zenon manual

Message Control

v.7.11

©2014 Ing. Punzenberger COPA-DATA GmbH

All rights reserved.

Distribution and/or reproduction of this document or parts thereof in any form are permitted solely with the written permission of the company COPA-DATA. The technical data contained herein has been provided solely for informational purposes and is not legally binding. Subject to change, technical or otherwise.

Contents

1. Welcome to COPA-DATA help 5

2. Message Control 5

3. Compatibility with version 6.xx 6

4. Requirements 9

5. Supported AT commands 11

6. Configure Message Control 13

6.1 General settings for sending 15

6.1.1 E-mail message via Outlook 16

6.1.2 E-mail message via SMTP 17

6.1.3 SMS message via GSM modem 19

6.1.4 SMS message via SMS gateway 23

6.1.5 Voice message audiofile via modem 25

6.1.6 Voice message text-to-speech via modem 26

6.1.7 zenon6.ini entries 28

6.2 Project-specific settings 34

6.3 Create a screen of type Message Control 37

6.4 User administration 40

6.5 Functions 44

6.5.1 Screen switching to a screen of type Message Control 44

6.5.2 Save current queue 48

6.5.3 Suppress groups/classes/areas/equipment 49

6.5.4 Send a Message 54

6.5.5 Send Message: activate 73

6.5.6 Send Message: deactivate 73

6.6 Parameters for messages 74

6.6.1 Text from limit value and free text 75

7. Message Control in Runtime 80

7.1 Connect screen of the type Message Control 82

7.2 Acknowledgement of messages 85
7.3 Network 89

8. Messages and error handling 90

8.1 Check list 92
8.2 Log entries 96

1. Welcome to COPA-DATA help

GENERAL HELP

If you cannot find any information you require in this help chapter or can think of anything that you would like added, please send an email to documentation@copadata.com (mailto:documentation@copadata.com).

PROJECT SUPPORT

You can receive support for any real project you may have from our Support Team, who you can contact via email at support@copadata.com (mailto:support@copadata.com).

LICENSES AND MODULES

If you find that you need other modules or licenses, our staff will be happy to help you. Email sales@copadata.com (mailto:sales@copadata.com).

2. Message Control

The optional module Message Control allows for an automatic sending and acknowledgement of messages. The sending is triggered via a function which can be linked with an event. Available media for sending include the following:

- E-mail message via Outlook
- E-mail message via SMTP

- SMS message via GSM modem
- SMS message via SMS gateway
- Voice message with audio file via modem on telephone
- Voice message text-to-speech via modem on telephone

The status of this transmission is logged in the Chronologic Event List (CEL).

The configuration takes place in two stages:

- General settings for sending: in the properties of the `Message Control` group of the workspace.
- Project-specific settings: in the properties of the Message Control node in the respective project.

> **License information**
>
> *Must be licensed for Editor and Runtime (single-user, server, standby). Not available for clients.*

CONTEXT MENU PROJECT-SPECIFIC SETTINGS IN PROJECT MANAGER.

Menu item	Action
Help	Opens online help.

3. Compatibility with version 6.xx

CHANGES IN VERSION 7 COMPARED TO OLDER VERSIONS

As of version zenon 7.00 SP0 module Message Control differs basically from earlier versions in terms of technology and configuration.

Important technical changes:

- COM Server is no longer used

- the additional component of company DerDack is no longer used
- the ZenMsgQueue is replaced by an own screen of type `Message Control`
- the configuration is carried out in property `Message Control` of the workspace (sending) and properties `Project-specific settings` for module Message Control in the project (project-specific)
- there is no detail view anymore
- the shift model and the calendar functionality has been removed
- there are no Runtime changeable files anymore
- Sending e-mails is possible via Outlook or a SMTP Server whereas SMTP allows the sending of attachments
- the configuration of the sending type is no longer saved in file message32.ini but in file zenon6.ini
- Evaluating the limit texts: Up to now the evaluation of compound texts in module Message Control differed from the evaluation of standard limit texts. From version 7.00 on both are evaluated in the same way. **`@StringTabelle+%var1`**

You can find details about the configuration in chapter Configure Message Control (on page 13).

> **Attention**
>
> *Only projects from version 5.50 SP7 on can be converted to version 7.*

CONVERSION

Due to the profound changes a 100% compatibility cannot be guaranteed for the conversion. This is also true for compiling RT files for older versions. At converting especially take care for:

- User:
  - Users with the same name (first name, last name) existing: User is used and information is added.
  - No according user available: A new user is created. The link to the replacement and to the user group is resolved. The user is added to the existing or at the conversion to the created group.
- User groups:

- User group with same name exists: User group is used and information is added.
- No according user group available: A new user group is created.

- Functions:

`Show recipient-database` function was removed.
This function can no longer be created with the Editor. At the conversion it is not deleted however. Its call up in the Runtime has no effect and creates a log entry (on page 90).

- Paging:
  Paging is no longer available as sending type. Existing functions with sending type `Paging` are changed to sending type `GSM` at the conversion. A message (on page 90) in the output window indicates this. After the conversion you must check the settings of the function.

- RT changeable files:
  As the user administration was changed, RT changeable files are no longer used for Message Control. There is no possibility in version 7 to read back old Runtime data. If the Runtime files of a project prior to version 7 are needed, you must read them back in an Editor prior to zenon 7 and then converted.

- Shifts and calendars:
  The functionality for shifts and calendars was removed. Existing functions with target type **`Shift`** are changed to target type **`Group`** at the conversion. However no group is linked. A message (on page 90) in the output window indicates this. After the conversion you must check the settings of the function.

- SMS-Gateway:
  As the simple interface does not offer a technical possibility to assign messages distinctly, from version 7 on only the enhanced interface is supported. At conversion you must make sure that the SMS Server from company Dialogs is configured correspondingly. Otherwise the sending fails.

IMPORT AND EXPORT

Previous versions of message control and versions from zenon 7.00 SP0 on save their settings in different files:

- up to 6.51 SP0:
- from version 7.00 SP0 on: zenon6.ini

IMPORT SETTINGS

If a previous version of zenon is opened in version 7 or higher the settings from the previous message32.ini are taken over and adapted as far as possible. The configuration always has to be checked manually since not all settings can be converted 100% correctly. For instance, shift schedules are assigend to groups which have to be further configured.

EXPORT SETTINGS

The configuration can be written from version 7 to the message32.ini for previous versions. To do so, click on the global working area property `Message Control`/`Export settings to Messag32.ini (v6.XX)` and confirm the security query.

Attention: By executing the property `Export settings to Messag32.ini (v6.XX)` all corresponding entries in message32.ini are overwritten. For this reason the INI file might contain entries which cannot be carried out in the present module message control or which disable configurations.

4. Requirements

The requirements for using Message Control depend on the desired transmission media:

Medium	Requirement
E-mail message via Outlook (on page 16):	Running Microsoft Outlook instance
E-mail message via SMTP (on page 17):	The Runtime server executing the sending must be connected to a network. An e-mail account must be available. The servers with access to the account must be available.
SMS via GSM (on page 19):	A GSM modem must be connected to the computer and must be approachable via a serial port (COM port). Usually USB modems create a virtual COM port during the installation through which they can be approached.
SMS message via SMS gateway (on page 23):	SMS server of DIALOGS Software GmbH
Speech output (on page 26):	Voice modem with DTMF functionality. Text-to-speech engine. (Partially contained in operating systems. Extendable by a separate text-to-speech package from COPA-DATA.)
Telephone (on page 25):	Voice modem with DTMF functionality. If an ISDN adapter is used it must support the following functions. ▶ Support of voice messages (LINEMEDIAMODE_AUTOMATEDVOICE) ▶ Support of DTMF signals (for receiving the PIN code when acknowledging a message) ▸ Support of Telephony Application Programming Interface (TAPI)

LIMITATIONS

- This module is not available under Windows CE.

5. Supported AT commands

Message Control supports the following AT commands:

Goal	Command	Answer	Comment on the response.
Connection test:	`AT\r`	`\r\nOK\r\n`	at the end

Query PIN status	AT+CPIN?\r	+CPIN:	At a desired point in the response plus \r\n at the end
Enter PIN:	AT+CPIN="[PIN-Code]"\r	\r\nOK\r\n	at the end
Switch modem to SMS-PDU mode:	AT+CMGF=0\r	\r\nOK\r\n	at the end
Set the telephone number of the SMS message center:	AT+CSCA="[SMSC telephone number]"\r	\r\nOK\r\n	at the end
Check to see if the modem has switched to SMS-PDU mode:	AT+CMGF?\r	\r\nOK\r\n	at the end
Check to see if the modem has set the given SMS message center:	AT+CSCA?\r	\r\nOK\r\n	at the end
Send message to this number. Modem signalizes Ready for PDU if the command was accepted:	AT+CMGS="[PDU-length]"\r	\r\n>[space]	
PDU for the SMS to be sent to the number specified previously with completion flag:	[SMS-PDU][0x1A]	\r\nOK\r\n	at the end
List SMSs from the memory with status set by flag:	AT+CMGL=[Flag]\r	\r\nOK\r\n	at the end
Delete all SMSs with status set by flag from the memory (regardless of index):	AT+CMGD=0,[Flag]\r	\r\nOK\r\n	at the end
Delete SMS with the given index from the memory:	AT+CMGD=[Index]\r	\r\nOK\r\n	at the end

Note: For AT+CMGD commands, it is sufficient if one of the two versions is supported.

Arguments:

- \r --> Carriage Return --> 0x0D
- \n --> Line Feed --> 0x0A

ERROR DETECTION

Error messages contain an `ERROR` in the response. You can find details on troubleshooting and testing the connection in the Messages and error handling (on page 90) chapter.

6. Configure Message Control

Message Control is configured via:

- global properties (on page 15) of the media for sending
- project-specific properties (on page 34) for the message
- Functions (on page 44) which are executed in Runtime for sending and management of messages

Messages can be sent and acknowledged as:

- E-Mail
- SMS
- Voice message

CONFIGURING THE SENDING OF A MESSAGE

In order to configure the sending of a message:

1. Select the medium for sending
2. activate the property `Sending mode active` for this medium
3. configure (on page 15) the medium
4. configure (on page 34) the project-specific properties
5. configure the user (on page 40) or user groups to be reached via the medium
6. create a function Send message (on page 54) and

link the function:

a) with a button or

b) with the property `Function` in case of limit values which are supposed to send this message in case of violations

CONFIGURATION OF THE MESSAGE

A message is composed of several parts:

- Subject: defined for e-mails using:
  - `Subject (ID)` in the global settings for Outlook
  - `Subject for outgoing e-mails` in the global settings for e--mail message via SMTP
  - `Dynamic part of the subject` property in the project-specific settings for e--mail via SMTP or Outlook
- Welcome and misentry: is defined for language messages using:
  - Voice message text-to-speech via modem (on page 69)

  or
  - Voice message audiofile via modem (on page 66)
- Message text: defined for e-mails, SMS and voice messages in:
  - the Option (on page 58) `Constant Text`

CONFIGURE THE ADMINISTRATION OF THE MESSAGE QUEUE

In order to administrate messages in the Runtime:

1. create a screen of type Message Control (on page 37)
2. create and configure a function Screen switch (on page 44) for this screen.
3. link the function to a button

By doing so in Runtime messages can be checked for their status and the message queue can be supervised.

MESSAGE CONTROL IN RUNTIME

In order to use Message Control in Runtime:

1. activate Message Control via

- `Active during Runtime start` property Once this property is activated Message Control will automatically start in Runtime.
- Function Send Message: activate (on page 73): If this function is carried out Message Control is started. Messages are only processed from this point of time on.

During normal closing of Runtime an image of the current message queue is saved. This image can also be created anytime in Runtime using the Save current queue (on page 48) function. Message Control can be closed with the function Send Message: deactivate (on page 73) in Runtime.

SUPPRESS MESSAGE

Message Control has a list of deactivated elements. Deactivated means that their alarms do not trigger a message.

If during a violation of limit values the function **`Send Message`** (on page 54) is linked the messages are only created and sent if neither variable nor alarm are suppressed.

- The variable is suppressed if its area or a linked equipment group is contained in the list of suppressed elements.
- The alarm is suppressed if the alarm/event group, the alarm/event class or the alarm area is contained in the list of suppressed elements.

This list can be modified with the function Suppress groups/classes/areas/equipment (on page 49) in Runtime. Alarms and equipment can be activated or deactivated. Suppressed entries are logged in the CEL provided for the property `Log in CEL` `all confirmations` or `only negative confirmations` was selected.

Activating or deactivating entries is carried out remanently in the network. If for instance for maintentance purposes the alarms of an equipment section are switched off this setting will remain active even after closing Runtime and will be re-applied during a restart.

