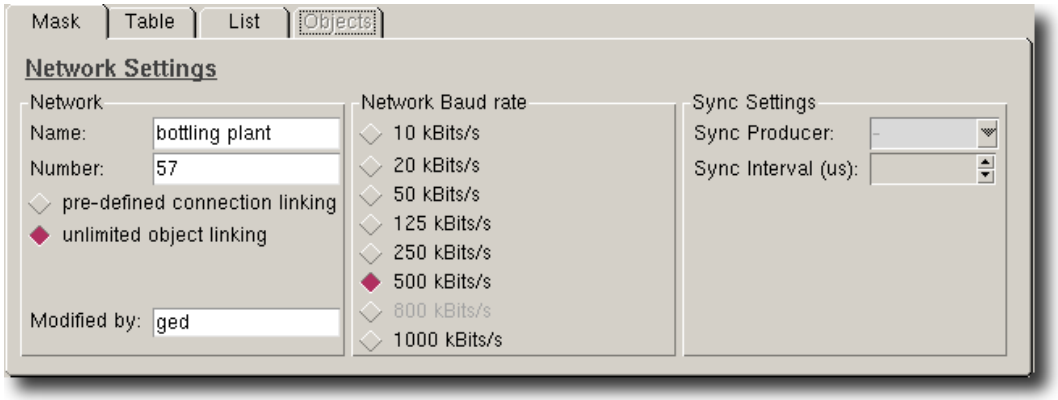


Figure 6, Mask tab (network)

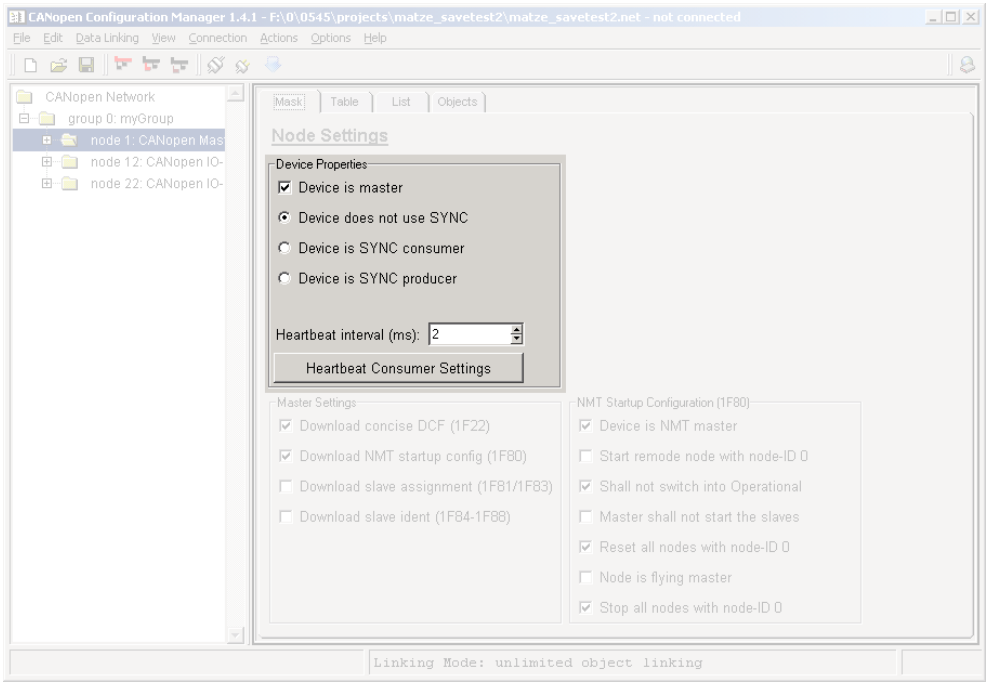


Figure 7, Mask tab (node settings)

The mask for nodes contains the following items:

Section **Device Properties**	
Device is master	Set the master flag
Device does not use SYNC/ Device is SYNC Consumer/ Device is SYNC Producer	The SYNC behavior of the device is configured according to this selection.
Heartbeat interval	Configure the heartbeat producer interval in ms
SYNC Period	Configure the SYNC period for SYNC Consumers, if supported by this device
Synchronous Windows	Configure the synchronous windows object
Heartbeat Consumer Settings	Configuration of Heartbeat Consumer settings if object 1016h exists.

