User documentation:
The CoDeSys Gateway Server Manual

Document Version 2.0

tech_doc_e.doc / V1.2

CONTENT

1 The gateway server manual

This chapter gives you a quick overview of the document, helping you to decide whether it's the document you're looking for.

1.1 Overview

The CoDeSys gateway server manual (Gateway Manual) is divided up into the following chapters:

The gateway server manual:
General information about this document and how to use it.

Installing the CoDeSys gateway server:
A short guideline of how to install the gateway server on your PC.

The CoDeSys gateway server:
An overview over the concepts and intentions of the gateway server.

User interface:
Describes the user interface of the gateway server.

Configuring the CoDeSys gateway server:
A description of how to adapt the gateway server to your needs.

A description of the standard drivers and how to configure them is included in the appendix.

1.2 Who is it for?

This document addresses users of the CoDeSys gateway server who want to adapt the gateway server to their needs or who simply want to get an understanding of the general concepts of the gateway server. You will also find information about the user interface and how to install the gateway server. If you want to write your own client or driver, please check out the „Gateway Programmers Guide" and the „Gateway Reference Guide".

As this document isn't about implementing gateway clients or drivers, you don't have to be familiar with things like C++ or the Windows 32 bit API. However, for some chapters you should be familiar with the concepts of the Windows registry.

1.3 Related documents

The GM comes as part of the CoDeSys gateway server SDK (Software Development Kit) documentation. The SDK documentation consists of the following documents:

- The CoDeSys gateway server manual (the one you're looking at right now).
- The CoDeSys gateway server programmers guide (Gateway Programmers Guide): Developing client applications and device drivers step by step.
- The CoDeSys gateway server reference guide: A description of all functions and data types of the various APIs.

2 Installing the CoDeSys gateway server

The easiest way to install the gateway server is to use the standard CoDeSys setup or the separate Gateway Server setup and simply follow the instructions. If you want or have to install the server without one of the setups, here's what you have to consider to have the system run smoothly on your PC:

- On a single PC, there should be only one instance of each gateway file. Different versions of client and server side files can lead to strange behaviour or errors of the server which are hard to track down. See the version tab of the files property dialog to check the versions of the files if you are not sure, whether they match or not. All Files should have the same "Product Version" (Caution: The "Product Version" may differ from the "File Version").
- We recommend to put all files in the Windows system directory or in an own directory which then is included into the systems search path for executables to achieve the point mentioned above.
- Always copy all files of a version to the gateway directory to ensure all have the same "Product version".

Note: We recommend to always use the CoDeSys setup or the separate Gateway Server setup to ensure a working system!

3 The CoDeSys gateway server

The CoDeSys gateway server was developed bearing the following aims in mind:

To allow the integration of custom communication drivers into CoDeSys without having to build a specific executable containing the driver. The customer should be able to develop and integrate drivers without having to contact us to link them into his executable.

The user should be able to log on devices, which use a non network communication with a PC other than the users. E.g. the user can log into a target connected via RS232 to the COM1 port of the PC next to his own.

Third party applications should be able to communicate with the target, even if CoDeSys isn't available to handle the online interface. This should be achieved without the need to extend the runtime system to enable it to handle two clients simultaneously.

Third party applications should be able to use a symbolic interface to the target, without the need to parse any CoDeSys generated files.

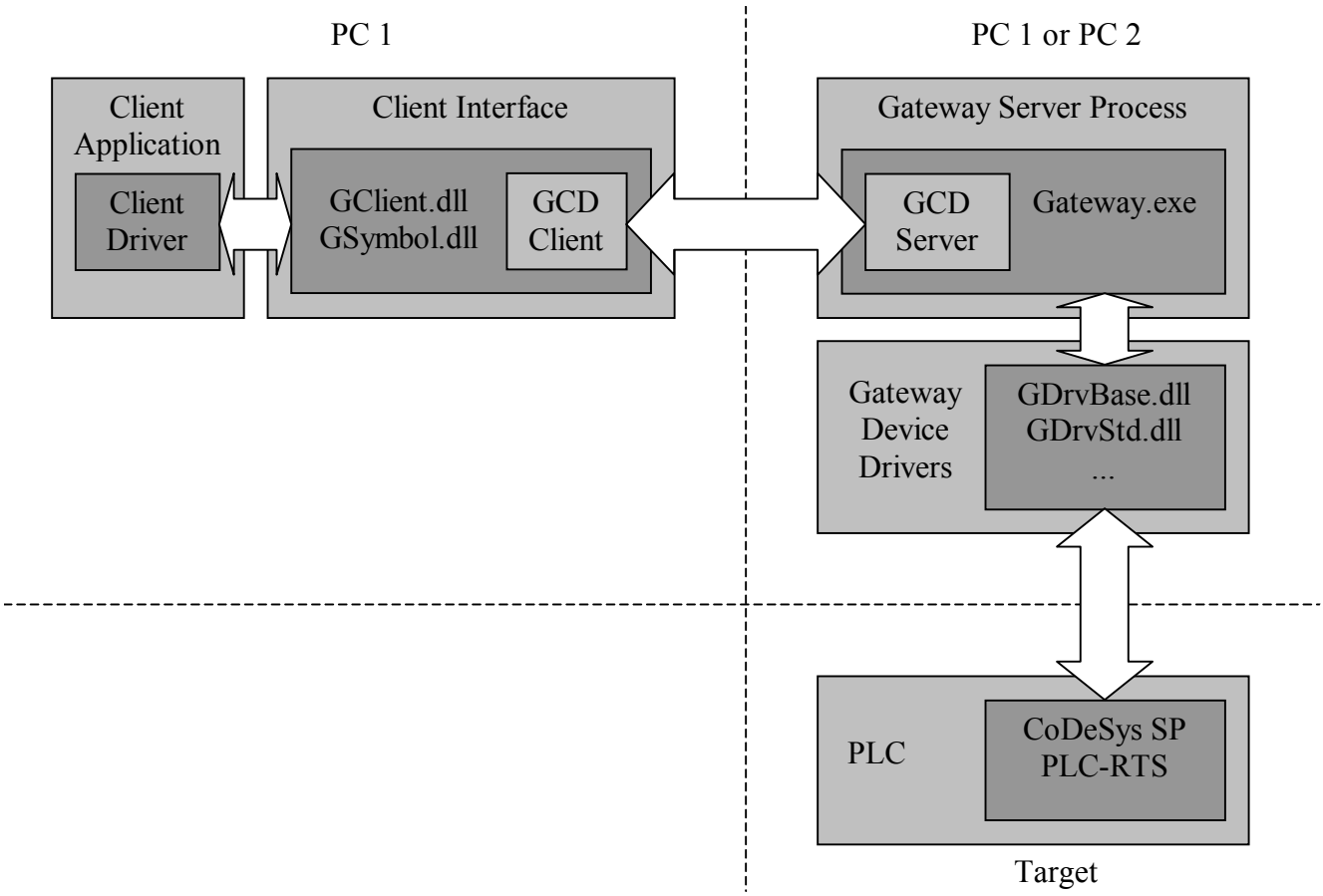
3.1 Client and server

With aims 1 and 2 in mind, the gateway server was divided into three main modules:

The client interface.