6.1 General settings for sending

The media used employed for sending the messages are globally defined and can be used by every project and every working area.

Available media for sending include the following:

- E-mail message via Outlook (on page 16)

- E-mail message via SMTP (on page 17)
- SMS message via GSM modem (on page 19)
- SMS message via SMS gateway (on page 23)
- Voice message audiofile via modem (on page 25)
- Voice message text-to-speech via modem (on page 26)

In order to configure the properties:

1. highlight the working area
2. select the `Message Control` group in properties
3. activate and configure the desired media types

The settings in **`zenon6.ini`** are saved in the area [Message Control] (on page 28).

6.1.1 E-mail message via Outlook

In order to send a message as e-mail via Microsoft Outlook it is required that:

- the sending mode was activated via the property `Sending mode active`
- on the executing computer a running Microsoft Outlook instance with a connection to the network/internet is available

If messages must be acknowledged it has to be taken care that the mailbox is retrieved within the timeframe available (on page 71)for acknowledgement. This period of time is defined with the function **`Send message`** (tab acknowledgement (on page 71)).

Attention

For Outlook, the following applies for use with message control:

- 32-bit Outlook requires the 32-bit version of zenon.
- 64-bit Outlook requires the 64-bit version of zenon.

Mixed operation is not possible in any combination.

CONFIGURATION

1. Activate the sending mode via the property `Sending mode active`.
2. Define the name of the Outlook profile to be used via the property `Profile`.
3. Define an unambiguous subject via the property `Subject (ID)`.
   This subject is the criterion for passing on incoming messages to the module Message Control. E-mails which do not contain this ID are not passed on to the Message Control. It has to:
   - have a length of at least four characters
   - remain unchanged in case of answers
4. Configure (on page 34) the properties for `Project-specific settings`.
5. Configure Users (on page 40) and user groups.
6. Create required functions (on page 44) and connect them with buttons or events.

BEHAVIOUR IN RUNTIME

1. As soon as the Runtime is started messages are retrieved.
2. Messages are composed according to the settings and are sent as soon as a corresponding function is carried out.
3. If a user responds to a message their response is read when the messages are retrieved and is then passed on to Message Control.
4. Receipt of the message is acknowledged.

6.1.2 E-mail message via SMTP

In order to send a message as e-mail via an SMTP server it is required that:

- the sending mode was activated via the property `Sending mode active`
- the executing computer is connected with the mail server
- an e-mail account is defined

- the mail server is configured with the access data

If messages must be acknowledged it has to be taken care that the mailbox is retrieved within the timeframe available (on page 71)for acknowledgement. This period of time is defined with the function **`Send message`** (tab acknowledgement (on page 71)).

CONFIGURATION

1. Activate the sending mode via the property `Sending mode active`.
2. Configure the properties for:
   - `Incoming mail server`
   - `Outgoing mail server`
3. Configure (on page 34) the properties for project-specific settings.
4. Configure Users (on page 40) and user groups.
5. Create required functions (on page 44) and connect them with buttons or events.

BEHAVIOUR IN RUNTIME

1. As soon as the Runtime is started messages are retrieved.
2. Messages are composed according to the settings and are sent as soon as a corresponding function is carried out.
3. If a user responds to a message their response is read when the messages are retrieved and is then passed on to Message Control.
4. Receipt of the message is acknowledged.

Only one thread at a time can open connections. A connection is always opened for the thread requiring it and is closed immediately after exchanging data. While a message is being sent no other thread can retrieve the incoming messages or send messages at the same time. If the incoming messages are retrieved it is also not possible for another thread to start a retrieval or send messages.

If during this procedure an error occurs, such as authentication failed or command not supported, all active connections are closed.

The correct settings for the configuration are available from the provider of the employed e-mail account.

PROCEDURE

RETRIEVING INCOMING E-MAILS

1. Establishing a connection with the parameterized incoming mail server and the corresponding port.
   If the connection is or is not secure depends on the settings of property `Encryption`.
2. All mails in the mail inbox are retrieved by the commands STAT and RETR. It is decided on the basis of the subject whether an e--mail is relevant. Not relevant messages are skipped. Relevant messages are forwarded to Message Control and - as long as the property `Leave read e-mail on server` is deactivated - deleted from the server with the command DELE.
3. The connection is closed by sending the QUIT command and subsequent closing of the connection.

SENDING OF AN E-MAIL

1. According to the setting a secure or unsecured connection to the SMTP server is established.
2. If configured the SMTP authentication is carried out with the AUTH command.
3. The message to be sent is transferred in MIME format with the SMTP commands MAIL, RCPT and DATA.
4. The SMTP connection is closed by sending the QUIT command and subsequent closing of the connection.
5. If there is a POP3 connection this is also closed.

6.1.3 SMS message via GSM modem

For sending a message as SMS via a GSM modem:

- the sending mode must have been activated via the property `Sending mode active`
- a GSM modem must be connected to the sending computer and must be approachable via a serial port (COM port)

Note: usually USB modems create a virtual COM port during the installation through which they can be approached.

- The GSM modem probably has to be initialized with a manufacturer software in order to be able to connect to the GSM network.

CONFIGURATION

1. Activate the sending mode via the property `Sending mode active`.
2. Select the COM port of the modem from the drop-down menu via the property`Modem connection (serial)`.
   The correct port is displayed in the system properties of the modem.
   Note: After a reconnection of the modem or a reboot of the computer the modem probably will have to be re-initialized with the manufacturer software before it can be used with Message Control. The COM port may change with a reconnection of the modem or a reboot of the computer.
3. In the property `PIN code` enter the PIN for the modem, if required.
4. Check the connection by clicking `Check connection`.
5. Enter the telephone number for the short message center in the property field `Number of SMS center`.
6. Configure (on page 34) the properties for `Project-specific settings`.
7. Configure Users (on page 40) and user groups.
8. Create required functions (on page 44) and connect them with buttons or events.

BEHAVIOUR IN RUNTIME

1. As soon as the Runtime is started the connection to the modem is being established and configured.
2. Messages are being retrieved.
3. Messages are composed according to the settings and are sent as soon as a corresponding function is carried out.

4. If a user responds to a message their response is read when the messages are retrieved and is then passed on to message control.
5. Receipt of the message is acknowledged.
6. If after retrieving no incomplete multi-part messages are present anymore the messages which were read are deleted at the modem.
7. During the closing of Runtime the modem is disconnected.

Information

If the modem is reconnected or the computer rebooted it is recommended to:

- execute the configuration software delivered by the GSM operator so that the modem is initialized and connects with the network.
- check in the system control if the modem uses the same COM port again after reconnecting.

CONSTANTS, TIMEOUTS AND PARAMETERS

Entry	Value	Description
Timeout for AT commands	10000 ms	The modem has 10 seconds to respond to the transmission of an AT command.
SMS polling interval	20000 ms	Every 20 seconds it is checked if new SMS have arrived and they are being read. If no incomplete multi-part SMS are present anymore all messages which were read are deleted.
COM timeouts	Write: 5 seconds read: immediate return	5 seconds are available for writing the bytes on the COM port (max. 400 in case of SMS-PDU with maximum size). During the reading of the COM port the current content of the receive buffer is read and immediately returned.
COM state configuration	▸ Baud: 9600 ▸ Binary mode ▸ Parity check: inactive ▸ Clear-to-Send and Data-Set-Ready: not controlled ▸ Data-Terminal-Ready and Request-To-Send Flow Control: activated ▸ Bit per byte: 8 ▸ Parity: None ▸ Stop Bit:1	Standard configuration of a serial port.

BUFFER

Limit	Maximum value	Description
Size of send buffer and size of receive buffer for the serial port in byte	`4096`	The individual PDUs have a maximum size of 400 byte. During the sending of an SMS the individual PDUs are sent one after another. During receiving all SMS arrived since the last `polling interval` are retrieved at once as block. Thus the modem can move at least 10 PDUs at once into the receive buffer of the serial port.
Size of the total buffer for conversion and incoming SMS in byte	`12288`	Since an SMS-PDU occupies a maximum of 400 byte in the receive buffer within one `polling interval` of 20 seconds a maximum of 30 SMS parts with maximum size can be received with this buffer. The maximum size of an SMS to be sent is 6144 Unicode characters.

6.1.4 SMS message via SMS gateway

For sending a message as SMS via an SMS gateway:

- the sending mode must have been activated via the property `Sending mode active`
- an SMS server of the company DIALOGS GmbH must be present and configured.

PROCEDURE

SMS SERVER CONFIGURATION

For the correct configuration of the SMS servers, please contact the manufacturer, DIALOGS GmbH.

The following is applicable in order for it to work with zenon:

- The following folders must be present:
  - Inbox
  - Outbox
  - State

- zenon moves messages to these folders.
- The SMS gateway reads and fills these folders.
- zenon monitors these folders:
  As soon as new messages or information is in them, these are forwarded to Message Control.

CONFIGURATION IN ZENON

1. Activate the sending mode via the property `Sending mode active`.
2. Configure `Outbox folder`, `Inbox folder` and `State folder`.
3. Configure the `First letter of the SMS file`.
   This initial letter is unique in the project. If an SMS gateway is used by various projects, for each project a unique prefix must be defined.
4. configure `First letter of the lock/semaphore file`: Sets initial letter for the lock/semaphore files. This starting letter is unique in the project. If an SMS gateway is used by various projects, for each project a unique prefix has to be defined.
5. Configure the property `Smart alarming`.
   If the property is activated the SMS are treated as follows:
   - all outgoing messages are being checked. If the message starts with the text **`ALARM!`**, the message is not sent in the usual format (**`ID; message`**) but in the format: **`ALARM! ID; message`**.
   - Along with the sending of a message starting with **`ALARM!`** the recipient's mobile phone is automatically switched to "`very loud`", provided this is supported by the phone.
   - Incoming messages are also checked for this character string.
     If there is a (**`!`**) in position 6, the first six characters are cut off and the message is forwarded by Message Control.
6. Configure (on page 34) the properties for `Project-specific settings`.
7. Configure Users (on page 40) and user groups.
8. Create required functions (on page 44) and connect them with buttons or events.

BEHAVIOUR IN RUNTIME

1. As soon as the Runtime is started messages are retrieved.
2. Messages are composed according to the settings and are sent as soon as a corresponding function is carried out.
3. If a user responds to a message their response is read when the messages are retrieved and is then passed on to Message Control.
4. Receipt of the message is acknowledged.

6.1.5 Voice message audiofile via modem

For sending a voice message as audiofile:

- the sending mode must have been activated via the property `Sending mode active`
- one audiofile respectively has to be deposited in the node Files/Multimedia audiofiles for:
  - Greeting
  - Message
  - Failure entry
- a modem must be connected to the sending computer and must be configured
- The telephone of the recipient must support DTMF

PROCEDURE

CONFIGURATION

1. Activate the sending mode via the property `Sending mode active`.
2. Configure the settings for `Time-out` and `Repeat welcome text`.
3. Select the modem for sending via property `Line name`.
   Selection from a drop-down menu which lists all modems configured in the system control.

4. Configure (on page 34) the properties for `Project-specific settings`.
5. Configure Users (on page 40) and user groups.
6. Create required functions (on page 44) and connect them with buttons or events.
7. This method must be acknowledged.

BEHAVIOUR IN RUNTIME

1. As soon as a function activates the sending mode a call is initiated via the modem.
2. As soon as the call is answered the greeting and message are replayed.
3. The message must be acknowledged (on page 85) by the recipient with the respectively valid code:
   - `PIN code` for confirmation
   - `NA code` for rejection

In addition, the recipient can:

- replay the file just listened to by pressing the hash key (#) on the phone
- undo the entry by pressing the star key (*); in this case the greeting is being replayed

6.1.6 Voice message text-to-speech via modem

For sending a voice message via text-to-speech:

- the sending mode must have been activated via the property `Sending mode active`
- a modem must be connected to the sending computer and must be configured
- The telephone of the recipient must support DTMF
- A text-to-speech (TTS) engine must be installed.
- The corresponding language file must be installed

LANGUAGE FILES

Depending on the version of the Editor, (32-bit or 64-bit), the corresponding speech (32-bit-compatible or 64-bit-compatible) must be installed and configured.

To do this:

1. Close the Editor.
2. Select the correct speech for the Editor (**`sapi.cpl`**):
   - 32-bit-compatible speech for the 32-bit Editor:
     `C:\Windows\SysWOW64\Speech\SpeechUX\sapi.cpl`
   - 64-bit-compatible speech for the 64-bit Editor:
     `C:\Windows\System32\Speech\SpeechUX\sapi.cpl`
3. Start the Editor and select the configured speech in the `Voice` property.