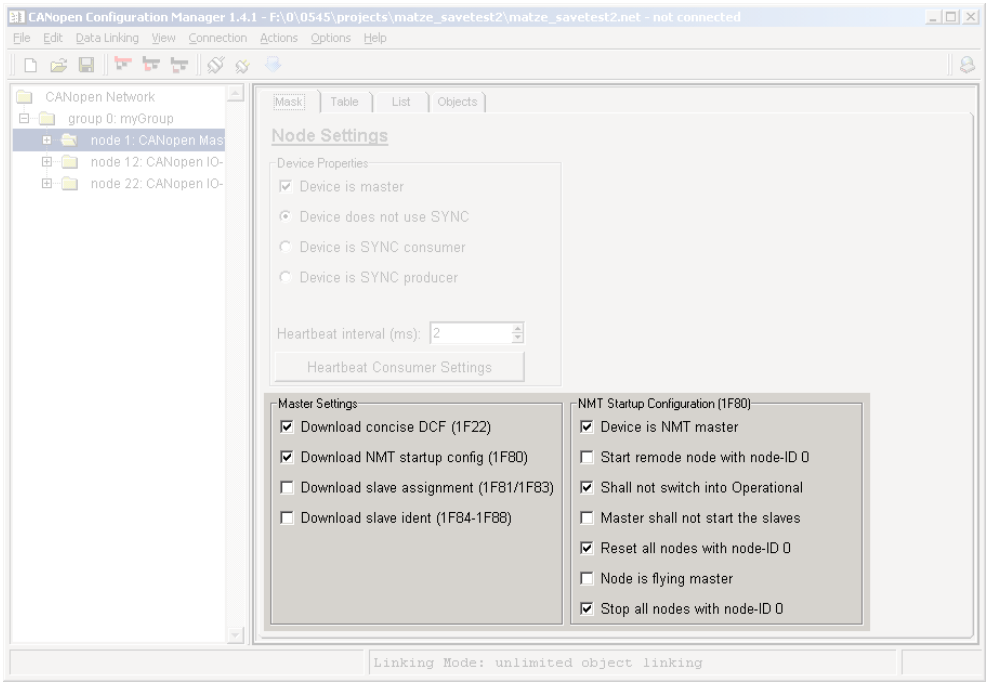


Figure 8, Mask tab (node settings)

The mask to set the configuration for this nodes contains the following items:

Section **Master Settings**	
Download concise DCF (1F22)	Download of the particular DCFs (from slaves) onto the master.
Download NMT startup config (1F80)	Download of the particular slave startup configuration (from slaves) onto the master.
Download slave assignment (1F81)	Download of particular slave assignments onto the master.
Download slave ident (1F85-1F88)	Download of slave identification information (object 1018h) onto the master.
Boot Time (1F89) in ms	Configuration of the boot time of the master in milliseconds.

Section **NMT Startup Configuration (1F80)**	
Device is NMT master	If marked so the device is the NMT master.
Start remode node with node-ID 0	If marked so the NMT will start the remote node with ID 0.
Shall not switch into Operational	If marked so the device does not switch into **operational** mode by itself.
Master shall not start the slaves	If marked, the master shall not start the NMT slaves (this will be done by the application).
Reset all nodes with node-ID 0	If marked reset all nodes if an error control event occurs (send node-ID 0).
Node is flying master	If marked the device will participate the NMT flying master negotiation.
Stop all nodes with node-ID 0	If set stop all nodes if an error control event occurs by sending node-ID 0

Section **Slave Assignment Configuration (1F81)**	
Node is NMT slave and available	If marked so the device is a NMT slave and available in the network.
Node shall be bootup by NMT Master	If marked so the NMT service 'start remote node' will be performed if an error control event or NMT service
Node is mandatory in network	If marked so the device is present prior to network startup.
Node shall not be reset in Operational	If marked, the NMT service 'reset communication' shall **not** be executed if the device is in state operational.
Check software version	If set so a software verification shall be performed for this device.
Do software update	If marked a software update shall be performed for the device.
Restore factory default before use	If set so the device shall be reset to factory defaults
Guarding Time (ms)	This value indicates the cycle time for node guarding of this device. Value 0 disables node guarding.
Retry Factor	A value of 0 disables the node guarding for that device. Other values define the number of retries the NMT master issues node guarding events.

At groups only the group members are listed. And if an EDS file is selected an overview about all main EDS parameters is shown.

7.2. Table tab

The table tab provides access to the main part of the CANopen Configuration Manager, the link table.
The output objects of the node, selected at 'Select Producer', are listed at the left legend of the table. The names of the objects are taken from the parameter name from the EDS file. If the output objects are network variables according to DSP302, their names are built from the node ID and the EDS-Name of the corresponding consumer object. For these network variables a popup menu is available that allows deleting and renaming of them.

The top bar shows the input objects of the other nodes arranged in groups and nodes. To keep track of these objects nodes and groups can be folded together by a popup menu that is available via the right mouse button. Additionally the objects can be filtered by data types.

In the table body all objects can be linked with each other. If a link between two objects is possible, the intersection is displayed lightgreen. If the background of the intersection is yellowgrey, it is not possibly to link the objects because of different data types. For connections between nodes that are folded or are generally not possible also a yellowgray background is used. Pressing the left mouse button in a white field activates the connection and the field is marked red.