The server process

The device drivers



Picture 1: The main modules of the gateway server

A client application links to the dlls of the client interface. The gateway communication drivers (GCD) are the link between the client interface and the gateways server process. The GCDs are real

communication drivers, thus enabling the gateway server process to run on a different PC than the client, using a tcp/ip connection, enabling us to achieve aim 2.

In addition, all information about the devices is transferred to the client in a serialized format. The interfaces of the devices are constructed from this format again. This mechanism encapsulates the device driver from the client side. A driver can be added to the system just by supplying the driver dll and entering it into the registry, so that the gateway server can load it. There's no need to change the client to use the new driver, thus achieving goal 1.

3.2 Sharing devices

To achieve goal 3, the gateway server has to be able to open several channels to one target device. The server does this by returning a unique channel id enabling it to multiplex one device driver between multiple client channels. After a client application has opened a communication channel to the desired device, the connection is transparent to the client, being identified by it's unique channel id.

The client uses the standard level 7 interface of the CoDeSys SP runtime system. As the gateway multiplexes the different channels, the runtime system only communicates with one client at a time. Therefore there's no need to alter the runtime system to enable it to communicate with two clients at a time.

Furthermore, as CoDeSys is simply another client to the gateway server, CoDeSys doesn't have to be active for a third party application to communicate with the target.

3.3 Symbolic access to PLCs

To enable the user to develop client applications without the need to know the intrinsics of the complex CoDeSys SP level 7 interface, the client interface includes functions to access the services via a symbolic interface. E.g. there are services to login, logout, or reading and writing of variables etc. The symbolic interface generates it's information from a symbol database generated by the CoDeSys programming system on compile. This database is transferred to the gateway server and linked to the active channel on download, thus ensuring consistency of the available symbolic information and the program on the target.

4 User interface

4.1 System tray icon

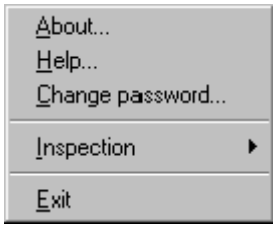
Once the gateway server is running on your system, it displays an icon  in the system tray.

You can tell whether at least one connection is active by the state of the icon. If no connection is active, the icon is shown like above. If an active connection exists, the icon is displayed in highlighted state: 

The menu is accessible only, if no dialog of the gateway server is open.

4.2 Tray menu

If you click the icon with the right mouse button, the following menu is displayed:



'Exit'

Clicking the ‘exit’ command terminates the gateway server. Depending on the settings of the gateway server, this command may be disabled if the are active connections. If it is enabled with active connections, you are queried whether you really want to exit.

'About…'

Shows the about info dialog of the gateway server. It displays some need information like version and service pack numbers.

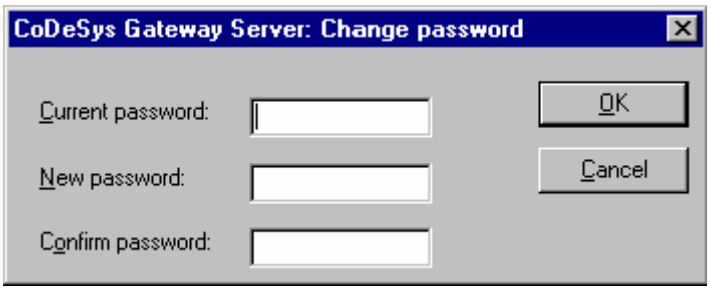
'Help…'

Opens the online help.

'Change password…'

You can achieve a little privacy for the devices connected to your gateways PC by specifying a Gateway Server password. In that case, any client trying to connect to the gateway server is forced to give the correct password or the connection is refused. Furthermore, anybody who wants to use the inspection features of the gateway, has to enter the correct password in the inspection settings (see chapter 4.3.2).

You can specify a password by selecting the ‘Change password’ command. If you do so, the following dialog pops up:



If you want to change the password, you have to enter the current password, the new one and the confirmation of the new one. This is necessary to protect you from accidentally entering a different password than the desired one.

Pressing the ‘OK’ button checks the given passwords, and if everything’s fine, sets the new password and closes the dialog. Choosing the ‘Cancel’ button closes the dialog not performing any operation.

If you have forgotten the password of your gateway, please contact the supplier you received the gateway server from.

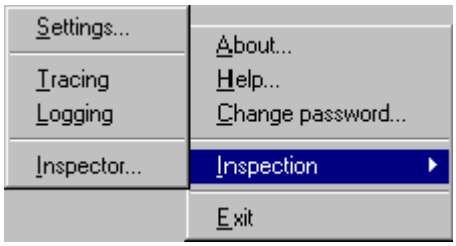
‘Inspection' submenu

The commands of the 'Inspection' submenu are described in chapter 4.3.1.

4.3 Gateway Inspector

4.3.1 'Inspection' submenu

On activation of the 'Inspection' command, the following submenu pos up:



The bottom three commands can only be accessed, if the current inspection password matches the current gateway password.

'Inspection' 'Settings'

Selecting this command opens the 'Inspection settings' dialog for configuring the gateways tracing and logging features. The dialog is described in chapter 4.3.2.

'Inspection' 'Tracing'

Selecting this command enables or disabled the gateways tracing feature. If tracing is enabled, the menu command will be displayed checked, unchecked otherwise. The command can only be selected if the current inspection password matches the gateway password.

If tracing is enabled, a time limited ring buffer will be created where the binary data for each service is stored with a timestamp. This is done independently of the current tracing level. Therefore you can switch between levels and will always get all necessary information for all services.

‘Inspection’ ‘Logging’

Selecting this command enables or disables the gateways logging feature. If logging is enabled, the menu command will be checked, unchecked otherwise. This command can only be selected if the current inspection password matches the gateway password.

If logging is enabled, a log file will be written for each channel. The log files follow the following naming convention:

<Instance name>#<Channel id>(<Driver name> YYYY-MM-DD hh.mm.ss).log.