PROCEDURE

CONFIGURATION

1. Activate the sending mode in the `Voice message (Text to Speech)` section via the `Sending mode active` property.
2. Configure the settings for `Time-out` and `Repeat welcome text` in the `Voice message (Audio file)` section
3. Select the modem for sending in the `Voice message (Audio file)` section using the `Line name` property.
   Selection from a drop-down menu which lists all modems configured in the system control.
4. Define language and voice in the `Voice message (Text to Speech)` using the `Voice` property.
   Note: The COPA-DATA sales partner will provide you with additional voices and languages.
5. Configure (on page 34) the properties for `Project-specific settings`.
6. Configure Users (on page 40) and user groups.
7. Create required functions (on page 44) and connect them with buttons or events.

8. This method must be acknowledged.

BEHAVIOUR IN RUNTIME

1. As soon as a function activates the sending mode a call is initiated via the modem.
2. As soon as the call is answered the greeting and message are replayed.
3. The message must be acknowledged or rejected by the recipient with the respectively valid code.
   - `PIN code` for confirmation
   - `NA code` for rejection

In addition, the recipient can:

- replay the file just listened to by pressing the hash key (#) on the phone
- undo the entry by pressing the star key (*); in this case the greeting is being replayed

> **Information**
>
> Depending on the operating system and the software installed voices and languages are already provided which zenon can access. The COPA-DATA sales partner will provide you with further voices and languages. Currently available are: German, English (US, UK, Indian), French and Spanish.

6.1.7 zenon6.ini entries

Message Control is mainly configured via global (on page 15) and project-specific (on page 34) properties. In the **`zenon6.ini`** the global properties of the media are displayed in section **`[Message Control]`**:

Entry	Description
[MESSAGE CONTROL]	Section in **zenon6.ini**
Outlook_Profile=	Name of the Outlook profile which should be used for sending. Default: empty
Outlook=	Sending active? 0: inactive 1: active Default: 0
Subject=	Unique ID which incoming e-mails must contain in order to be processed in Message Control. Default: empty
Telephone_Timeout=	Time in minutes after which a standing condition should be canceled and closed. Time interval must be longer as the time needed for playing back and confirming the message. Default: 1
Telephone_WelcomeMessageCount=	Number of repetitions for the welcome text. Default: 5
Telephone_IgnoreDisconnect=	1: A disconnection (e.g. recipient ends call) is ignored and the message is played back completely before the line is closed. 0: Message is aborted when the connection breaks. Default: 0
Telephone=	Sending active? 0: inactive 1: active Default: 0
Speech=	Sending active? 0: inactive 1: active Default: 0

Speech_Rate=	Speech speed. Minimum: -10 Maximum: 10 Default: 0
Speech_Volume=	Speech volume. Number equals the percent value of the maximum value of the selected speech. Maximum: 100 Minimum: 1 0: Number of the operating system is take over without change. Default: 0
Speech_Name=	Selection of speech and language Default: empty
SMSGateway=	Sending active? 0: inactive 1: active Default: 0
SMSGateway_Inbox=	Income folder for sending SMS. Default: empty
SMSGateway_Outbox=	Outgoing folder for sending SMS. Default: empty
SMSGateway_Statusbox=	Folder for the status message at SMS sending. Default: empty
SMSGateway_Prefix=	First letter of the SMS files. Length: 1 characters Default: F
SMSGateway_SemaphorPrefix=	First letter of the lock file. Length: 1 characters

	Default: S
SMSGateway_OriginId=	Sender identification for sending SMS. Default: empty
SMSGateway_TimeOut=	Timeout for outgoing messages in minutes. Defines after what period of time a message is interpreted as "not send successfully". Default: 60
SMSGateway_SmartAlarm=	Smart alarming is used. Default: 0
SMTP_POP_MAIL=	The entry defines whether the sending type is active. 1: active 0: inactive Default: 0
SMTP_SRV_IS_POP=	This entry defines whether the POP3 Server is used as SMTP Server. 1: POP 3 is SMTP 0: POP3 and SMTP are different Servers Default: 0
SMTP_SERVER=	The SMTP Server entered by the user. Default: empty
SMTP_PORT=	Defines the used port at the SMTP Server. Default: 25
SMTP_SECURITY=	Type of connection protection to the SMTP Server. 0, no security 1: SSLv2 and SSLv3 2: TLSv1 Default: 0
SMTP_AUTH=	Authentication at the outgoing server.

	0, no security 1: log in to the POP3 Server before sending 2: SMTP AUTH Default: 0
SMTP_OTHER_CREDS=	Defines whether the outgoing server uses different log in data than the incoming server. 0: inactive 1: active Default: 0
SMTP_USER=	User name saved for the outgoing server. Default: empty
SMTP_PASSWORD=	Hex dump of the encrypted password for authentication at the outgoing server. Default: empty
SMTP_SUBJECT=	Subject for outgoing e-mails and for detecting whether an incoming e-mail at the server is relevant for the sending type. Default: MsgCtrl_Alert
SMTP_USER_IS_ADDR=	Defines whether the user name for authentication towards the outgoing server is used as sender address for outgoing mails. 1: active 0: inactive Default: 0
SMTP_OUT_ADDR=	Address for outgoing e-mails. Default: empty
POP_SERVER=	POP3 Server saved. Default: empty
POP_PORT=	Defines the used port at the POP3 Server. Default: 110
POP_SECURITY=	Type of connection protection to the POP3 Server.

	0, no security 1: SSLv2 and SSLv3 2: TLSv1 Default: 0
POP_APOP=	Defines whether the APOP command should be used. 1: active 0: inactive Default: 0
POP_KEEP_MAILS=	Defines whether read e-mails remain on the server. 1: E-mails are not deleted after they were fetched from the server 0: E-mails are deleted after they were fetched from the server Default: 0
POP_USER=	User name for the incoming server. Default: empty
POP_PASSWORD=	Hex dump of the encrypted password for authentication at the incoming server. Default: empty
POP_POLL_INTERVALL=	Minimum period between two POP3 requests in seconds. Minimal: 10 Default: 60
GSM_SMS=	Activation of SMS via GMS as sending type. 0: active not 0: inactive Default: 0 The entry equals entry **[GSM]** à On in file Message32.ini and is considered at the import/export of the INI setting.
GSM_SMS_COM=	COM port which is used for the connecton to the modem.

	Default: empty
`GSM_SMS_SMSC=`	Telephone number of the message center of the GSM provider. Default: empty
`GSM_SMS_PIN=`	PIN code which is used for authentication towards the modem. Default: empty

Information

As some properties can take over the values of other properties and they remember the values entered last, the values in the INI entry must not always concur with the values of the properties displayed in the Editor. The following entries are concerned if they are displayed as not available:

SMTP_SERVER -> `Server address`

SMTP_USER -> `User identification`

SMTP_PASSWORD -> `Password`

SMTP_OUT_ADDR -> `Address for outward e-mails`

6.2 Project-specific settings

You can find the project-specific settings in the node Message Control in the corresponding project. The properties for configuration of the message and its delivery are contained there. General properties (on page 15)for sending media are configured in the properties `Message Control` of the working area.

ACTIVATING MESSAGE CONTROL FOR THE PROJECT

In Runtime, Message Control can be activated as follows:

- `Active during Runtime start` property: Once this property is activated Message Control will automatically start in Runtime.

- Function **`Send Message: activate`** (on page 73): If this function is carried out Message Control is started. Messages are only processed from this point of time on.

Message Control can be closed with the function **`Send Message: deactivate`** (on page 73) in Runtime.

PROJECT-SPECIFIC PROPERTIES

The following can be set in the project-specific properties:

- Dynamic part of the subject via the property `Dynamic part of the subject`: Incoming messages are filtered for this content for forwarding to Message Control.
- Logging the message via property `Log in CEL`: In the Chronologic Event List messages can be logged according to different criteria:
  - all messages
  - None
  - only messages which have not been rejected or responded to
  - only messages which have been acknowledged
- Marking the message status by means of color and graphics: The column **`status`** of the screen Message Control can display the status of a message by means of a graphic and color.
- Rules for the repetition of interrupted calls

> **Information**
>
> Rules for interrupted calls:
>
> - If a call is interrupted without confirmation or a decline and the interruption is not classified as a hardware error, then:
> - The call is not repeated for the same addressee
> - If substitutes or group members who have not yet been messaged are messaged
> - If a call is interrupted by a hardware error, then the call for the same addressee is sent again according to the number of repetitions set in the `Max. number of attempts` property.

CONFIGURATION OF THE MESSAGE

It comprises:

1. Static part: unalterable subject which is defined in the global settings (on page 15) for e-mails.
2. Dynamic part: project-specific part of the subject. Is defined with the property `Dynamic part of the subject`.

STRUCTURE OF THE DYNAMIC PART OF THE SUBJECT:

- Text parts are separated from one another with a semi colon (`;`).
- **`$`**: marks text that contains the parameter.
- A parameter (on page 74) can:
  - Be a key from the language table, such as **`@MyText;`**
  - Relate to a variable, when executing the function via AML or a limit value breach such as **`%Var1;`**
  - Be a compiled entry in the language table: for example **`%@Var2+MultipleText`**
- `@` marks language switching
- `%` marks variables
- `%%` marks **`limit value text parameters`** (on page 75) for variables

A variable can be stated between the two percentage marks.
If no variable is given, the value relates to the main variable.

- Messages end with a semicolon (`;`).

Parameters	In the Runtime	Action
key: **`@MyText`**	A check is made to see if an entry in @MyText is present in the language table for the current language.	▸ `Existing`: entry is added to the message. ▸ `Not existing`: According to the settings of the project properties `Display not translated keywords` either **`@MyText`** or **`MyText`** is added to the message.

Variable: **%Var1%**	A check is made if the variable exists and the value can be read.	▸ `Existing`: Value of the variable is taken and added to the text as string. ▸ `Not existing/not readable`: The text **xxx** is added to the message.
Compound entry: **@StringTable+%var1Text**	A check is made if: ▸ 1. the variable exists and the value can be read. Value is attached to the prefix text ("MultipleText") as string. ▸ 2. an entry in the language table exists for the text	▸ Variable and text exist: text is added to the message. Example: Value of the variable is `33`. The language table is then checked for **@StringTable33** and the corresponding text is added. ▸ `Not existing`: According to the settings of the project properties `Display not translated keywords` either **@MyText** or **MyText** is added to the message.

Example: **$@Attention;%Var1;**

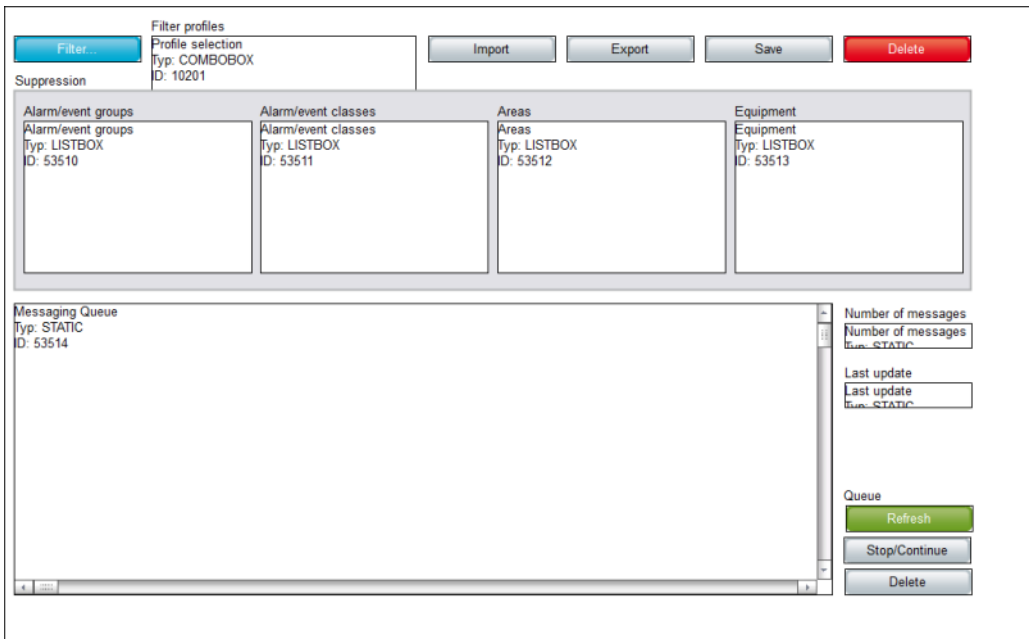
6.3 Create a screen of type Message Control

With a screen of type `Message Control` you manage the message queue of a project in Runtime.