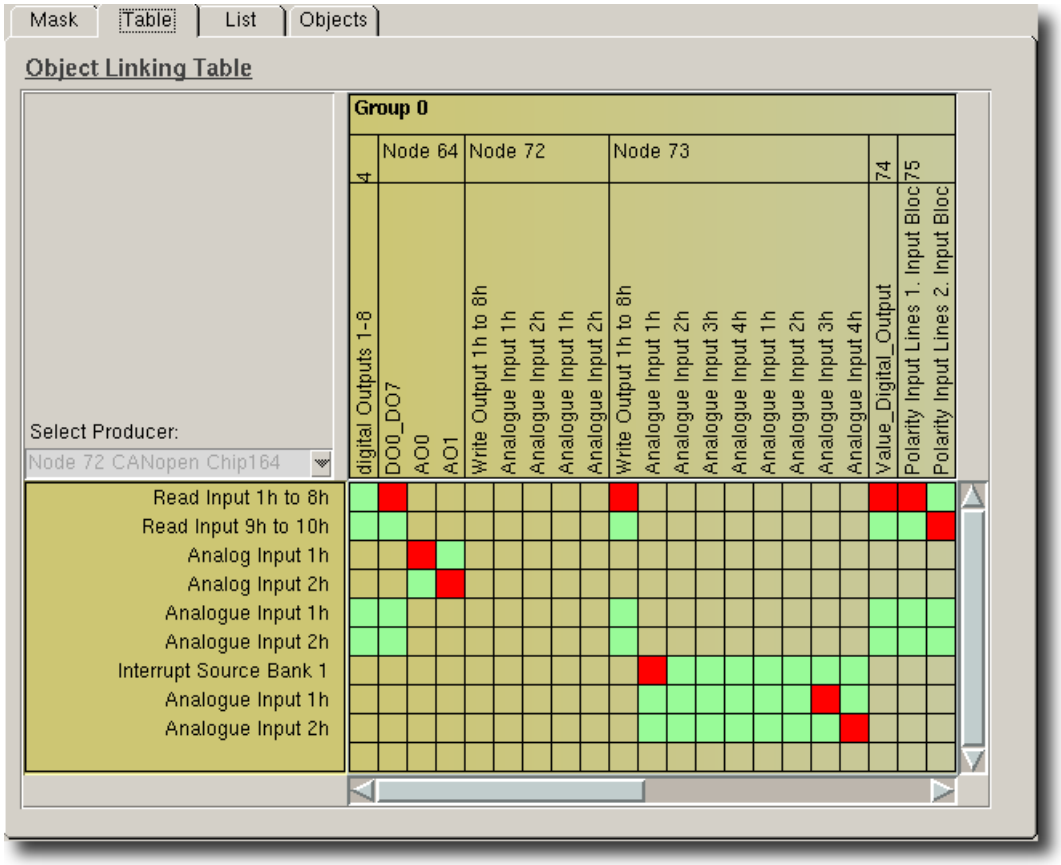


Figure 9, Table tab

Thereby the following restrictions have to be considered:

- previous connections to an input (consumer) object are deleted, when selecting a new one
- an output (producer) object can be linked with several input objects in different devices
- an input object can only be linked with one producer

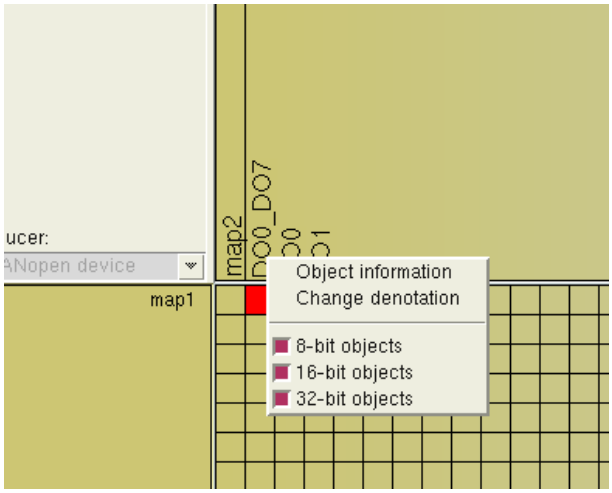


Figure 10, Context menu within the linking table

7.2.1. Context menu in linking table

As figure 10 shows it is possible to filter out different objects. Here you can select filtering of 8-bit, 16-bit and 32-bit objects. The filter menu can be activated by clicking into the empty field in the upper left corner of the Object Linking Table.

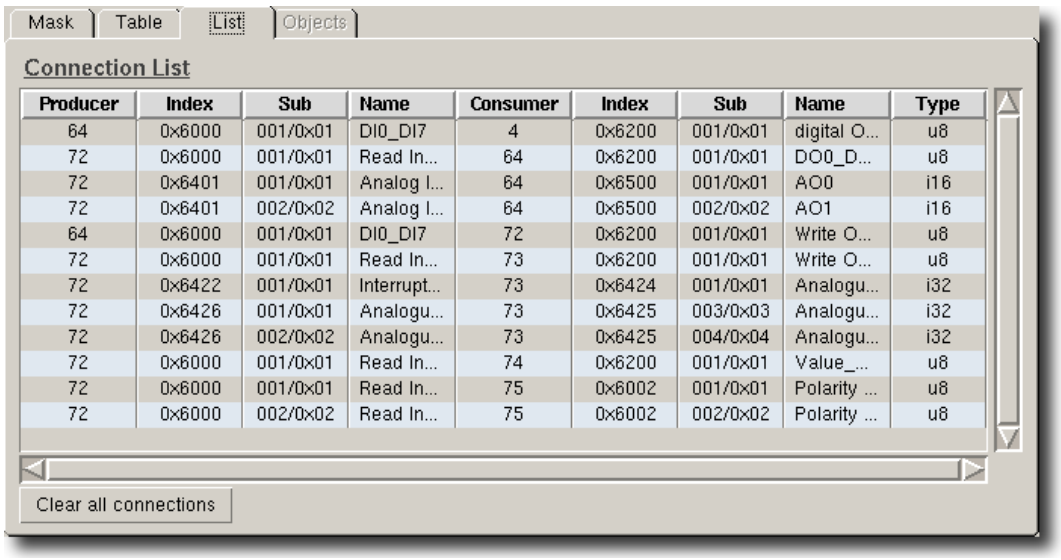
7.2.2. Horizontal scrolling

For large project with several hundred process data per device the following special funtions of the horizontal scrollbar are useful:

Klicksequenz	
Click into free space between slider and arrow	Linking table is moved by one screen
<Ctrl>+Click into free space between slider and arrow	Linking table is moved by 4 screens