Each creation of a channel creates a new log file. The logging is done independently from the tracing feature. Therefore it is possible to log channels with the tracing feature disabled.

If the maximum log file size is surpassed, the current log file will be copied to the file <Log file name>_.log. The file <Log file name>.log is emptied and is used again to receive the log output. The file <log file name>_.log contains a reference to the follow up file:

##SUCCESSOR-FILE: <Log directory>\<Log file name>.log ##

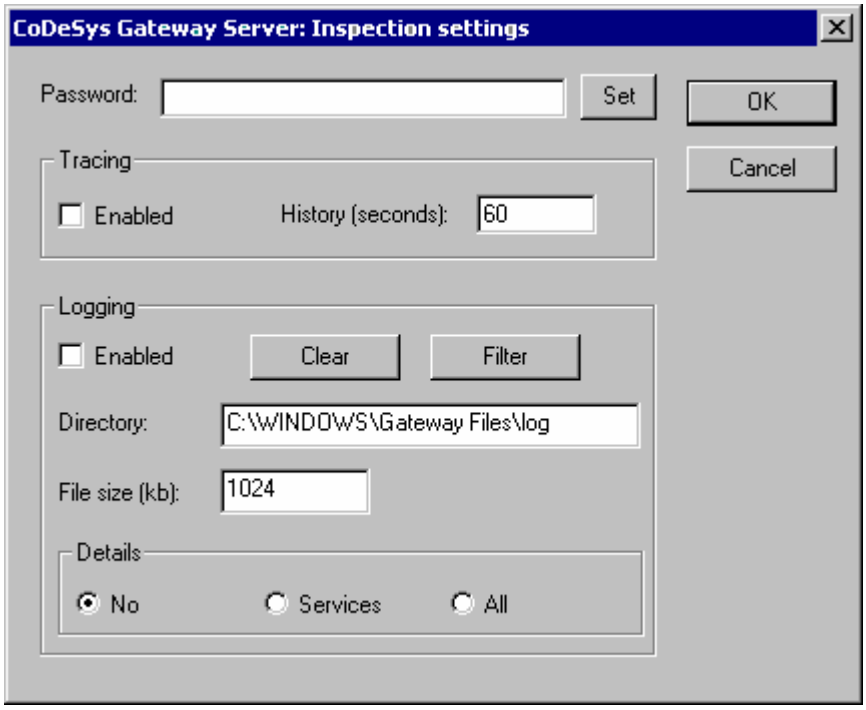
One channel therefore uses a maximum of 2*(maximum log file size) of hard disc space.

'Inspection’ ‘Inspector'

Selecting this command opens the gateways 'Inspector' window described in chapter 4.3.3.

4.3.2 The 'Inspection settings' dialog

This dialog is opened by selecting the 'Inspection:Settings' command from the tray menu or by using the 'Settings' button in the inspector window. The dialog is used to configure the gateways tracing and logging features.



'OK' button

The settings in the dialog are set in the gateway and simultaneously written to the registry.

'Cancel' button

The settings in the dialog are discarded.

'Password' field

To enable the gateways inspection features, you have to enter the current gateway server password in this field. If the passwords do not match, the gateways tracing and logging features will be disabled. The inspection password is not stored in permanent memory, therefore you have to enter it each time the gateway server has been started, to activate the inspection features. The password is set without closing the dialog by using the 'Set' button.

'Set' button

This button sets the given password as the current inspection password. If the password does not match the gateways current password, the gateways tracing and logging features will be disabled. In this case, all elements of the groups 'Tracing' and 'Logging' will be disabled.

'Tracing : Enabled' checkbox

If this checkbox is checked, the gateways tracing feature will be enabled. Otherwise it is disabled. The checkbox reflects the state of the menu command 'Inspection:Tracing'.

'Tracing : History' field

This field denotes the time period for which the data exchange for the channels is monitored in seconds. Entries which are older than this period, will be removed from the trace if new data is recorded. The maximum period is 300 second (5 minutes).

'Logging : Enabled' checkbox

If this checkbox is checked, the gateways logging feature will be enabled. Otherwise it is disabled. The checkbox reflects the state of the menu command 'Inspection:Logging'.

'Logging : Clear' button

Pressing this button deletes all log files from the logging directory.

'Logging : Filter' button

This button opens the dialog for selecting the service filters for data logging. The functionality of the dialog is described in chapter 4.3.4.

'Logging : Directory' field

This field contains the path of the directory, where the log files are stored. If this field is empty the, the directory <Gateway files directory>\log is used. If an invalid directory is given when 'ok' is selected, a message box will pop up and the dialog remain open.

'Logging : File size' field

This field contain the maximum size of a single log file in KByte. This size has to be in the range between 1KByte and 256MByte. If a size outside this range is given, the size will be set to the corresponding border when closing the dialog.

'Logging : Details' radio buttons

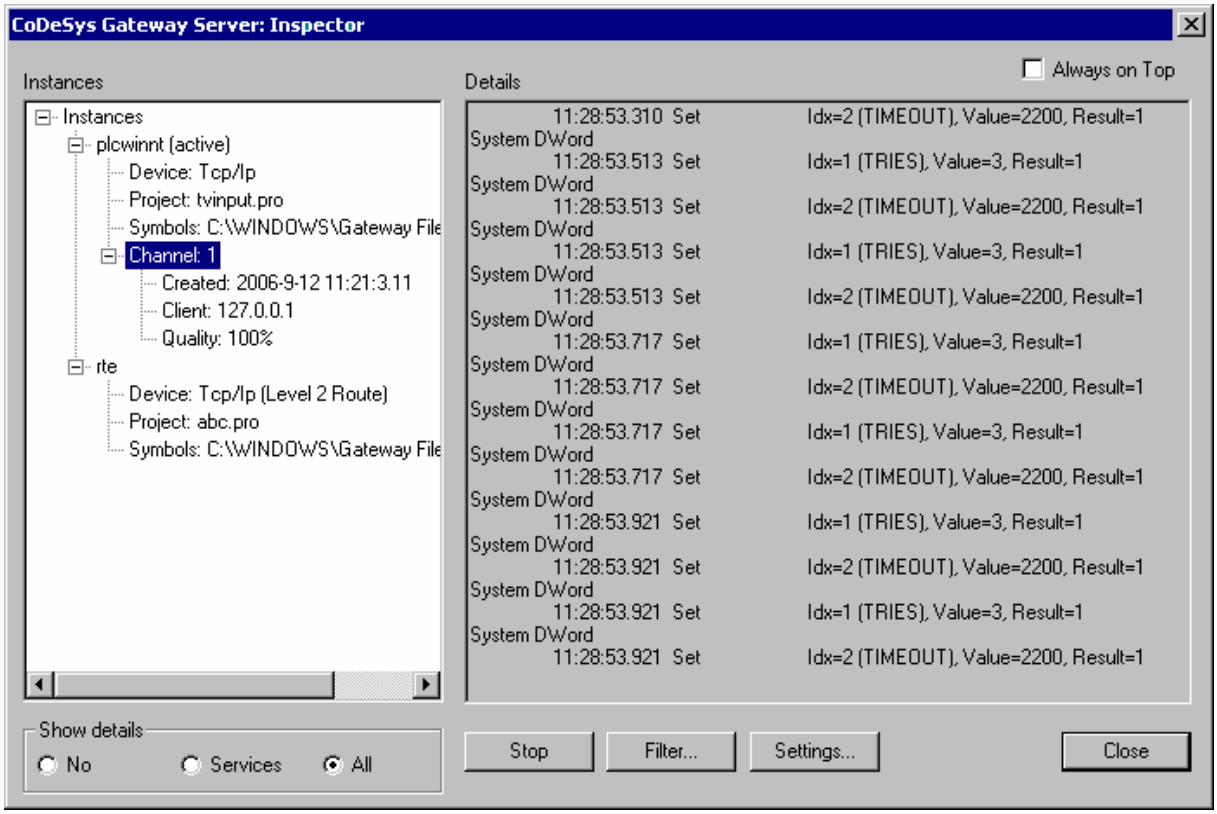
These buttons denote the current level of detailed logging for all channels:

Level	Description
No	Global connection information
Services	Logging of services
All	Binary service data

Each level also contains all information of the lower level.

4.3.3 The 'Inspector' window

You can open the gateways 'Inspector' window by selecting the 'Inspection' 'Inspector' command of the tray menu or by doubleclicking on the gateways tray icon. The contents of the window is continuously updated by the gateway.



'Instances' tree

This tree contains the information of all currently defined driver instances of the gateway. Below the node 'Instances' all driver instances with their user-defined names are shown. If an instance is active (at least one valid connection is handled via it), the string "(active)" will be appended to the name. Furthermore each instance node contains the following child nodes:

- Device: <Name of the driver of this instance>

- Project: <Name of the project the instance is linked to>
- Symbols: <Name of the symbol information file currently used with this instance>

If the instance is active, a child node for each channel will be added:

- Channel: <Unique id of the channel>

The channel nodes contain the following child nodes:

- Created: <Time stamp when the channel has been created>
- Client: <Identification of the Client which has opened the channel>
  Either a Tcp/Ip address or the description of a local client (e.g. Local #0)
- Quality: <Percentage of successfully transmitted blocks>
  If the driver does not support this feature, the string "Unknown" is shown

'Details' field

Dependent of the selected node in the 'Instances' tree, the following information is displayed in this field:

- Node 'Instances': Empty
- 'Instance' nodes: Empty
- 'Device' nodes: Communication parameters of the instance
- 'Project' nodes: Empty
- 'Symbols' nodes: No symbol information file: Empty
  SDB file: Version = <SDB file Version>
  Identity = <Project-Id of the SDB file>
  Number of symbols = <Number of symbols>
  SYM file: ;Version=<SYM file Version>
  ;ProjectId=<Project identity of the SYM file>
- 'Channel' nodes: Empty for level 'No'
  Additional information for the levels 'Services' and 'All'. The shown information is described in chapter 4.3.5
- 'Created' nodes: Empty
- 'Client' nodes: Empty
- 'Quality': Total number of transmitted blocks and the number of failed blocks:

'Always on top' checkbox

If this box is checked, the inspector will remain on top, even if another application is activated.

'No, Services, All' radio buttons

These buttons denote the current level of detailed tracing for all channels:

Level	Description
No	Global connection information
Services	Logging of services
All	Binary service data

Each level contains all information of the lower level.

These buttons can only be selected, if tracing is enabled.

'Start/Stop' button

This button is used to start or stop the monitoring for the channels. If the monitoring is stopped, no information is stored in the internal buffers for the channels and the 'Details' field isn't updated anymore. This way, two channels may be easily compared at a given point in time. If monitoring is started again, the 'Details' field is continuously updated again.

If monitoring is started, the button will display the text "Stop", "Start" otherwise.

These buttons can only be selected, if tracing is enabled.

'Filter' button

This button opens the dialog for defining the service filters for the levels 'Medium' and 'Low' described in 0.

This button can only be selected, if tracing is enabled.

'Settings' button

This button opens the 'Inspection settings' dialog. If an invalid inspection password is given in the dialog, the inspector window will be closed.

'Close' button

Terminates the inspector.

4.3.4 The 'Inspection filter' and 'Logging filter' dialog

This dialog pops up if you select the tracing or logging filter configuration buttons in the 'Inspector' window or the 'Inspection settings' dialog. One configures the trace filter settings, the other the log filter settings.

It is used to configure the service filters for the levels 'Services' and 'All'. Only services which are enabled in the filter are shown in the 'Details' field of the inspector or the log file respectively. The dialog lists all services available in the standard 3S layer 7 protocol.

The 'Ok' button

The given filter settings are set in the gateway and written to the registry.

The 'Cancel' button

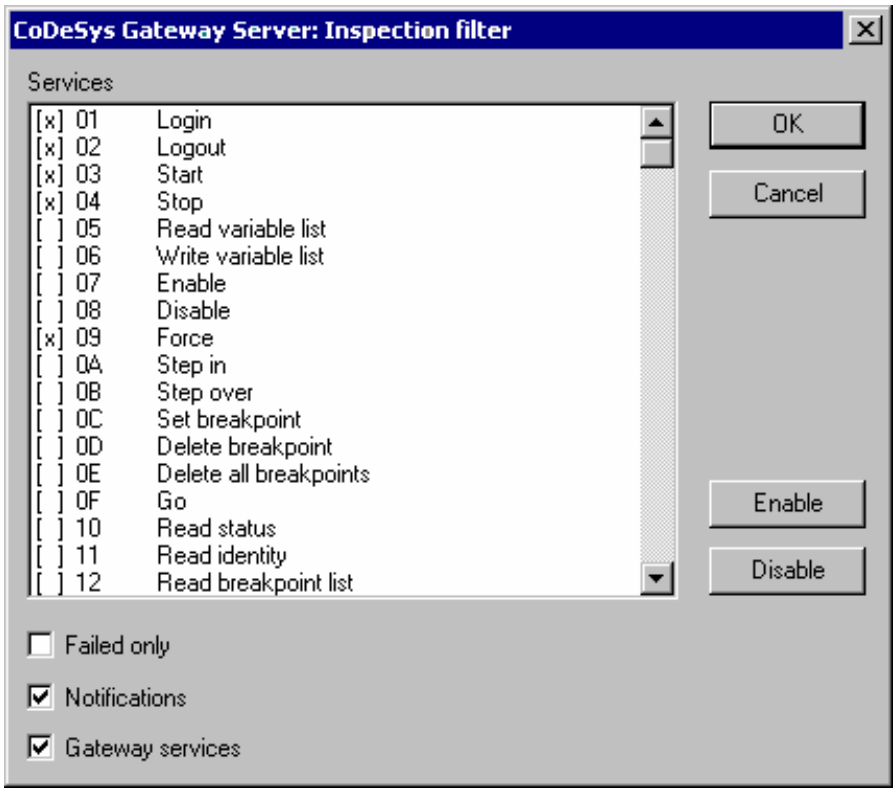
Changes are discarded.