To create a screen of type `Message Control`:

1. create a new screen
2. select`Message Control` in the drop-down menu for the screen type
3. Click in the screen.
4. select the control elements menu ->Insert template
5. the control elements are added to the screen

6. create a function for the screen switching (on page 44) and connect this with a button in order to be able to connect the screen to Runtime



Control elements	Description
Insert template	Opens the dialog for selecting a template for the screen type. Templates are shipped together with zenon and can also be created by the user. Templates add pre-defined control elements to pre-defined locations in the screen. Elements that are not necessary can also be removed individually once they have been created. Additional elements are selected from the drop-down list and palced in the screen. Elements can be moved in the screen and placed individually.
Buttons	Buttons for Runtime actions.
Refresh	Refreshes view. The current data are loaded onto the screen from the message queue. The point of time of the last update can be displayed with the field **Last update**.
Clear	Deletes selected entry from the list. It is only possible to delete messages which have not been sent yet.
Filter	Opens Dialog (on page 45) to configure the filters for the view.
Stop/continue	Starts and stops the cyclic update of the list.
Windows	Lists and fields which can be displayed in Runtime.
Number of messages	Number of currently existing messages.
Last Update	Point of time of last update.
Suppressed alarm/event groups	**Alarm/event groups** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Suppressed alarm/event classes	**Alarm/event classes** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Suppressed alarm/event areas	**Alarm/event areas** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Suppressed equipment	**Equipment** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.

Control elements	Description
`Message waiting list`	List of messages. Contains messages which have been sent, confirmed or deleted or remain to be sent. The messages are only displayed and cannot be edited anymore. Column headings can be named individually (on page 47) and are localizable by putting `@` before them.
Filter profiles	Profile administration
`Profile selection`	Opens the dialog for selecting a profile.
`Save`	Saves current setting as a profile.
`Clear`	Deletes profile.
`Import`	Opens dialog for importing profiles from a file.
`Export`	Opens dialog for exporting profiles from a file.

Information

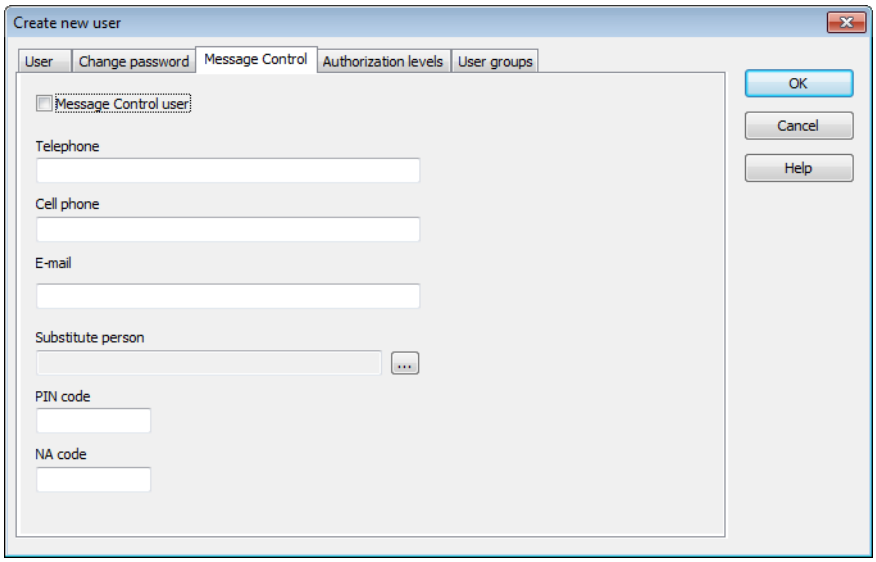
The columns of the list can be confirgured via the Filter (on page 45) for the function screen switching in the Editor, in Runtime via screen switching -> show dialog in the RT or the button ***`Filter`****. Columns of the list can be provided with individual labels and are localizable.*

6.4 User administration

Users and user groups for Message Control are created and managed in the zenonuser administration. In the user administration the property `Message Control user` or respectively `Message Control group` must be active in order to use a user in Message Control.

Users and user groups are referenced by names.

USERS


Create new user
User
Change password
Message Control
Authorization levels
User groups
Message Control user
Telephone
Cell phone
E-mail
Substitute person
PIN code
NA code
OK
Cancel
Help

Parameters	Description
Message Control User	Active: The user is used by the module Message Control.
Telephone	Fixed network telephone number of the user. Used for text to speech. Enter numbers. The prefix + abbreviating 00 of the international area code is permitted.
Cell phone	Mobile phone number of the user. Used for messages via GSM and SMS (text messages). Enter numbers, the prefix + abbreviating 00 of the international area code is permitted.
Email	E-mail address of the user
Substitute person	Select a substitute person if the user cannot be reached or the receipt of the message is rejected. A click on button . . . Opens the dialog to select a user.
PIN code	PIN code with which the user confirms the message.
NA code	PIN code with which the user rejects the receipt of the message (not available). Message is subsequently sent to the next user in the list.
OK	Applies all changes in all tabs and closes the dialog.
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

Attention

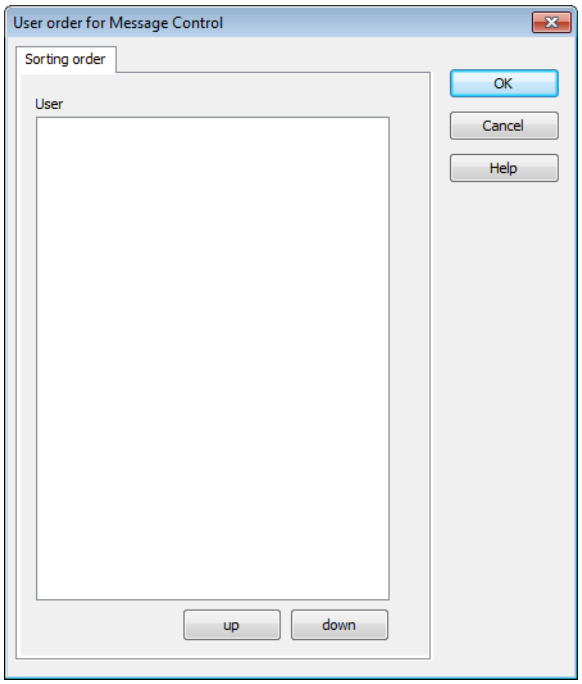
The acknowledgment codes for PIN (confirmation) and NA (rejection) must differ and should not be too similar.

If both codes are identical the code is interpreted as PIN and therefore as confirmation of the message.

If an unknown code is received, an SMS and e--mail is sent to the substitute person. The error message is played back for voice messages.

SEQUENCE WITHIN THE USER GROUP

Users can be sequenced through the property `User order` within a user group.



Parameters	Description
`Users`	List of all available users.
`Up`	Moves selected user up one place.
`Down`	Moves selected user down one place.
`OK`	Applies settings and closes the dialog.
`Cancel`	Discards all changes and closes the dialog.
`Help`	Opens online help.

For Runtime the sequence of the users from the global user administration is added to the defined sequence in this property. As a consequence, local users are always displayed before global users.

Information

Get details on the user administration in zenon in the manual User administration.

6.5 Functions

Functions control the use of Message Control in Runtime.

With this:

- Message control is activated (on page 73) and deactivated (on page 73)
- messages are sent (on page 54)
- the screen of type Message Control is connected (on page 44) for managing the message queue
- the message queue is saved (on page 48)
- elements for messages are disabled (on page 49)

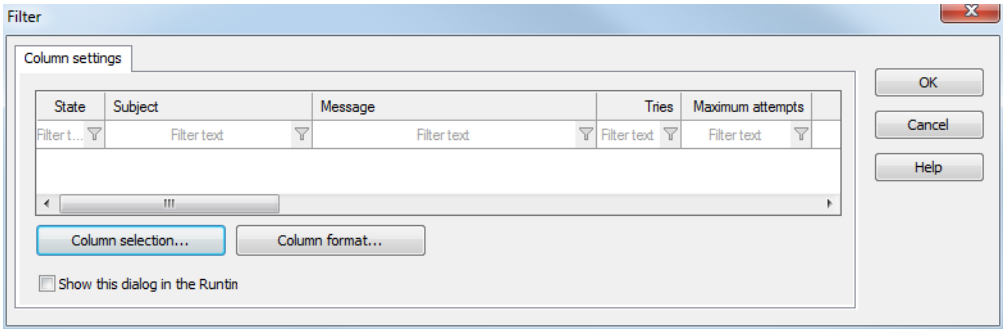
6.5.1 Screen switching to a screen of type Message Control

With a screen switching of type Message Control you access a in for Runtime in order to supervise the message queue. To configure screen switching:

1. create a new function
2. select `Screen switch`
3. the selection dialog for the screen is opened
4. select the desired screen of type `Message Control` (on page 37)
5. the dialog for configuring the column settings (on page 45) is opened
6. configure the columns
7. close the configuration by clicking ok
8. connect the function with a button in order to be able to access it for Runtime

Column settings

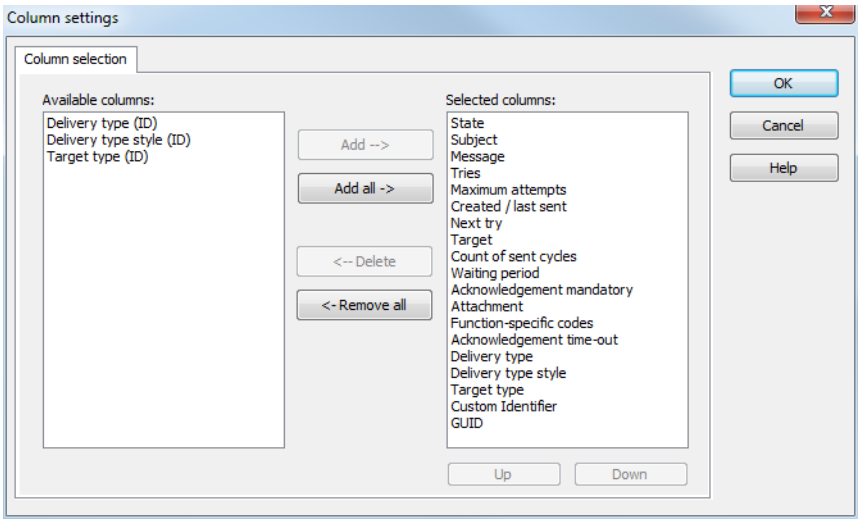
Settings for filtering messages



Parameters	Description
List field	Display of the configured columns.
Column selection	Opens dialog to select the character columns.
Column format	Opens a dialog to format the columns.
OK	Applies all changes and closes dialog.
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

Column selection

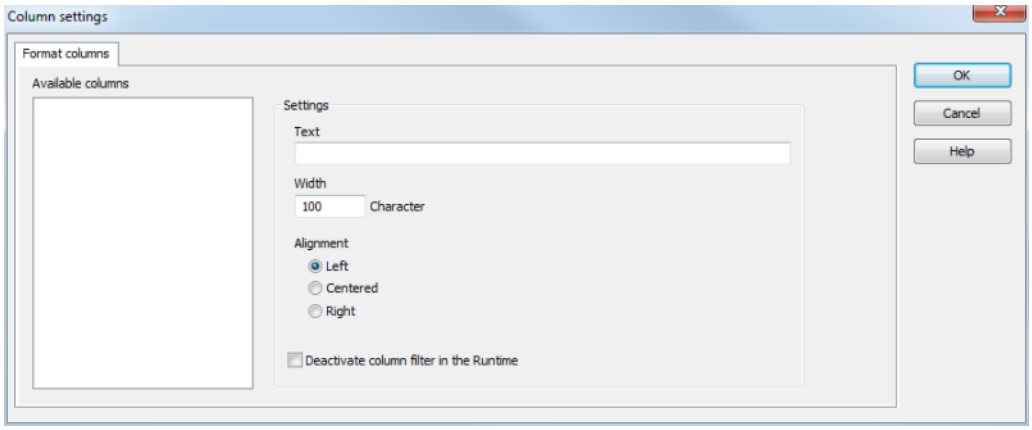
Definition of the columns displayed in the Runtime.



Button	Function
Add	Moves the selected column from the available ones to the selected items. After you confirm the dialog with OK, they are shown in the detail view.
Add all	Moves all available columns to the selected columns.
Remove	Removes the marked columns from the selected items and shows them in the list of available columns. After you confirm the dialog with OK, they are removed from the detail view.
Remove all	All columns are removed from the list of the selected columns.
Up	Moves the selected entry upward. This function is only available for unique entries, multiple selection is not possible.
Down	Moves the selected entry downward. This function is only available for unique entries, multiple selection is not possible.
OK	Applies settings and closes the dialog.
Cancel	Discards settings and closes the dialog.
Help	Opens online help.

Column format

Formatting of columns.