7.3. List tab

Connection list



Producer	Index	Sub	Name	Consumer	Index	Sub	Name	Type
64	0x6000	001/0x01	DI0_DI7	4	0x6200	001/0x01	digital O...	u8
72	0x6000	001/0x01	Read In...	64	0x6200	001/0x01	DO0_D...	u8
72	0x6401	001/0x01	Analog I...	64	0x6500	001/0x01	AO0	i16
72	0x6401	002/0x02	Analog I...	64	0x6500	002/0x02	AO1	i16
64	0x6000	001/0x01	DI0_DI7	72	0x6200	001/0x01	Write O...	u8
72	0x6000	001/0x01	Read In...	73	0x6200	001/0x01	Write O...	u8
72	0x6422	001/0x01	Interrupt...	73	0x6424	001/0x01	Analogu...	i32
72	0x6426	001/0x01	Analogu...	73	0x6425	003/0x03	Analogu...	i32
72	0x6426	002/0x02	Analogu...	73	0x6425	004/0x04	Analogu...	i32
72	0x6000	001/0x01	Read In...	74	0x6200	001/0x01	Value_...	u8
72	0x6000	001/0x01	Read In...	75	0x6002	001/0x01	Polarity ...	u8
72	0x6000	002/0x02	Read In...	75	0x6002	002/0x02	Polarity ...	u8

Figure 11, List tab (network)

The list tab for the CANopen network shows all connections within the network. By pressing the button "Clear all connection" all connections can be deleted.

PDO configuration list

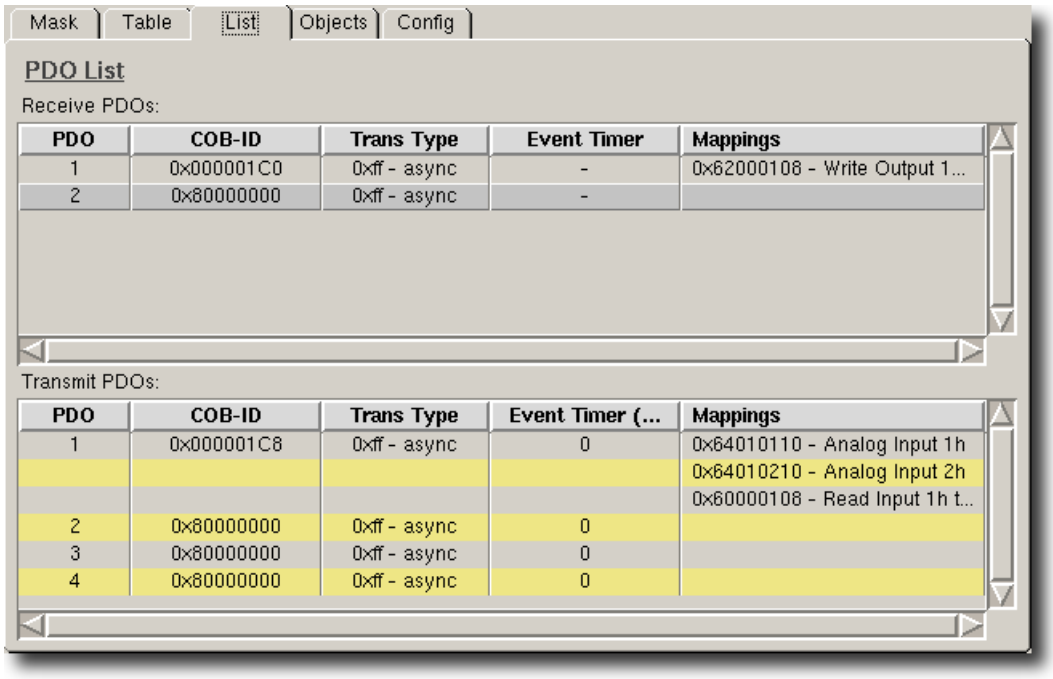


Figure 11, List tab (node)
The list tab for nodes shows the PDOs of a device with their COB-ID, transmission type, event timer, inhibit time and their mapping entries. Before a valid configuration has been created by activating "Data Linking -> Generate PDO Linking", default values read from the EDS files are shown. If a valid configuration has been created, this list shows the configured PDOs as they are written into the DCF files. In this view the transmission type and the event timer of the PDOs can be configured.

7.4. Object configuration tab

The object configuration tab provides access to all manufacturer and device profile specific objects. The ParameterValues of these objects can be configured and the configured values can be transferred to the devices when writing the configuration via CAN.

The modified ParameterValues are saved in the project file and can also be written to the DCF files.