The 'Enable' button

All selected services are enabled.

The 'Disable' button

All selected services are disabled.



The 'Failed only' checkbox

If this checkbox is checked, only failed services will be shown in the trace/log.

The 'Notifications' checkbox

If this checkbox is checked, driver notifications send to the clients will be traced/logged too.

The 'Gateway services' checkbox

If this checkbox is checked, gateway services send to the drivers will be traced/logged too.

4.3.5 Format of the inspection log files

Header

Each log file contains a header with the channels connection information:

##INSPECTOR-LOG-FILE##

##SECTION:HEADER##

Instance=<Instance name>

Driver=<Driver name>

<Name parameter 1>=<Value parameter 1>;<Comment parameter 1>

...

<Name parameter n>=<Value parameter n>;<Comment parameter n>

Project=<Project the instance is linked too>

Channel=<Channel id>

Created=YYYY-MM-DD hh:mm:ss.ms

Client=<Client identification>

Trace

For the levels ‘Services’ and 'All', the trace section of the channel, following the header, contains all enabled services with a time stamp. Thereafter follow the binary data of the service and the reply in hex-ASCII representation.

For enabling or disabling of the logging as well as for changes of the date particular entries will be created.

##SECTION:TRACE##

Example for level ‘Services’:

...

hh:mm:ss.ms Login=Ok

...

##LOGGING STOPPED: 2000-10-12 20:43:13.123##

##LOGGING STARTED: 2000-10-12 20:53:19.223##

...

##NEW DATE: 2000-10-13 ##

...

Example for level 'All’:

...

Read varlist=Ok

06:20:45.846	Tx (1)	05
06:20:45.862	Rx (2)	00 00

..

If the channel is linked to a new project, the following entry will be created:

...

##NEW PROJECT 'Test.pro': 08:34:23.765##

...

If the gateway services are enabled, additional entries will be created for level 'All':

<Gateway Service>

hh:mm:ss.ms <Sub-Service> <Service-Info>

If the log file is copied because the maximum log file size has been surpassed, a reference to the follow-up file will be created:

##SUCCESSOR-FILE: <Log-directory>\<Log-File-Name>.log ##

Summary

If a channel is closed, a summary will be written to the logfile.

##SECTION:SUMMARY##

Terminated=2000-10-13 13:14:16.456

Quality=100%

Total=102354

Failed=0

µs per read=Unknown

5 Configuring the CoDeSys gateway server

The gateway can be configured via the Windows registry.

Note:

- *HKLM* is used as a shortcut for *HKEY_LOCAL_MACHINE*.
- All settings become active after the next start of the gateway server.

5.1 Behaviour and appearance

The gateway server is configured in the registry with the following key:

[HKLM \SOFTWARE\3S-Smart Software Solutions GmbH\Gateway Server\Config]

With the following values, the behaviour and appearance of the server can be changed.

5.1.1 Debug (DWORD)

Entering a value other than zero, enables the debugging mode of the gateway server. If debug is enabled, the gateway dumps debug information for each message to the standard debug output. The default value is 0, hence no debugging mode.

Example:

"Debug"=dword:00000001

Enables the debugging mode.

5.1.2 DebugFile (STRING)

If this value exists and is nonempty, the debug output of the gateway server will be redirected to this file. The default value is the empty string, hence disabling the redirection.

Example:

"DebugFile"="C:\WINDOWS\Gateway Files\log\gateway.log"

Redirects the debug output to "C:\WINDOWS\Gateway Files\log\gateway.log".

5.1.3 DebugFileSize (DWORD)

This value sets the maximum size of the debug file. If the debug file reaches this size, the current debug file is renamed to <DebugFileName>_.<DebugFileExtension>. Subsequent debug output is written to <DebugFile> again. Therefore a maximum of twice the given size is located on harddisk. The default value is 0x100000 (1Mbyte).

Example:

"DebugFileSize"=dword:00010000

Sets the maximum size to 64Kbyte.

5.1.4 ExitOnActiveInstances (DWORD)

Entering a value other than zero, enables the debugging mode of the gateway server. If debug is enabled, the gateway dumps debug information for each message to the standard debug output. The default value is 1, hence the gateway server can be terminated while active connections exist.

Example:

"ExitOnActiveInstances"=dword:00000000

The gateway server can’t be quit via the menu, while there are active connections.

5.1.5 Name (STRING)

With this value, you can set the name of the gateway server as it appears in the tray icons tooltip and the about info dialog. The default value is “CoDeSys Gateway Server”.

Example:

"Name"="My Gateway Server"

The tooltip displays “My Gateway Server (idle)”.

5.1.6 Priority (STRING)

The priority of the gateway servers main thread can be set with this value. The following strings are valid:

“HIGH”: The gateway runs with high priority.

“REALTIME”: The gateway runs with realtime priority.

All other values let the gateway run at standard priority. The default is standard priority.

Note: The values are compared case-insensitive.

Example:

"Priority"="High"

Lets the gateway run at high priority.

5.1.7 Files (STRING)

A client can send files to the gateway server, e.g. to save symbolic information for a specific connection. With this value you can specify a directory, where such files are placed. The default value is "<Windows Dir>\Gateway Files".

Example:

"Files"="C:\My Gateway Files"

All files are placed and searched in "C:\My Gateway Files".

Caution: Altering this value after active connections had been established may result in invalid symbol information for these connections.