Parameters	Description
Available columns	List of the available columns via **Column selection**. The column selected here is configured using the settings in the **Parameters** section.
Parameters	Settings for selected column.
Labeling	Name for column title. The column title is online language switchable. For this you must enter the @ character in front of the name.
Width	Width of the column in characters. Calculation: Number time average character width of the selected font.
Alignment	Alignment. Possible settings: ▸ Left-justified: Text is justified on the left edge of the column. ▸ Centered: Text is displayed centered in the column. ▸ Right Text is justified on the right edge of the column.
Block column filter in Runtime	Active: The filter for this column cannot be changed in Runtime. **Note:** Only available for: ▸ Batch Control ▸ Extended Trend ▸ Message Control ▸ Recipegroup Manager
OK	Applies settings and closes the dialog.
Cancel	Discards settings and closes the dialog.
Help	Opens online help.

6.5.2 Save current queue

This function saves an image of the current message queue. If the Runtime is closed normally this image will be replaced by a valid image of the queue when closing the Runtime. To engineer the function:

1. create a new function

2. select in the group `Message Control` function **`Save current queue`**
3. connect the function with a button in order to be able to access it for Runtime

This function is always performed at the computer executing the process.

6.5.3 Suppress groups/classes/areas/equipment

Message Control has a list of deactivated elements. Deactivated means that their alarms do not trigger a message. The list can be changed with this function. Alarms and equipment can be activated or deactivated. The projection takes place in the Editor and can be released for changes in Runtime.

If a `Limits` is linked to a function `Send a Message` the messages are only created and sent if neither variable nor alarm are suppressed.

- The variable is suppressed if its area or a linked equipment group is contained in the list of suppressed elements.
- The alarm is suppressed if the alarm/event group, the alarm/event class or the alarm area is contained in the list of suppressed elements.

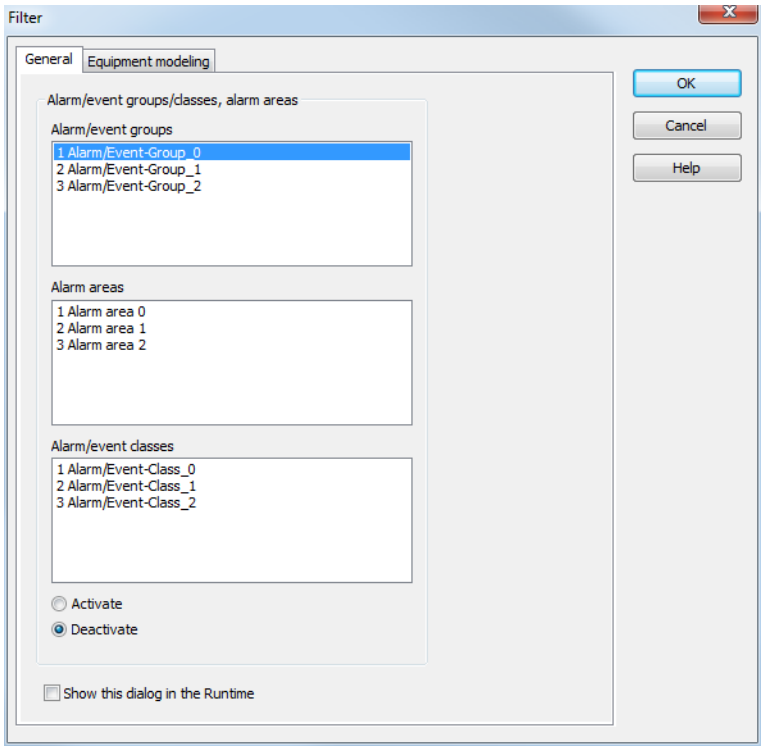
Suppressed entries are logged in the CEL provided for the property `Log in CEL` `all confirmations` or `only negative confirmations` was selected.

CONFIGURING FUNCTIONS

To configure the function:

1. create a new function
2. select in group `Message Control` the function **`Suppress Groups/Classes/Areas/Equipment`**
3. the dialog for configuring the elements to be suppressed or activated is opened.
4. configure the elements
5. Close the dialog by clicking on **OK**
6. connect the function with a button in order to be able to access it for Runtime

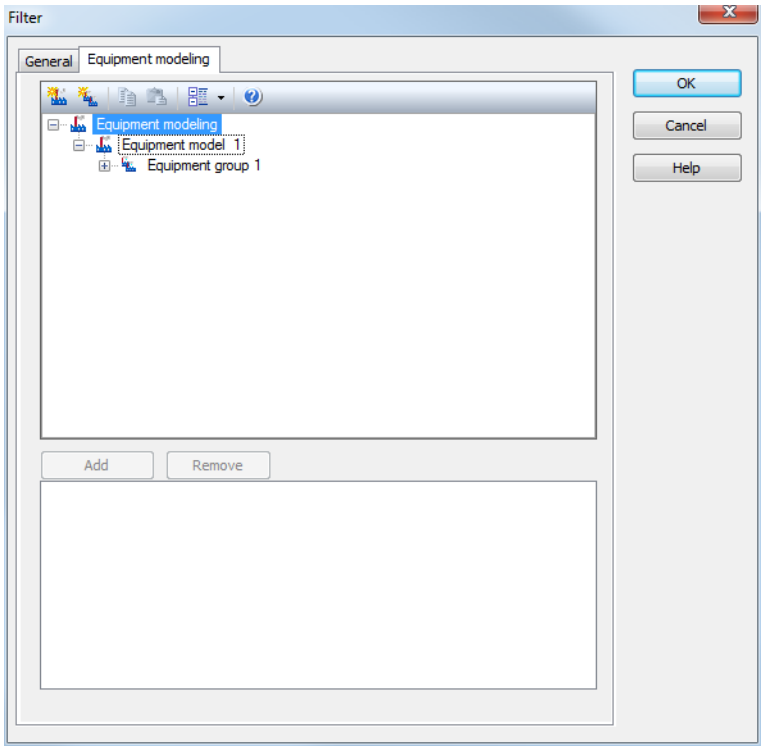
General


Filter
General
Equipment modeling
Alarm/event groups/classes, alarm areas
Alarm/event groups
1 Alarm/Event-Group_0
2 Alarm/Event-Group_1
3 Alarm/Event-Group_2
Alarm areas
1 Alarm area 0
2 Alarm area 1
3 Alarm area 2
Alarm/event classes
1 Alarm/Event-Class_0
2 Alarm/Event-Class_1
3 Alarm/Event-Class_2
Activate
Deactivate
Show this dialog in the Runtime
OK
Cancel
Help

Parameters	Description
Alarm/event groups	List of configured alarm/event groups.
Alarm areas	List of configured alarm areas.
Alarm/event classes	List of configured alarm/event classes.
Activate	Active: When activating the function the selected elements are added to the list of suppressed elements.
Deactivate	Active: When activating the function the selected elements are removed from the list of suppressed elements.
Display this dialog in Runtime	When activating the function this dialog is opened and the user can adjust the configuration before execution. The dialog is displayed on the current computer in Runtime. During network operation when activating the client the dialog is also displayed on the client
OK	Accepts settings in all tabs and closes dialog.
Cancel	Discards settings in all tabs and closes dialog.
Help	Opens online help.

Selecting elements from a list performed via mouseclick. Multiple selection is possible. A simple mouseclick is enough, it is not necessary to press additional keys. Clicking twice on a selected element deselects the element.

Equipment modeling



Property	Description
Tool bar	Symbols to: ▸ Edit local equipment models ▸ Expand or collapse the display ▸ Display of information
List of equipment models	provides models and groups for selection The list separates the display into equipment models from the global project and from local projects. Local equipment models can be created, edited or deleted.
Add	Adds the selected group to the filter list.
Remove	Removes all selected groups from the filter list.
Filter list	Shows all equipment groups that are to be filtered.
OK	Applies settings and closes the dialog.
Cancel	Discards the selection and closes the dialog. **Attention:** Any changes that have been made to the structure of local equipment models are retained.
Help	Opens online help.

ADD GROUPS

- select the desired equipment model
  Attention: If there are naming conflicts between global and local equipment models, the local equipment models are displayed and the global ones are ignored. You can get information on possible conflicts by clicking on the corresponding symbol (triangle with exclamation mark) in the tool bar.
- Select an equipment group or level.
- Add the new group to the list with the **Add** button in the lower area of the dialog
- Subgroups are not automatically added.
- it is possible to link as many groups as you want.

DELETE GROUPS

- Select the desired elements in the list in the lower area of the dialog (multiple selection is possible)
- Click the **`Delete`** button

Note: Changes in a tree element remain preserved independent of clicking button **`Cancel`**. **`Cancel`** only means that no element has been selected.

IN THE RUNTIME

When executing the function the setting configured in the dialog is sent as request to the process-executing computer. It is then analyzed.

6.5.4 Send a Message

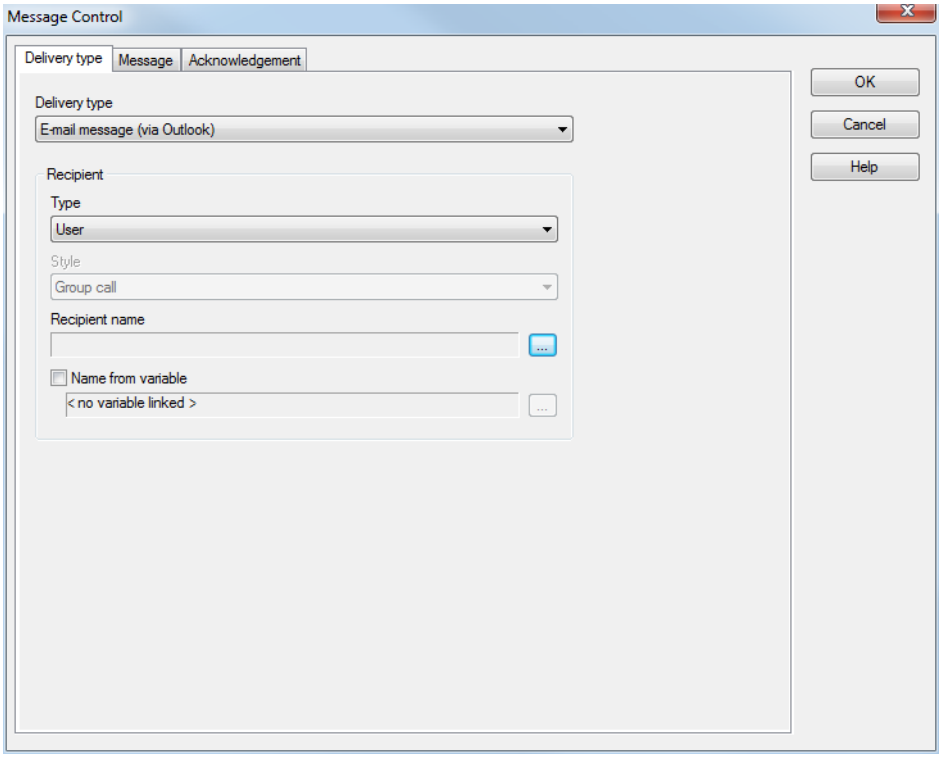
This function allows for a sending of messages in Runtime. To do so, link this function with:

- an alarm (function for group, class or area)
- a limit value (property `Limits/Function`),
- a response matrix (function)
- a time control (via button or PFS)

To configure the function:

1. create a new function
2. in group `Message Control` select the function `Send a Message`

3. The dialog for configuration is opened

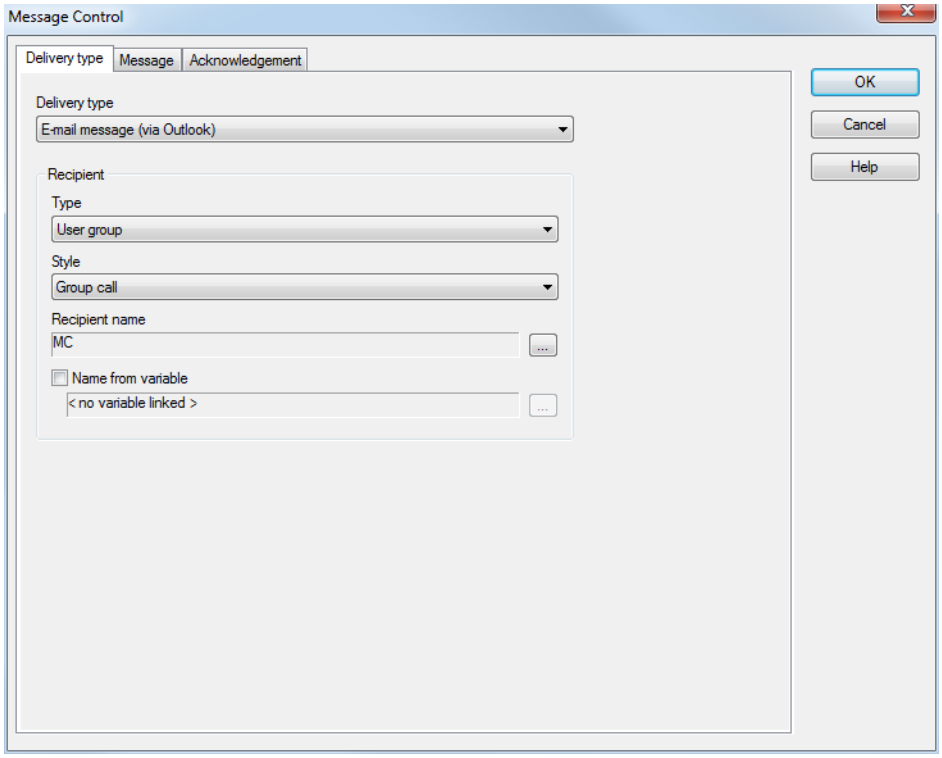


4. configure
   - Sending mode (on page 56)
   - Message (on page 58)
   - Acknowledgement of receipt (on page 71)
5. Close the dialog by clicking on OK
6. link the function

> **Information**
>
> In Runtime the function `Send a Message` is always performed at the computer executing the process.