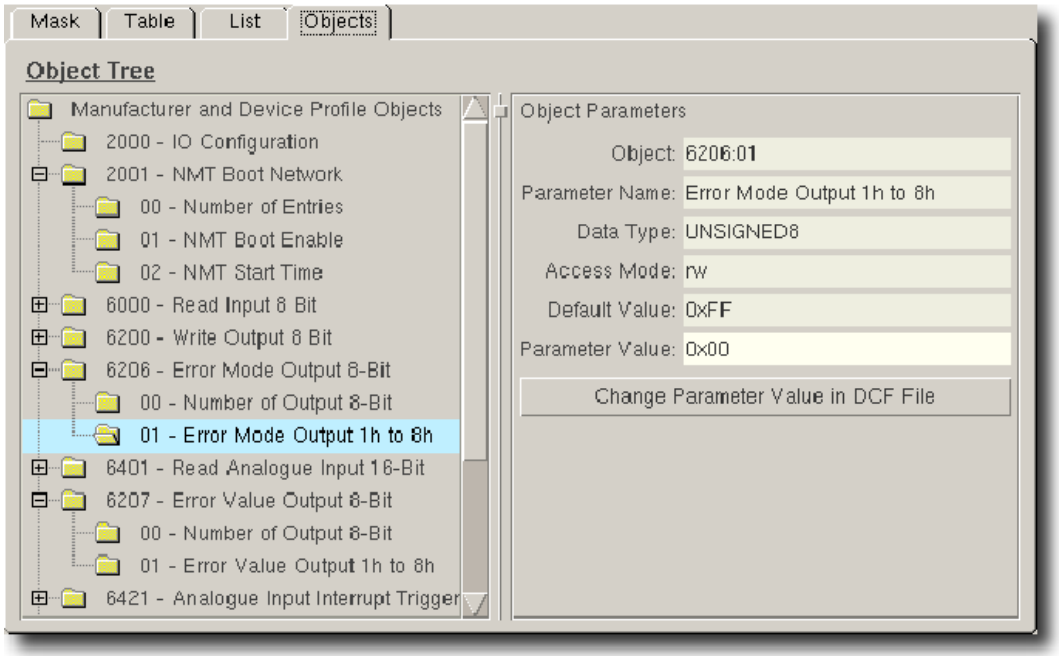


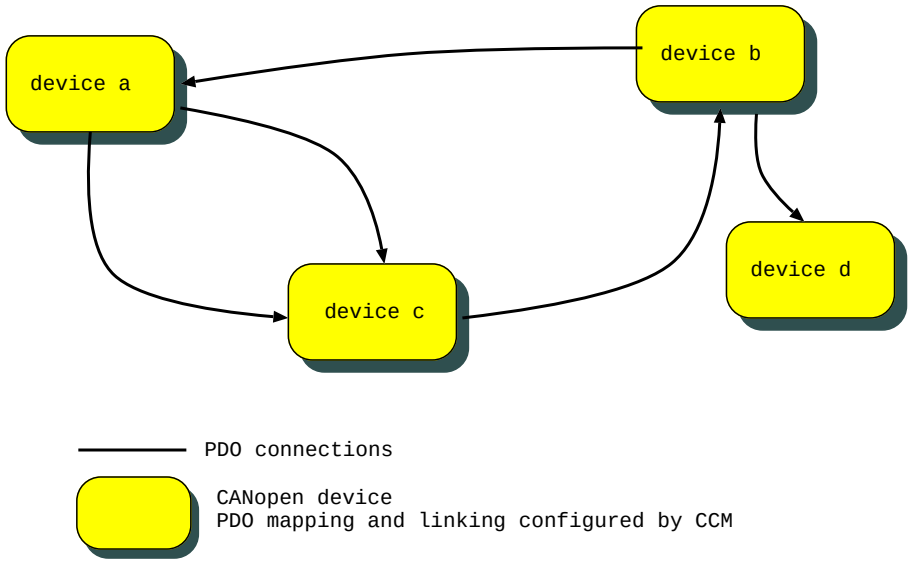
Figure 11, object configuration tree

8. PDO linking methods

8.1. unlimited object linking

The unlimited object linking allows to link objects of different devices. The COB-IDs of the PDOs and the PDO mapping entries of the devices are modified to realize the selected connections. The default mapping entries of devices which support dynamic mapping are overwritten by the CANopen Configuration Manager. When the configuration is computed and written into the DCF files, the devices have to be configured with the DCF files.

If the CANopen Configuration Manager is connected to a CAN network, the configuration can be transmitted to the devices directly.



PDO connections

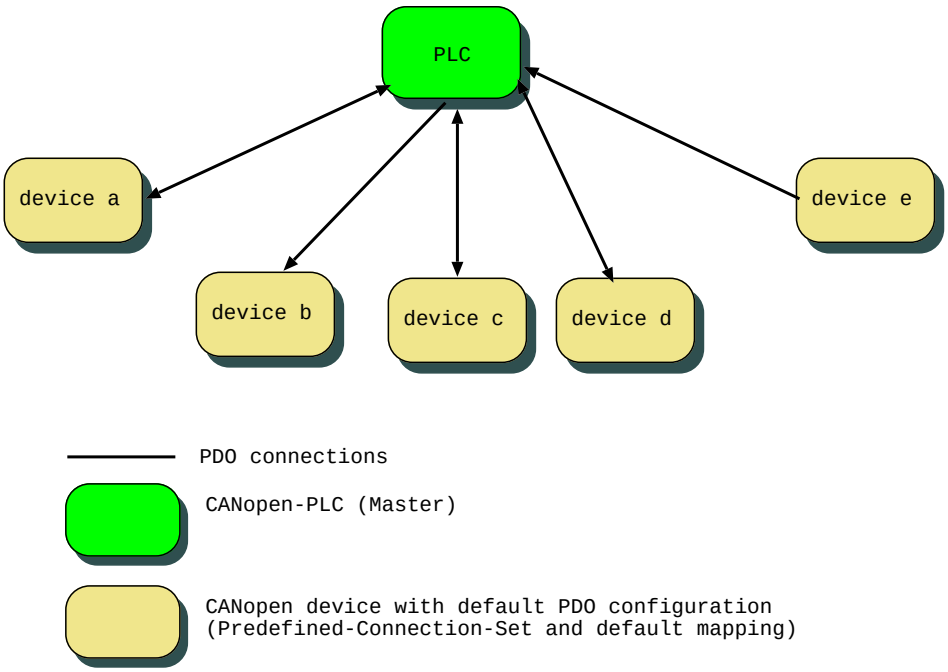
The unlimited object linking is not supported by all versions of the CANopen Configuration Manager.

8.2. Pre-defined connection linking

The pre-defined connection linking limits the flexibility of CANopen so that only connections between one master and several slaves are possible but no connections between two slaves.

At the slaves the access is limited to these objects, that are transmitted/received via the predefined mapping entries (default mapping). The information about the predefined

mapping entries is taken from the EDS file. A further restriction is that only the first 4 PDOs of the Predefined-Connection-Set can be used. For these PDOs the standard-COB-ID based on the PDO number and the node ID is used.