5.2 The gateways Tcp/Ip remote interface

The gateway server contains a Tcp/Ip server, so it can be accessed from remote PCs. The Tcp/Ip server can be configured in the key:

[HKLM\SOFTWARE\3S-Smart Software Solutions GmbH\Gateway Server\Config\TcpIp]

5.2.1 Port (DWORD)

The port number of the server is given with this value. The default is 1210 (04BAHex).

Example:

"Port"=dword:000004b0

Sets the port number of the Tcp/Ip server to 1200.

5.3 Adding custom drivers

As described above, custom drivers can be made available to all clients via the registry. The drivers for the gateway are given in the key:

[HKLM \SOFTWARE\3S-Smart Software Solutions GmbH\Gateway Server\Drivers]

To add a driver, all you have to do is add a key to this key and specify the location of your driver dll in the value “Path” in it.

Example:

[HKLM \SOFTWARE\3S-Smart Software Solutions GmbH\Gateway Server\Drivers\My Driver]

"Path"="C:\WinNT\System32\MyDriver.dll"

If the driver dll complies to the interface of a gateway server driver dll, it is added to the gateway.

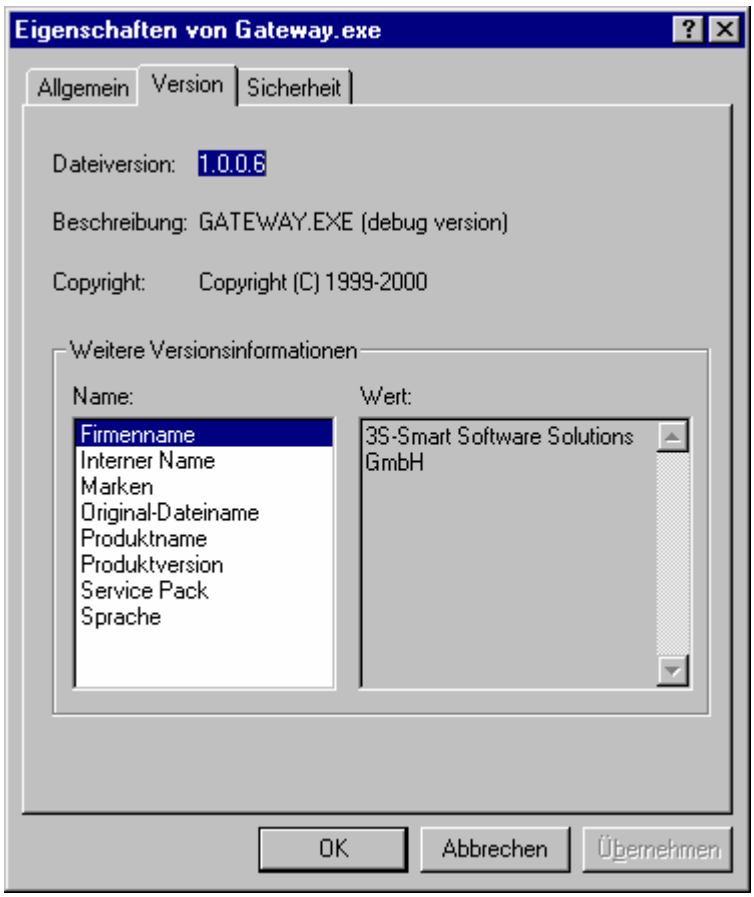
Note: The gateway server always adds the standard 3S driver dll at startup.

5.4 The gateway files properties

All files of the gateway server contain a version tab in their property dialog. Information like the files version, the products version or the service pack number etc. can be obtained from this dialog. If you want to check whether your installation consists of files of the same product version, this is the way to do it.

Example:

Version properties of gateway.exe.



6 Configuring the CoDeSys gateway client dll

The gateway client dll can be configured via the Windows registry.

Note:

- *HKLM* is used as a shortcut for *HKEY_LOCAL_MACHINE*.
- All settings become active after the next start of the client. Exceptions are noted within the setting.

6.1 Behaviour and appearance

The gateway server is configured in the registry with the following key:

[HKLM \SOFTWARE\3S-Smart Software Solutions GmbH\Gateway Server\Config\Clients]

With the following values, the behaviour and appearance of the server can be changed.

6.1.1 Debug (DWORD)

Entering a value other than zero, enables the debugging mode of the client dll. If debug is enabled, the client dll dumps debug information for each message to the standard debug output. The default value is 0, hence no debugging mode.

Example:

"Debug"=dword:00000001

Enables the debugging mode.

6.1.2 DebugFile (STRING)

If this value exists and is nonempty, the debug output of the client dll will be redirected to a file using this entry as a template. The file name will be build of <DebugFileName>(<Client executable name>).<DebugFileExtension>. The default value is the empty string, hence disabling the redirection.

Example:

"DebugFile"="C:\WINDOWS\Gateway Files\log\clients.log"

CoDeSys.Exe would produce a debug file "C:\WINDOWS\Gateway Files\log\clients(CoDeSys).log".

6.1.3 DebugFileSize (DWORD)

This value sets the maximum size of the debug file. If the debug file reaches this size, the current debug file is renamed to <DebugFileName>_.<DebugFileExtension>. Subsequent debug output is written to <DebugFile> again. Therefore a maximum of twice the given size is located on harddisk. The default value is 0x100000 (1Mbyte).

Example:

"DebugFileSize"=dword:00010000

Sets the maximum size to 64Kbyte.

6.1.4 AutostartTimeout (DWORD)

The timeout used for the automatic startup of the local gateway server on connect can be set with this value. The timeout is given in milliseconds. The default value is 50000 (50 seconds). Any client connect function will check for the specified timeout whether the local gateway is running. If no running gateway can be detected after the timeout, the connect function returns with an error. Changing this value takes effect with the next connect.

Example:

"AutostartTimeout"=dword:000186A0

Sets the timeout to 100 seconds.

6.1.5 Ping (DWORD)

If this value is not 0, the client dll checks whether a remote gateway's address is available with a prefixed ping on connect. This enables the client dll to avoid lengthy socket timeouts when connecting to a non existing address. The default value is 1, hence pinging is enabled by default. However, some firewalls do not allow ping services to be passed. In this case, set this value to 0 to suppress the ping. Changing this value takes effect with the next connect.

Example:

"Ping"=dword:00000000

Disables pinging.

Appendix A: Standard driver settings

The gateway server installation contains various drivers you can use right away. Each driver has various settings to adapt it's behaviour. The drivers are configured in the registry with subkeys of the following key:

[HKLM\SOFTWARE\3S-Smart Software Solutions GmbH\Gateway Server\Drivers\Standard\Settings]

This key will be referred to as *HK_SETTINGS* in the following.