Sending mode

In this tab the sending mode, recipient and method are configured.



Parameters	Description
Sending mode	Selection of the sending mode from the drop-down list: In this list only media are shown whose property Sending mode active had been activated in the Message Control Properties (on page 15) of the working area.
Recipient	Recipient settings:
Type	Selection of recipient pattern from the drop-down list. Possible selection: ▸ User group: Sending to a group. Selection of a group in case of switching the option or via option recipient name. The sending mode for the group is defined in the option Method. ▸ User: Sending to an individual user (and the substitute person defined in their profile). Selection of a user in case of switching the option or via option recipient name.
Method	Only available if the option Type user group was selected. Selection of sending method from drop-down list. Possible methods: ▸ Group call: All users of the group will receive the message parallelly. (Note: in previous versions to zenon 7.00 this method corresponded the sending type **Group sending**.) ▸ Successive call: The message is delivered to the users according to the defined sequence (on page 40) group successively until the first user positively acknowledges receipt. The group is only run through once. If there is no positive acknowledgement the sending is ended. ▸ Infinite successive call: In principle corresponds to the Successive call, however, includes a repetition: After the last recipient the group is being run through again beginning on the top. The message is being sent until it is positively acknowledged by the first recipient.
Recipient name	Only available if the option Name from variable is deactivated. Opens dialog for selection of a recipient or a recipient group - depending on the selection in the option Type.
Name from variable	Active: The variable is read in Runtime and a user or user group from the user administration is read out through its value. Click on button . . . in order to open the dialog for selecting a variable.
OK	Applies all changes in all tabs and closes the dialog.

	Dialog can only be closed if all required options have been configured.
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

Message

In this tab the content of the message is defined. Available options and configuration depend on the selection of the sending media in the option `sending mode` on the tab sending mode (on page 56).

Messages are individually configured:

- E-mail via Outlook and SMS via GSM or SMS gateway (on page 58)
- E-mail message via SMTP (on page 62)
- Voice message audiofile via modem (on page 66)
- Voice message text-to-speech via modem (on page 69)

You can find details on configuration of the parameters for messages in the **`Parameters for messages`** (on page 74) chapter.

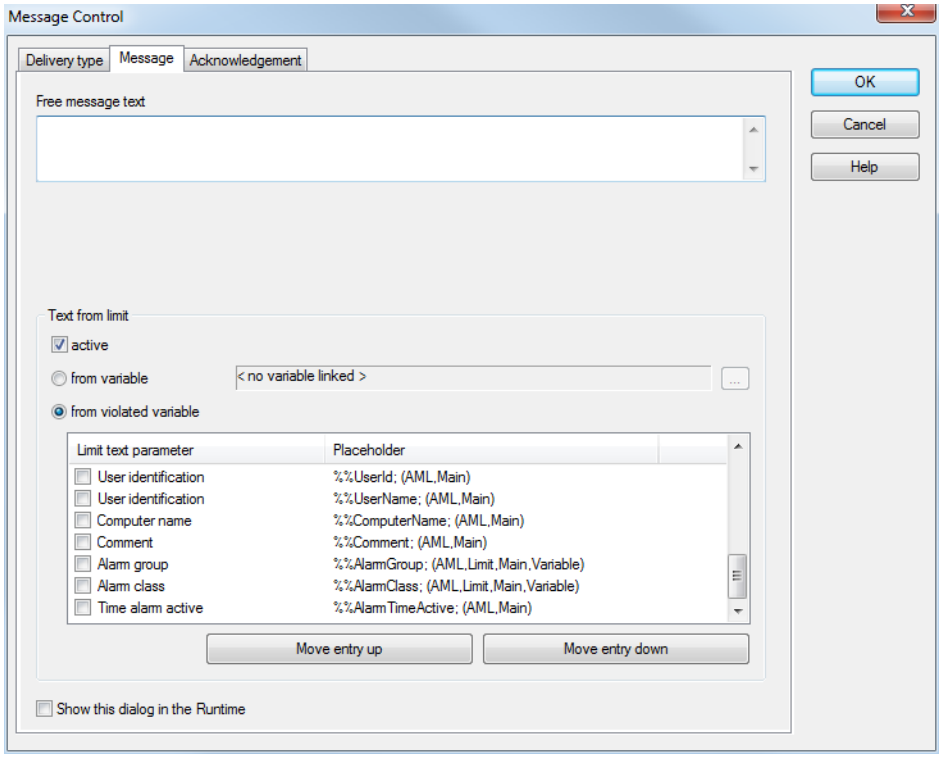
> **Information**
>
> *If parameters or values of additional variables are included in a message, these variables are not automatically logged on at the driver. If a variable is not logged on at the point of time the function is being carried out it must be logged on during the activity. Under certain circumstances (e.g. in case of slow serial communication with the hardware) this can result in delays since it has to be waited until the first value is provided by the driver and can be analyzed by the function.*

E-mail via Outlook and SMS via GSM or SMS gateway

Configuration of the message text for:

- E-mail message via Outlook
- SMS message via GSM modem

- SMS message via SMS gateway



Parameters	Description
Free message text	Integral part of the message. This text is searched for placeholders: ▸ **$**: indicates "formatted" text. The text is subsequently parsed for parameters. ▸ Text components are considered parameters if they are separated from the rest of the text with a semicolon (**;**) and end with a semicolon (**;**). ▸ A Parameter (on page 74) can be: - key of the language table - name of a variable - compound entry in the language table Parameters can be copied from the List (on page 75) of limit value text parameters. You can read more about the subject in the **Parameters for meesages** (on page 74) chapter.
Text from limit value	Properties for additional parameters from limit values.
active	Active: Adds additional parameters from the limit value to the constant text. Configuration via options: ▸ from variable ▸ from penultimate variable ▸ **List limit value text parameters** **Attention:** If this option is active only limit value violations can trigger this function.
from variable	Parameters are taken over from the defined variable. Click on button **...** in order to open the dialog for selecting a variable.
from penultimate variable	Parameters are taken over from the variable whose limit value was violated.
List limit value text parameters	Selecting parameters from list by activating the corresponding checkbox. Sequence is definded by drag & drop with the mouse or the buttons **move entry up** and **move entry down**. Split placeholders: Displays placeholders which can be used in free message text. If the option from variable or from penultimate variable or the variable selection is changed the example

	entry will be adjusted. Placeholders can be copied. Structure: **%Variable%Parameter; (origin)** ▸ **%**: percentage sign as prefix for variables and parameters ▸ **Variable**: Variable. If no variable is indicated the following parameter refers to the main variable. ▸ **Parameter**: Parameters ▸ ;: semicolon ends string. ▸ Origin: Only information, not used for free text. Indicates which variables can be used for the parameter : main variable via limit value main variable via AML additional variable via limit value or AML For details, see the **Parameters for messages** (on page 74) and Parameter limit value text (on page 75) chapters. If invalid parameters are used a corresponding error message is displayed in the message: **<Invalid Parameter: 'Parameter'>.**

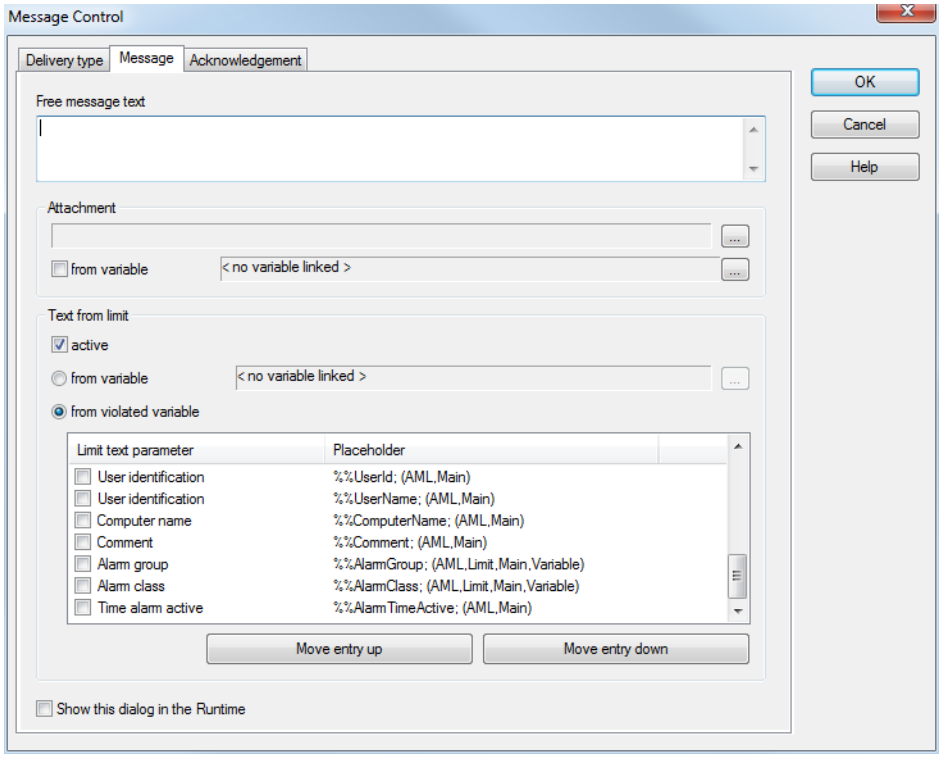
Move selected entry up	Places selected element one step higher.
Move selected entry down	Places selected element one step lower.
Show this dialog in the Runtime	Active: Opens this dialog in Runtime before executing the function. **Attention - Behaviour in Runtime:** ▸ In Runtime this function is always performed at the computer executing the process. If the function is started on a client the dialog is displayed on the computer executing the process. ▸ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out. ▸ If the dialog is edited and closed with OK the changes will be saved and the function carried out. ▸ If the dialog is closed with **cancel** no message is being sent. This process is documented in the CEL provided for the property Log in CEL all confirmations or only negative confirmations was selected.
OK	Applies all changes in all tabs and closes the dialog. Dialog can only be closed if all required options have been configured.
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

E-mail message via SMTP

E-MAIL MESSAGE VIA SMTP

Configuration of the message text for:

- E-mail message via SMTP



Parameters	Description
`Free message text`	Integral part of the message. This text is searched for placeholders: ▸ **$**: indicates "formatted" text. The text is subsequently parsed for parameters. ▸ Text components are considered parameters if they are separated from the rest of the text with a semicolon (**;**) and end with a semicolon (**;**). ▸ A Parameter (on page 74) can be: - key of the language table - name of a variable - compound entry in the language table Parameters can be copied from the List (on page 75) of `limit value text parameters`. You can read more about the subject in the **`Parameters for meesages`** (on page 74) chapter.
`Appendix`	Configuration of an appendix to the e-mail. Screenshots can be sent as appendix. Click on button **...** in order to open the dialog for selecting a file. This has to be deposited in the node Files/Graphics. When sending the message the selected file is added as attachment to the e-mail. Note: To send files from other folders with desired folder information, select the `from variable` option.
`from variable`	`Active`: Select a string variable. Click on button **...** in order to open the dialog for selecting a variable. The value of the string variables is read in Runtime and the content is analyzed. The path to the appendix is set according to the analysis: ▸ Absolute path: this is used without changes ▸ No absolute path: the value of the string variables is attached to the path to the zenon graphics folder When setting the path, a check is not carried out to see which expansion the file has. Thus in doing so, NO CHECK IS CARRIED OUT TO SEE WHICH EXPANSION HAS THE FILE. Variables can be sent desired file formats from the attachment path, in particular PDFs and Office documents from any desired path.