PDO Connections

For the user these restrictions are only visible in the connection table, where the number of input and output objects is highly limited. Furthermore, connections between slaves are not possible.

The advantage of this procedure is, that the PDO configuration of the slaves need not to be configured, because their default PDO configuration described in the EDS file is used.

The choice of a linking method has to be done as the very first step. It can be configured at the mask tab of the network.

9. Making of an example project

9.1. Example for pre-defined connection Linking

9.1.1. Example network

The example net consists of a master with several IOs and 3 further IO-devices. All connections shall be realized via the master.

The node IDs are distributed as follows:

Node-ID	Device	EDS file
30	Master	`edsfiles/master_example.eds`
40	CO411A0	`edsfiles/CO4011A0.eds`
41	CANopen Chip164	`edsfiles/COP164_0.EDS`
42	BK5120	`edsfiles/BK1520.eds`

The following connections shall be configured:

Producer	Consumer
Node 30: Input Byte 1[2]	Node 40: Write OutputByte_1
Node 30: Input Byte 2	Node 40: Write OutputByte_2
Node 30: Input Byte 3	Node 41: Write Output 1h to 8h
Node 30: Input Byte 4	Node 42: DigOutput8_1
Node 30: Input Byte 5	Node 42: DigOutput8_2
Node 30: Integer16 Input 1	Node 42: Write Analogue Output 1
Node 30: Integer16 Input 2	Node 42: Write Analogue Output 2
Node 30: Integer16 Input 3	Node 42: Write Analogue Output 3
Node 30: Integer16 Input 4	Node 42: Write Analogue Output 4
Node 40: Read Input Byte_0	Node 30: Output Byte 1
Node 40: Read Input Byte_1	Node 30: Output Byte 2
Node 41: Read Input 1h to 8h	Node 30: Output Byte 3
Node 41: Read Input 9h to 10h	Node 30: Output Byte 4
Node 41: Analog Input 1h	Node 30: Integer16 Output 5

and more

At the end of the project configuration DCF files shall be available to configure the particular devices.

[1] The names of the objects are the parameter names from the EDS file.

9.1.2. Step-by-step guide

- run CANopen Configuration Manager
- select pre-defined connection linking at CANopen Network Settings
- right mouse click on "group 0", select "Rename group"

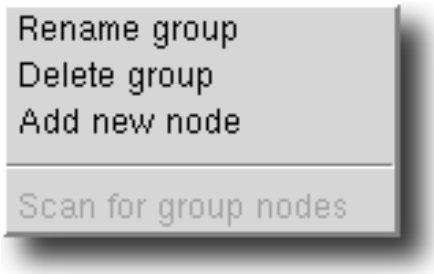


Figure 14, Popup menu of a group

- rename group 0 to "Master"

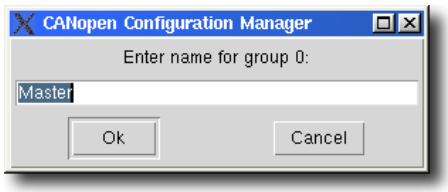


Figure 15, Rename dialog of a group

- right mouse click on "CANopen Network", select "Add new group"
- rename group 1 to "Slaves"
- right mouse click on "node 1", select "Change Node-ID"

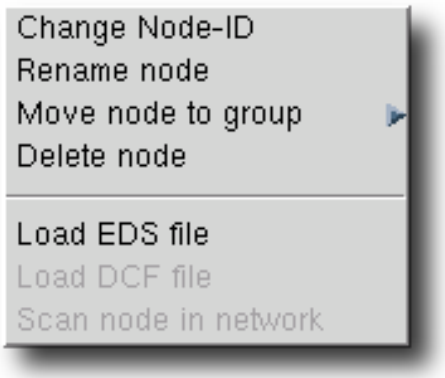


Figure 16, Popup menu of a node

- change node id to 30

- reselect node 30
- select "Device is Master" in the mask tab
- right mouse click on "node 30", select "Load EDS File"
- select EDS file `master_example.eds`
- select group 1
- right mouse click on group 1, select "Add new node"
- create 3 nodes
- change node IDs to 40, 41 and 42
- load EDS files for these nodes according to the table above

- select "CANopen Network"
- set baud rate to 125 kBits/s in mask tab
- Select node 30, activate table tab
- create links according to figure 17