A.1 Common settings

Some of the drivers have some settings in common. These are listed below. If a driver does not support the setting or differs in default value or interpretation, this is described in the drivers section.

A.1.1 Enable (DWORD)

All drivers can be en- or disabled depending on the devices you're using. Setting this value to zero disables the driver. This means, that it's not contained in the gateways driver list and therefore cannot be selected by the client applications. This avoids overloading the client dialogs with drivers which are not needed. See the drivers section whether a driver is initially enabled or not.

"Enable"=dword:00000000

The driver is disabled.

A.1.2 Motorola byteorder (STRING)

With this value, you can specify whether a driver expects a system with motorola byteorder or not. If this value is "Yes", a system with motorola byte order is expected. All others yield intel byteorder.

Example:

"Motorola byteorder"="Yes"

A system with motorola byteorder is expected.

A.1.3 Buffersize (DWORD)

All drivers can work with a given target buffersize to avoid conflicts with the target runtime systems. You can specify the targets communication buffersize in bytes with this value. The default values are given with the drivers in the respective sections. A buffersize of size 0 means unlimited buffersize.

Example:

"Buffersize"=dword:00001000

The driver uses a communication buffer of 4096 bytes.

A.2 Tcp/Ip

The standard Tcp/Ip driver is configured with values in the following key:

[HK_SETTINGS\Tcp/Ip]

The default values for the common settings are as follows.

A.2.1 Common settings default values

"Enable"=dword:00000001

"Motorola byteorder"="No"

"Buffersize"=dword:00000000

In addition, the following values can be set:

A.2.2 Specific settings

A.2.2.1 Address (STRING)

With this value, you can specify the drivers default Tcp/Ip address. This value may be a host name or a dotted Tcp/Ip address. The default is “localhost”.

Examples:

“Address”=”localhost”

or

“Address”=”127.0.0.1”

The local PC is searched in each case.

A.2.2.2 Port (DWORD)

With this value, you can specify the drivers default Tcp/Ip port. The default value is 1200 (04B0Hex).

Example:

“Port”=dword :000004b1

The port is set to 1201.

A.2.2.3 Ping (DWORD)

If this value is not 0, the driver checks whether the targets address is available with a prefixed ping on connect. This enables the driver to avoid lengthy socket timeouts when connecting to a non existing address. The default value is 1, hence pinging is enabled by default. However, some firewalls do not allow ping services to be passed. In this case, set this value to 0 to suppress the ping. Changing this value takes effect after restarting the gateway server or with every new driver instance.

Example:

"Ping"=dword:00000000

Disables pinging.

A.3 Tcp/Ip (Level 2)

The standard Tcp/Ip level 2 driver is configured with values in the following key:

[HK_SETTINGS\Tcp/Ip (Level 2)]

The default values for the common settings are as follows.

A.3.1 Common settings default values

“Enable”=dword:00000001

“Motorola byteorder”=”No”

“Buffersize”=dword:00001388 (5000Byte)

A.3.2 Specific settings

In addition to the value set of the Tcp/Ip driver, the driver supports the following settings.

A.3.2.1 Blocksize (DWORD)

You can specify the drivers blocksize used for blocking long messages with this value. The default value is 128KByte.

Example:

“Blocksize”=dword:00001000

The driver uses a communication block size of 4096 bytes.

A.4 Tcp/Ip (Level 2 Route)

The standard Tcp/Ip level 2 routing driver is configured with values in the following key:

[HK_SETTINGS\Tcp/Ip (Level 2 Route)]

The default values for the common settings are as follows.

A.4.1 Common settings default values

"Enable"=dword:00000001

"Motorola byteorder"="No"

"Buffersize"=dword:00020000 (128KByte)

A.4.2 Specific settings

In addition to the value set of the Tcp/Ip driver, the driver supports the following settings.

A.4.2.1 Target Id (DWORD)

With this value, you can specify the drivers default target id. The default value is 0.

Example:

"Target Id"=dword:00000020

Sets the default target id to 32.

A.5 Pipe

The standard Tcp/Ip driver is configured with values in the following key:

[HK_SETTINGS\Pipe]

The default values for the common settings are as follows.

A.5.1 Common settings default values

"Enable"=dword:00000001

"Motorola byteorder": Not supported

"Buffersize"=dword:00000000

A.5.2 Specific settings

A.5.2.1 Pipe (STRING)

With this value, you can specify the name of the communication pipe which is used as default. This value has to be a valid pipe name. The default is "CoDeSysSPCommunication".

Examples:

"Pipe"="mypipe"

"mypipe" is set as the default pipe name.

A.5.2.2 Timeout (DWORD)

With this value, you can specify the default value in seconds for the timeout used when opening the pipe. The default is 60 seconds.

Examples:

"Timeout"=dword :00000064

The default timeout is set to 100 seconds.

A.6 Serial (RS232)

The standard serial 16 bit RS232 driver is configured with values in the following key:

[HK_SETTINGS\Serial (RS232)]

The default values for the common settings are as follows.

A.6.1 Common settings default values

“Enable”=dword:00000001

“Motorola byteorder”=”No”

“Buffersize”=dword:00001388 (5000Byte)

A.6.2 Specific settings

A.6.2.1 Port (STRING)

With this value, you can specify the drivers default COM port. The default value is “COM1” or the first available COM port found by the system. Valid values are strings with the format “COM<x>”, where <x> is the number of the COM port.

Example:

“Port”=”COM2”

COM port 2 is set as the default port.

A.6.2.2 Baudrate (STRING)

With this value, you can specify the drivers default baudrate. The default value is “38400”. Valid values are: “4800”, “9600”, “19200”, “38400”, “57600” or “115200”.

Example:

“Baudrate”=”19200”

Sets the default baudrate to 19200 baud.

A.6.2.3 Parity (STRING)

With this value, you can specify the drivers default parity. The default value is no parity. Valid values are: “No”, “Even”, “Odd”.

Example:

“Parity”=”Odd”

Sets the default parity to odd.

A.6.2.4 Stop bits (STRING)

With this value, you can specify the drivers default stop bits setting. The default value is one stop bit. Valid values are: “1”, “1,5”, “2”.

Example:

“Stop bits”=”1,5”

Sets the default stop bits setting to 1.5 stop bits.

A.6.2.5 Flow Control

With this setting the data flow control (“handshake”) for the serial communication can be activated (“On”) or deactivated (“Off”). Therefore, a special cable is necessary (no null modem cable!).