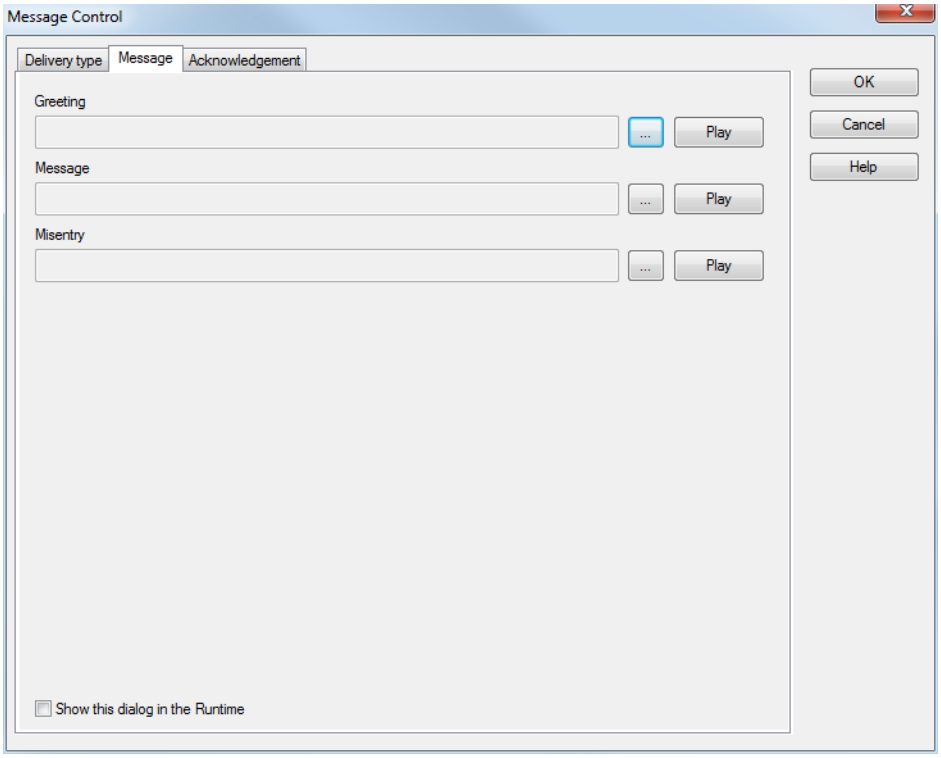
`Text from limit value`	Properties for additional parameters from limit values.
`active`	`Active`: Adds additional parameters from the limit value to the constant text. Configuration via options: ▸ `from variable` ▸ `from penultimate variable` ▸ **`List limit value text parameters`** **Attention:** If this option is active only limit value violations can trigger this function.
`from variable`	Parameters are taken over from the defined variable. Click on button `...` in order to open the dialog for selecting a variable.
`from penultimate variable`	Parameters are taken over from the variable whose limit value was violated.
`List limit value text parameters`	Selecting parameters from list by activating the corresponding checkbox. Sequence is definded by drag & drop with the mouse or the buttons **`move entry up`** and **`move entry down`**. <u>Split placeholders:</u> Displays placeholders which can be used in free message text. If the option `from variable` or `from penultimate variable` or the variable selection is changed the example entry will be adjusted. Placeholders can be copied. Structure: **`%Variable%Parameter; (origin)`** ▸ `%`: percentage sign as prefix for variables and parameters ▸ **`Variable`**: Variable. If no variable is indicated the following parameter refers to the main variable. ▸ **`Parameter`**: Parameters ▸ ;: semicolon ends string. ▸ Origin: Only information, not used for free text. Indicates which variables can be used for the parameter : main variable via limit value main variable via AML additional variable via limit value or AML For details, see the **`Parameters for messages`** (on page 74) and

	Parameter limit value text (on page 75) chapters. If invalid parameters are used a corresponding error message is displayed in the message: **<Invalid Parameter: 'Parameter'>.**
Move selected entry up	Places selected element one step higher.
Move selected entry down	Places selected element one step lower.
Show this dialog in the Runtime	Active: Opens this dialog in Runtime before executing the function. **Attention - Behaviour in Runtime:** ▸ In Runtime this function is always performed at the computer executing the process. If the function is started on a client the dialog is displayed on the computer executing the process. ▸ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out. ▸ If the dialog is edited and closed with OK the changes will be saved and the function carried out. ▸ If the dialog is closed with **cancel** no message is being sent. This process is documented in the CEL provided for the property entries into CEL all confirmations or only negative confirmations was selected.
OK	Applies all changes in all tabs and closes the dialog. Dialog can only be closed if all required options have been configured.
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

Voice message audiofile via modem

Configuration of the message text for:

- Voice message with audiofile via modem.



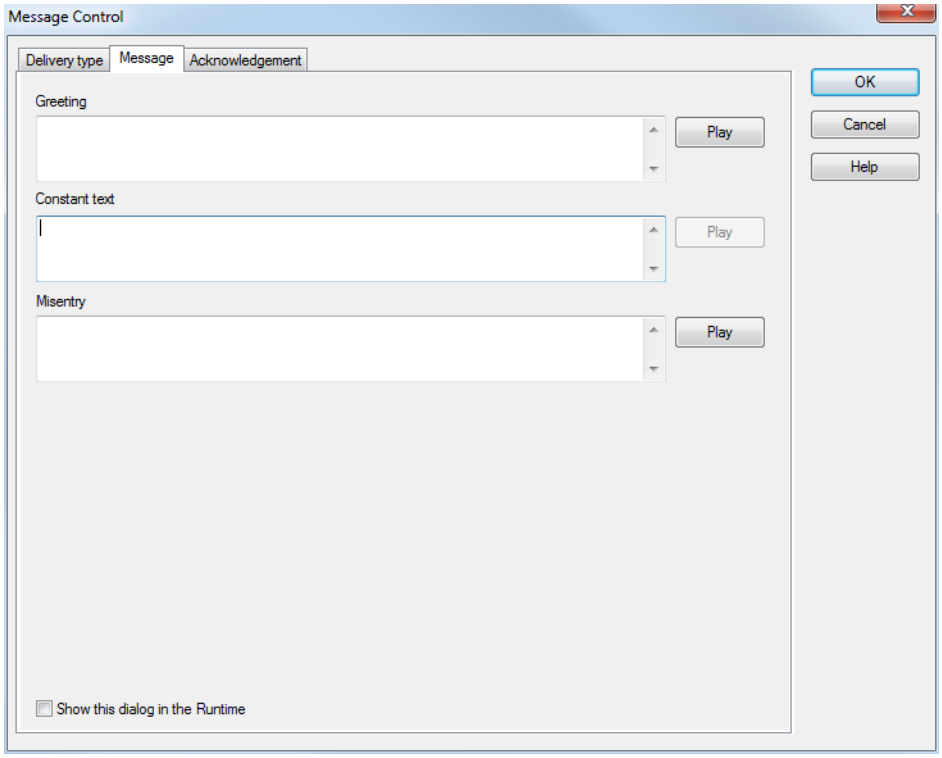
Parameters	Description
Greeting	Select the file containing the greeting text. Click on button . . . in order to open the dialog for selecting a file. This has to be deposited in the node File/Multimedia. The file can be played for testing with the button **Play**.
Message	Select the file containing the message text. Click on button . . . in order to open the dialog for selecting a file. This has to be deposited in the node File/Multimedia. The file can be played for testing with the button **Play**.
Failure entry	Select the file containing the text in case of failure entries by the recipient. Click on button . . . in order to open the dialog for selecting a file. This has to be deposited in the node File/Multimedia. The file can be played for testing with the button **Play**.
Show this dialog in the Runtime	Active: Opens this dialog in Runtime before executing the function. **Attention - Behaviour in Runtime:** ▸ In Runtime this function is always performed at the computer executing the process. If the function is started on a client the dialog is displayed on the computer executing the process. ▸ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out. ▸ If the dialog is edited and closed with OK the changes will be saved and the function carried out. ▸ If the dialog is closed with **cancel** no message is being sent. This process is documented in the CEL provided for the property entries into CEL all confirmations or only negative confirmations was selected.
OK	Applies all changes in all tabs and closes the dialog. Dialog can only be closed if all required options have been configured.

Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

Voice message text-to-speech via modem

Configuration of the message text for:

- Voice message with text-to-speech via modem.



Parameters	Description
`Greeting`	Enter the text to be displayed for greeting via the text-to-speech engine. The text can contain dynamic components. These are configured in the same way as dynamic texts (on page 74). The text can be played for testing with the button **`Play`**.
`Message`	Enter the text to be displayed as message via the text-to-speech engine. The text can contain dynamic components. These are configured in the same way as dynamic texts (on page 74). The text can be played for testing with the button **`Play`**.
`Failure entry`	Enter the text to be displayed in case of failure entries by the recipient via the text-to-speech engine. The text can contain dynamic components. These are configured in the same way as dynamic texts (on page 74). The text can be played for testing with the button **`Play`**.
`Show this dialog in the Runtime`	Active: Opens this dialog in Runtime before executing the function. **Attention - Behaviour in Runtime:** ▸ In Runtime this function is always performed at the computer executing the process. If the function is started on a client the dialog is displayed on the computer executing the process. ▸ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out. ▸ If the dialog is edited and closed with OK the changes will be saved and the function carried out. ▸ If the dialog is closed with **`cancel`** no message is being sent. This process is documented in the CEL provided for the property entries into CEL `all confirmations` or `only negative confirmations` was selected.
OK	Applies all changes in all tabs and closes the dialog.

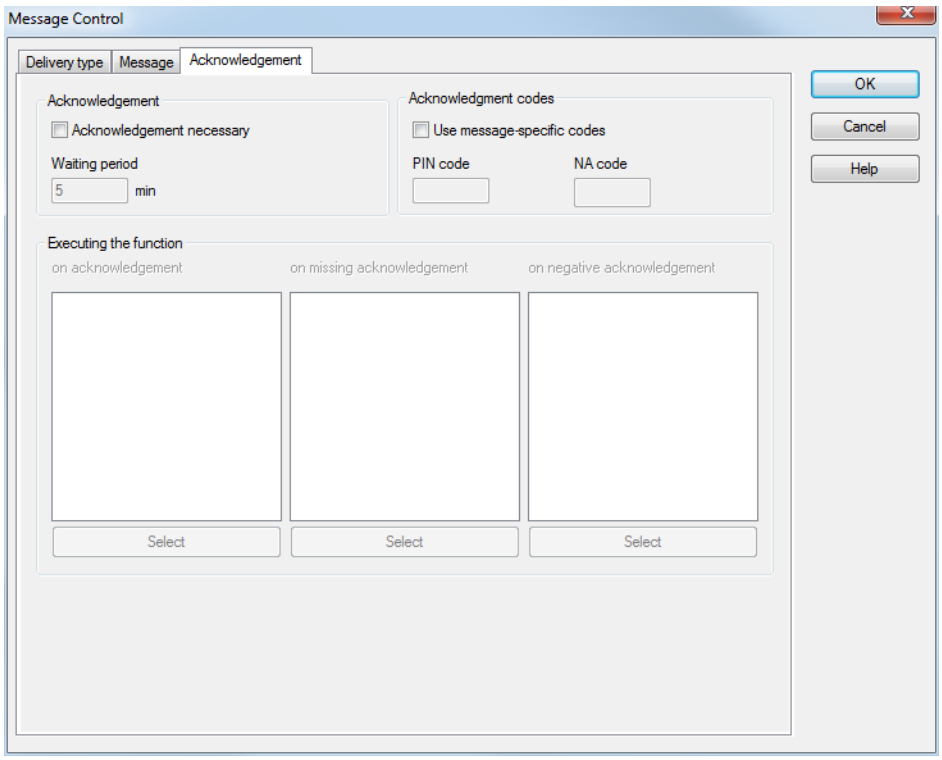
	Dialog can only be closed if all required options have been configured.
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

Acknowledgement of receipt

In this tab the required confirmations and the response to them are defined.