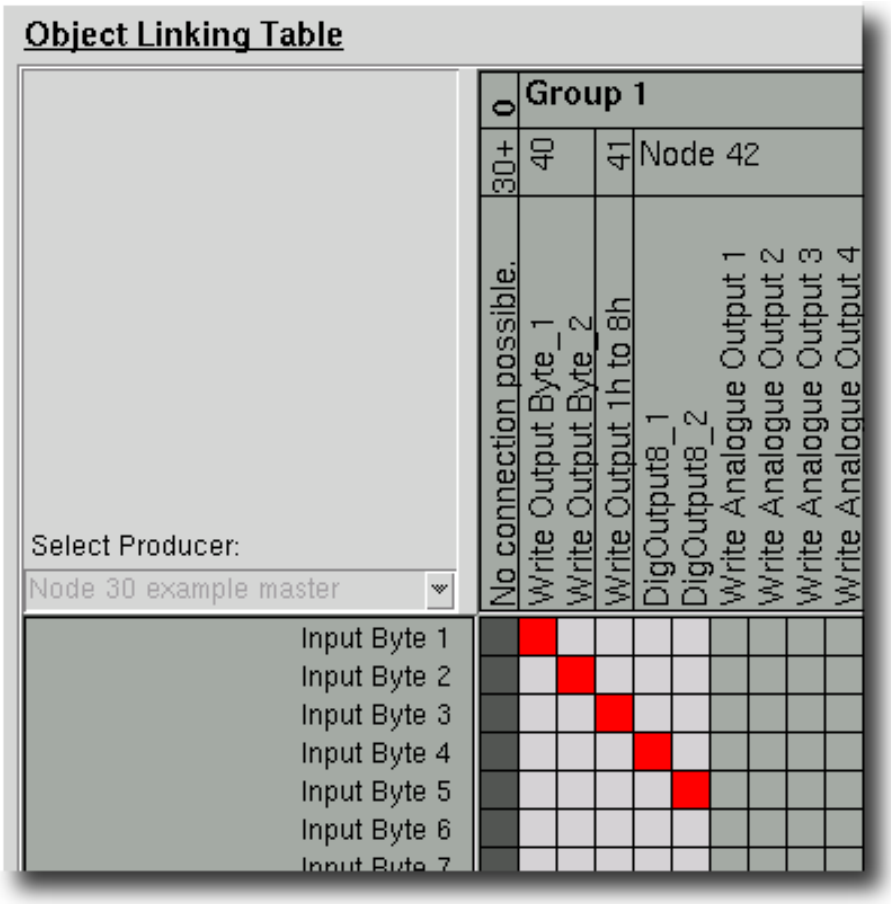


Figure 17, Part of the linking table

- scroll down in the linking table using the right scroll bar
- connect the first 4 Integer16 objects left with the 4 analogue objects of the node 42
- select node 40 in the network tree and activate the table tab
- create more connections according to the list above
- create some other connections
- save the project "File -> Save Project as"
- generate PDO configuration "Data Linking -> Generate PDO Linking"
- Write configuration into DCF files, if no error is reported

The DCF files are now available and can be used to configure the devices.

9.2. Example for unlimited object Linking

9.2.1. Example network

The small example net consists of 3 digital IO-devices that exchange data with each other

The node IDs are distributed as follows:

Node-ID	Device	EDS file
40	CO411A0	`edsfiles/CO4011A0.eds`
41	CANopen Chip164	`edsfiles/COP164_0.EDS`
42	BK5120	`edsfiles/BK1520.eds`

The following connections shall be configured:

Producer	Consumer
Node 40: Input Byte_0	Node 41: Write Output 1h to 8h
Node 40: Input Byte_0	Node 42: DigOutput8_3
Node 41: Read Input 1h to 8h	Node 40: Write Output Byte_1
Node 41: Read Input 1h to 8h	Node 42: DigOutput8_1

9.2.2. Step-by-step guide

- run CANopen Configuration Manager select unlimited object Linking at CANopen Network Settings
- add 3 nodes and load the corresponding EDS file set the node IDs to their right values
- select "CANopen Network"
- set baud rate to 125 kBits/s in mask tab
- select node 40, active table tab activate the connections as according to the list above in contrast to the previous example, any connections are possible now to the same with node 41
- save the project "File -> Save Project as"
- generate PDO configuration "Data Linking -> Generate PDO Linking"
- Write configuration into DCF files, if no error is reported

The DCF file are now available and can be used to configure the devices. If a connection to a CANopen-Server is available, so the configuration data can be sent directly to the devices.

10. CANopen-PLC Support

The CANopen Configuration Manager supports CANopen-PLCs according to DS405. These devices have dynamical entries in to object dictionary, called network variables. The usage of dynamic network variables is defined in *CiA-302 Framework for Programmable CANopen Devices* and in *CiA-405 Interface and Device Profile for IEC 61131-3 Programmable Devices* and shall not be explained here in detail.

In the connection table each CANopen-PLC has one type less network variable as input and output object. If this type less variable is connected to another object, it changes its datatype into the other's one and a name for this network variable is generated automatically based on the node ID and the parameter name of the other object. If two PLCs are connected, a dialog appears that asks for the desired datatype.

The generated DCF file for the CANopen-PLC can be loaded in an IEC61131 development environment (e.g. OpenPCS). There the network variables can be within PLC programs. Usually they have to be declared as extern for this purpose. For further information please have a look into the manual of your IEC61131 development environment.

Additionally the CANopen Configuration Manager creates a file with the variable declarations for each CANopen-PLC. The content of this file may be copied directly into the IEC61131 programming environment.

11. Import of existing projects

Existing projects, which consist of a set of DCF files in one directory, can be imported into the CANopen Configuration Manager. Before the import, please ensure that all EDS files of the used CANopen devices can be found by the CANopen Configuration Manager, too. Put the EDS files either into the EDS repository or into the project directory. The import itself can be started by File → Import Project and it will run autonomously.