A.6.2.6 COM-Port scan for Windows NT

The available COM ports in the driver instance can be identified by a scan for Windows NT. If the scan is not activated, 32 COM ports are available. This setting applies to all standard drivers using COM ports. The scan can be enabled with the following value:

[HKLM\SOFTWARE\3S-Smart Software Solutions GmbH\Gateway Server\Drivers\Standard]

"EnableComPortScan"=dword:00000001

The default setting is for gateway servers from version 2.3.6.0 is

"EnableComPortScan"=dword:00000000

Therefore the COM port scan is disabled by default. For previous versions, the scan is enabled by default.

A.7 Serial (RS232, 8 bit)

The standard serial 8 bit RS232 driver is configured with values in the following key:

[HK_SETTINGS\Serial (RS232, 8 bit)]

The default values for the common settings are as follows.

A.7.1 Common settings default values

"Enable"=dword:00000001

"Motorola byteorder": Not supported

"Buffersize"=dword:00001388 (5000Byte)

A.7.2 Specific settings

The value set equals that of the serial 16 bit RS232 driver.

A.8 Serial (RS232 Router)

The standard serial 16 bit RS232 router driver is configured with values in the following key:

[HK_SETTINGS\Serial (RS232 Router)]

The default values for the common settings are as follows.

A.8.1 Common settings default values

"Enable"=dword:00000001

"Motorola byteorder"="No"

"Buffersize"=dword:00001388 (5000Byte)

A.8.2 Specific settings

The value set equals that of the serial 16 bit RS232 driver.

In addition, the following value can be set.

A.8.2.1 Target Id (DWORD)

With this value, you can specify the drivers default target id. The default value is 0.

Example:

"Target Id"=dword:00000020

Sets the default target id to 32.

A.9 ASI Controller

The standard ASI controller driver is configured with values in the following key:

[HK_SETTINGS\ASI Controller]

The default values for the common settings are as follows.

A.9.1 Common settings default values

"Enable"=dword:00000000

"Motorola byteorder": Not supported

"Buffersize"=dword:00001388 (5000Byte)

A.9.2 Specific settings

In addition to the value set of the serial 8 bit RS232 driver, this driver supports the following settings.

A.9.2.1 ASI Controller ID (DWORD)

With this value, you can specify the drivers default ASI controller id. The default value is 1.

Example:

"ASI Controller ID""=dword:00000010

The ASI controller id is set to 16.

A.10 Serial (Modem)

The standard serial modem driver is configured with values in the following key:

[HK_SETTINGS\Serial (Modem)]

The default values for the common settings are as follows.

A.10.1 Common settings default values

"Enable"=dword:00000001

"Motorola byteorder"="No"

"Buffersize"=dword:00001388 (5000Byte)

A.10.2 Specific settings

The value set equals that of the serial 16 bit RS232 driver.

In addition, the following values can be set.

A.10.2.1 Init (STRING)

With this value, you can specify the drivers default init string sent to the modem on initialization. The default value is "AT&F0".

Example:

"Init"=""

No string is sent on initialization.

A.10.2.2 Dial (STRING)

With this value, you can specify the drivers default dial string prefix. The default value is "ATDT".

Example:

"Dial"="ATDP"

Pulse dialling is selected.

A.10.2.3 Hangup (STRING)

With this value, you can specify the drivers default hangup string. The default value is “ATH”.

Example:

“Hangup”=””

No hangup string is used.

A.10.2.4 Break (STRING)

With this value, you can specify the drivers default break string, used to set the modem in command mode. The default value is “+++”.

Example:

“Break”=”++++++”

A.10.2.5 Timeout (DWORD)

With this value, you can specify the drivers default timeout in seconds used while trying to establish a connection. The default value is 60 seconds.

Example:

“Timeout”=dword:00000064

Sets the default timeout to 100 seconds.

A.11 Modem (RS232) (Level 2 Route)

The standard serial modem driver is configured with values in the following key:

[HK_SETTINGS\Modem (RS232) (Level 2 Route)]

The default values for the common settings are as follows.

A.11.1 Common settings default values

“Enable”=dword:00000001

“Motorola byteorder”=”No”

“Buffersize”=dword:00001388 (5000Byte)

A.11.2 Specific settings

The value set equals that of the serial modem driver.

In addition, the following values can be set.

A.11.2.1 Target Id (DWORD)

With this value, you can specify the drivers default target id. The default value is 0.

Example:

“Target Id”=dword:00000020

Sets the default target id to 32.

A.12 CANopen DSP302

The standard CANopen DSP302 driver is configured with values in the following key:

[HK_SETTINGS\CANopen DSP302]

The default values for the common settings are as follows.

A.12.1 Common settings default values

“Enable”=dword:00000001

“Motorola byteorder”: Not supported

“Buffersize”=dword:00001388 (5000Byte)

A.12.2 Specific settings

A.12.2.1 CAN card driver (STRING)

With this value, you can specify the drivers default can driver to use. The default value is “CAN_VCI”.

Example:

“CAN card driver”=”mycan”

the mycan.dll is used as the base for the driver.

A.12.2.2 Expedited(DWORD)

With this value, you can specify whether the driver uses expedited telegrams or not. If you specify 0, no expedited telegrams are used. The default value is expedited telegrams are used.

Example:

“Expedited”=dword:00000000

No expedited telegrams are used.

Change History

Version	Description	Date
1.0	Issued	10.08.2000
1.1	Gateway Inspector	17.01.2001
1.2	Extensions: 5.1.2: DebugFile, 5.2.3: DebugFileSize, 6: Configuring the CoDeSys gateway client dll, A.3.2.1: Blocksize, A.4.2.1: Target ID, new A.11: Modem RS232 Level 2 Route (Improvement #4062)	12.10.2005
1.2	Review (ok), Release	12.10.2005
1.3	Changes: 6.1.5: Added info about firewalls Extensions: A2.2.3: Added description for parameter ping	18.11.2005
1.3	Review (ok), Release	18.11.2005
1.4	Ch.6.1: Path corrected, Release with CoDeSys V2.3.6.0	16.01.2006
1.5	A.6.2.5: COM port scan	27.01.2006
1.5	Formal Review, Heading levels corrected, Release	27.01.2006
1.6	Ch.4.3 Rework (#6312)	12.09.2006
1.6	Review	27.09.2006
1.6	Release (with CoDeSys+Gateway V2.3.7.0)	27.09.2006
2.0	RS232 handshake option added (#6805), Ch. A 6.2.5	27.04.2007
2.0	Review	27.04.2007
2.0	Release (with CoDeSys+Gateway V2.3.8.0)	27.04.2007