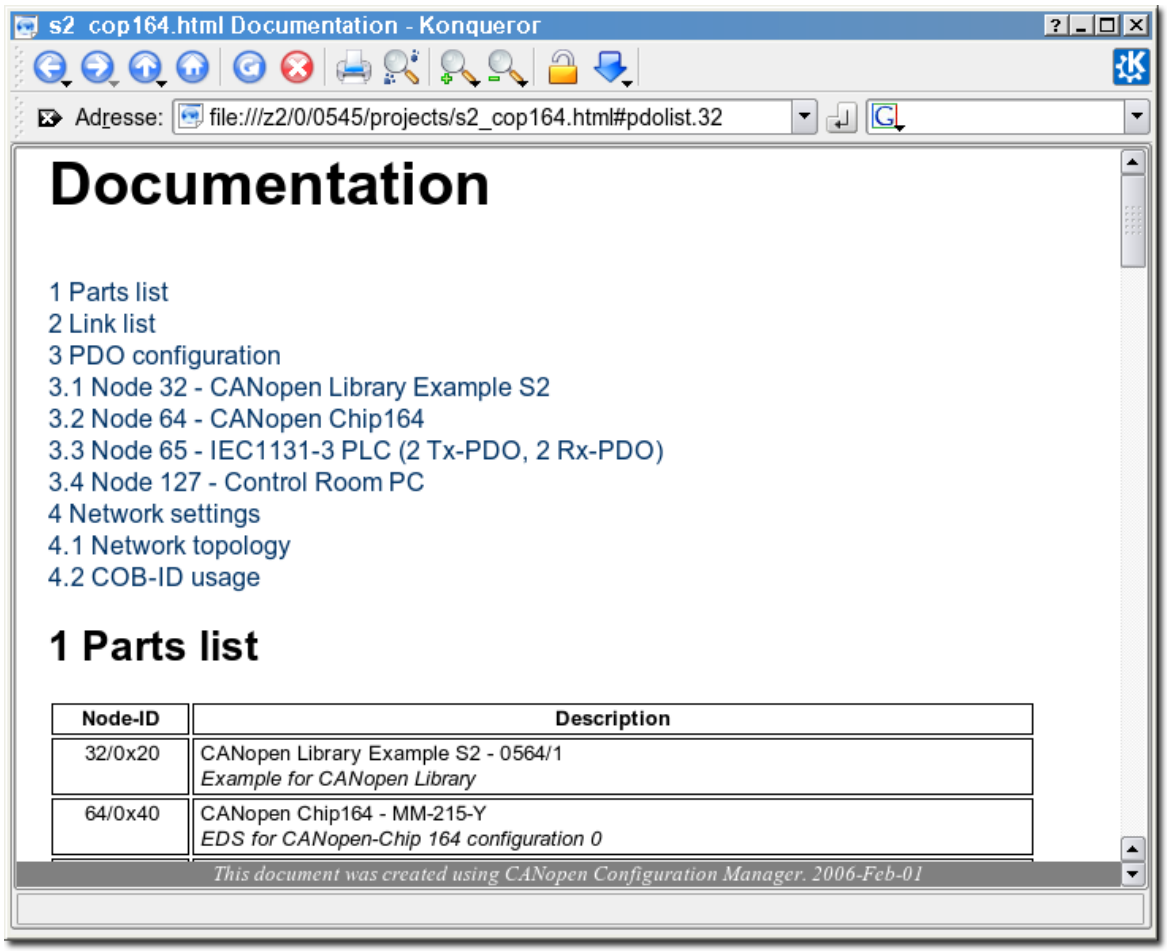
If the configuration of the project is invalid, the import is aborted. Reasons for an invalid configuration may be:

- missing EDS files,
- multiple usage of node-IDs,
- multiple usage of COB-IDs in TPDOs
- or RPDOs without fitting TPDO.

Please try to fix these problems before the import.

12. Documentation of the project

After the output of the DCF files a documentation of the project in HTML is created.



Documentation of the project

The file name of the HTML file is `<ProjectName>.html`. It contains:

- a list of the used devices
- a list of all object links
- the PDO configuration of all nodes
- the network settings
- and a list of the used COB-IDs.

13. Different versions

13.1. Standard versions

Depending on the version the following features are available:

Feature	CCMbasic	CCM
pre-definied connection linking	x	x
Creation of DCF files	x	x
PLC-Support	x	x
unlimited object linking		x
Network access		x
CAN message log		x

13.2. Eval (Demo) version

With the eval versionthe number of nodes is limited to 2 and only 125 kBit/s can be used.

14. About & Release Info Dialog

14.1. About Dialog

The about dialog provides information about:

- the current release
- the type of this release
- the licensee and
- the license.

14.2. Latest Release Info Dialog

When requesting information about the latest release of the CANopen Configuration Manager, a http connection to our server is established and the data (approx. 10 bytes) are downloaded from the server.

NOTHING (except your IP address) IS SENT TO THE SERVER WHEN RETRIEVING THE LATEST RELEASE INFO.

Appendix 1 — DCF creation errors

No connection found.

- No connections between any nodes resp. objects has been found.

More objects to transmit than available mapping entries in transmit PDOs (transmit channels).

- The number of the objects to transmit is larger than the number of available mapping entries in the TPDOs.

More objects to receive than available mapping entries in receive PDOs (receive channels).

- The number of the objects to receive is larger than the number of available mapping entries in the RPDOs.

More data to transmit than available in transmit PDOs (transmit channels).

- The data size (number of bits) of the objects is larger than the available size in the TPDOs.

More data to receive than available in receive PDOs (receive channels).

- The data size (number of bits) of the objects is larger than the available size in the RPDOs.

More data from different nodes to receive than available receive PDOs (receive channels)

- The number of nodes that want to transmit data to this node is bigger than the number of available receive PDOs.

Node xx: RPDO with COB-ID yyy expects n mapping entries, but only m shall be transmitted by the master.

- Error in pre-defined connection linking mode
  This error occurs if less data shall be sent to a node than it expects. In the pre-defined connection linking mode the configuration of the slaves shall not be changed, so it is necessary to send as much as data to the slave as it expects. I.e. The size of the TPDO the master sends to the slave must be the same as the size of the slave's RPDO.

Linking failed due to undefined error at node xx."

- Undefined error at node xx.
- Such error may point to an unrealizable configuration. An example for an unrealizable configuration is:

- a node has only one PDO to transmit
- this node wants to send some data (altogether 8 bytes) to two different devices
- one consumer has only 4 mapping entries per PDO
- this consumer cannot be configured to receive data from an PDO with 4 Dummmy-Mapping entries, and the producer cannot split the data to 2 PDOs