> **Attention**
>
> *Synchronously transmitted messages (text-to-speech message via modem (on page 69) and voice message with audio file via modem (on page 66)) can only be confirmed synchronously via the respective transmission medium. The use of other transmission methods for communication (such as email or SMS) is not possible. The message must be acknowledged immediately via the same connection which is still open.*

ACKNOWLEDGMENT SETTINGS



Parameters	Description
Acknowledgement of receipt	Settings for acknowledgement of receipt.
Acknowledgement of receipt required	Active: The message is only removed from the message queue if within the waiting period an acknowledgement of receipt has arrived or the waiting period has run out. For voice messages this option is always active and cannot be deactivated.
Waiting period	Period of time in minutes during which the receipt has to be acknowledged before the message is forwarded to a subsitute person.
Confirmation codes	Settings for acknowledgement codes.
Use message-specific codes	Inactive: The codes defined in the properties of the user administration for Message Control user/PIN code and Message Control user/NA code are used. Active: The defined codes in the following options PIN code and NA code are used for confirmation.
PIN code	Individual code for confirming the message.
NA code	Individual code for rejecting the message.
Function execution	Selection of functions to be executed after a message has been confirmed, rejected or not acknowledged at all.
In case of confirmation	Functions which are executed if the message is confirmed. A click on the **Select** button opens the function selection dialog.
In case of lack of confirmation	Functions which are executed if there is no response to the message. A click on the **Select** button opens the function selection dialog.
In case of negative confirmation	Functions which are executed if the receipt of message is rejected. A click on the **Select** button opens the function selection dialog.
OK	Applies all changes in all tabs and closes the dialog. Dialog can only be closed if all required options have been configured.
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

CONFIGURATION OF CONFIRMATION OF RECEIPT VIA-EMAIL

When interpreting receipt confirmations via email,-the first 3 characters of the subject line are ignored.

Background: Receipt confirmations are generally introduced with `RE:` etc. If outgoing messages are collected in the same folder as incoming ones, outgoing messages being interpreted as incoming confirmations of receipt must be prevented. To do this, it is assumed that receipt confirmations differ from outgoing messages in the 3 characters at the start.

Configuration: The subject of a receipt confirmation must be introduced by 3 desired characters.

6.5.5 Send Message: activate

This function activates Message Control for the project in which the function is located. When the function is called the project is logged on in Message Control. From this point of time on messages are created and transmitted.

To configure the function:

1. create a new function
2. in group Message Control select the function `Send Message: activate`
3. the function is created

Note: In order to securely log on a project during the Runtime start activate the property `Active during Runtime start` in property group `Project-specific settings` for Message Control.

6.5.6 Send Message: deactivate

This function deactivates Message Control for the project in which the function is located. When the function is called all messages of this project are removed from the message queue and the project is logged off from Message Control. Subsequently, the message queue of the project will be cleared. As long as Message Control is deactivated for a project no new messages can be created or transmitted.

To configure the function:

1. create a new function
2. in group Message Control select the function `Send Message: deactivate`
3. the function is created

6.6 Parameters for messages

The text of messages can be configured dynamically. It is searched for placeholders before dispatch and evaluated. Control characters are noted in the process. This applies for:

- **`E-mail via Outlook and SMS via GSM or SMS gateway`** (on page 58)
- **`E-mail message via SMTP`** (on page 62)
- **`Voice message text-to-speech via modem`** (on page 69)
- `Dynamic part of the subject` property:

CREATION OF A MESSAGE

- Text parts are separated from one another with a semi colon (`;`).
- **`$`**: marks text that contains the parameter.
- A parameter (on page 74) can:
  - Be a key from the language table, such as **`@MyText;`**
  - Relate to a variable, when executing the function via AML or a limit value breach such as **`%Var1;`**
  - Be a compiled entry in the language table: for example **`%@Var2+MultipleText`**
- `@` marks language switching
- `%` marks variables
- `%%` marks **`limit value text parameters`** (on page 75) for variables

A variable can be stated between the two percentage marks.
If no variable is given, the value relates to the main variable.

- Messages end with a semicolon (`;`).

Partly different parameters are available for the main variable, additional variable and depending on the function call.

Attention: If additional variables are added using a placeholder, these variables first have to be registered and read if they are not registered at the point of executing the function. The sending is then delayed. For this reason, additional variables should only be used cautiously.

LIST OF PARAMETER TEXT

Parameters	In the Runtime	Action
key: **@MyText**	A check is made to see if an entry in @MyText is present in the language table for the current language.	▸ Existing: entry is added to the message. ▸ Not existing: According to the settings of the project properties Display not translated keywords either **@MyText** or **MyText** is added to the message.
Variable: **%Var1%**	A check is made if the variable exists and the value can be read.	▸ Existing: Value of the variable is taken and added to the text as string. ▸ Not existing/not readable: The text **xxx** is added to the message.
Compound entry: **@StringTable+%var1Text**	A check is made if: ▸ 1. the variable exists and the value can be read. Value is attached to the prefix text ("MultipleText") as string. ▸ 2. an entry in the language table exists for the text	▸ Variable and text exist: text is added to the message. Example: Value of the variable is 33. The language table is then checked for **@StringTable33** and the corresponding text is added. ▸ Not existing: According to the settings of the project properties Display not translated keywords either **@MyText** or **MyText** is added to the message.

Example: **$@Caution ;%Var1%Name;**

6.6.1 Text from limit value and free text

CONFIGURATION OF TEXT FROM LIMIT VALUE

To link limit values:

1. In the **Message** tab, activate the option **Text from limit value** active (Message Control function Send a Message).
2. Activate the option from breached variable.
3. Activate the checkboxes for the desired values.
   For parameters, see **List of limit value text parameters**.
4. Order the selected parameters by dragging & dropping or with the corresponding buttons.
5. Ensure that the limit value breach triggers the function.

Alternatively: Configure the **Free message text** option.

CONFIGURATION OF FREE MESSAGE TEXT

STRUCTURE

$FREETEXT;%VARIABLE%TEXT;

Start sign -> free text between semicolon -> contains parameters with %% as a sign

- Start sign: **$** (parameters are then taken into account in free text)
- Free text: is inserted between semicolons (;)
- Prefix for **Limit field text parameters**: 2 percentage signs (%% or % variable name %)

If no variable is stated, the following parameter refers to the main variable.

- End of the character sequence: Semicolons (;)

EXAMPLE

$Text Message Control ;%%Name; Text ;%%LimitText; Text ;%Master%VariableStatus ;Text;

- **$**: Start of the message with the first message text
- **%%Name**: Name of the breached variable plus text
- **%%LimitText**: Limit value text of the breached variable plus text
- **%Master%VariableStatus**: Status of the "Master" variable plus text

Note: Place a space before the separator. The individual blocks are then separated correctly with a space-

LIST OF LIMIT VALUE TEXT PARAMETERS

Parameters	Main variable - Activating via limit value	Main variable - Activating via AML	Additional variable AML/limit value
%%Name (Variable name:)	Name of the variable, corresponds to property Name.	Name of the variable, corresponds to property Name.	Name of the variable, corresponds to property Name.
%%Identification (Identification)	Description of the variable, corresponds to property Identification.	Description of the variable, corresponds to property Identification.	Description of the variable, corresponds to property Identification.
%%LimitText (Limit value text)	Limit value text of the alarm object with interpretation of the dynamic limit value text.	Limit value text of the alarm object with interpretation of the dynamic limit value text.	Text of the state as defined in property Limit text.
%%MessageCreateTime Stamp (Time stamp of the message)	Point of time during which the message is created. (Function activation, not violation or sending).	Point of time during which the message is created. (Function activation, not violation or sending).	Point of time during which the message is created. (Function activation, not violation or sending).
%%ResourceLabel (Resource Label Identification)	Text for the resource label identification of the variable, corresponds to property Resources label.	Text for the resource label identification of the variable, corresponds to property Resources label.	Text for the resource label identification of the variable, corresponds to property Resources label.
%%LimitTimeReceived (Time alarm is generated)	Time stamp of the alarm object.	Time stamp of the alarm object.	-
%% (Current value)	No parameter. Value of the variable when activating the function (without unit).	No parameter. Value of the variable when activating the function (without unit).	No parameter. Value of the variable when activating the function (without unit).
%%VariableStatus (Status)	Status of the variable as string (when activating the function).	Status of the variable as string (when activating the function).	Status of the variable as string (when activating the function).
%%VariableTimeStamp (Time stamp of the	Time stamp of the variable (when activating	Time stamp of the variable (when activating	Time stamp of the variable (when activating

variable)	the function).	the function).	the function).
%%MessageTimeAcknowledge (Remaining time for confirmation)	Available time for acknowledgement in minutes. Only available if acknowledgement of receipt required (on page 71) was activated.	Available time for acknowledgement in minutes. Only available if acknowledgement of receipt required (on page 71) was activated.	Available time for acknowledgement in minutes. Only available if acknowledgement of receipt required (on page 71) was activated.
%%Address (Address)	Address.	Address.	Address.
%%Unit Measuring unit:	Technical unit according to `Measuring unit` property.	Technical unit according to `Measuring unit` property.	Technical unit according to `Measuring unit` property.
%%AlarmArea (Alarm area)	Alarm area of the main variable.	Alarm area of the main variable.	Alarm area of the additional variable.
%%LimitTimeAcknowledged (Time alarm is acknowledged)	-	Time stamp of the alarm object.	-
%%UserId (User identification)	-	User identification of the user who started the action. Conforms to property `User identification`.	-
%%UserName (User name)	-	Complete name of the user who started the action. Conforms to property `Complete name`.	-
%%ComputerName (Computer name)	-	Name of the computer on which the action was started.	-
%%Comment (Comment)	-	Comment of the alarm object.	-
%%AlarmGroup (Alarm/event group)	Alarm/event group of the alarm object (as text).	Alarm/event group of the alarm object (as text).	Alarm/event group of the alarm object (as text).
%%AlarmClass	Alarm/event class of the	Alarm/event class of the	Alarm/event class of the

(Alarm/event class)	alarm object (as text).	alarm object (as text).	alarm object (as text).
%%AlarmTimeActive (Time alarm is pending)	-	Period of time the alarm has been pending.	-

TROUBLESHOOTING

If invalid parameters are used a corresponding error message is displayed in the message:
<Invalid Parameter: 'Parameter'>.

This text can be changed and translated with the language switching. To do so, two keywords have to be created:

- **<Invalid Parameter:**
- **>**

If parameters are used in the wrong context or if a parameter cannot be read (for instance because a variable does not exist) the string **IDS_STRING2501 (---)** is added to the message instead of the value. This text can be changed and translated with the language switching.

7. Message Control in Runtime

In Runtime Message Control is either started automatically (property `Active during Runtime start` active) or with the function Send Message: activate (on page 73).

If Message Control is active and an event defined for message sending occurs:

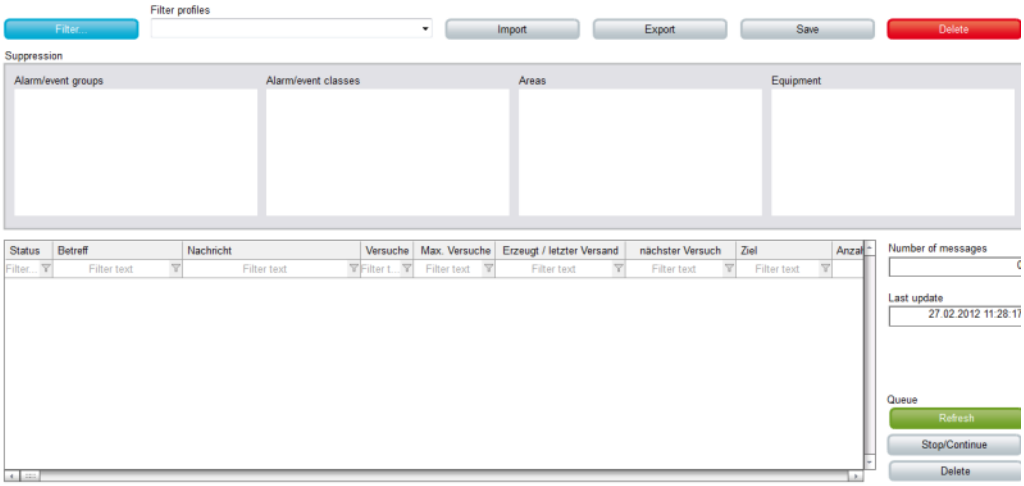
- a message configured for this is being sent
- its acknowledgement (on page 85) is being analyzed
- the functions used for acknowledgement are executed

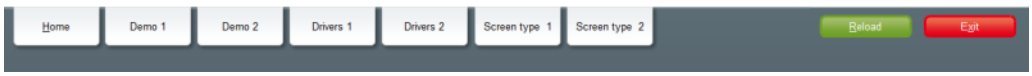
> **Attention**
>
> *During the analysis of messages in the e-mail folder all messages with matching subject line are interpreted and set to status "**read**". If in the mailbox for Message Control other messages are received, too, those may probably set to "**read**".*

MANAGEMENT IN RUNTIME

The Message Queue can be managed with a screen of type Message Control (on page 82).

Message Control is being closed when closing the Runtime or by the function Send Message: deactivate (on page 73).





TIME STAMP IN ALARM MESSAGE LIST AND MESSAGE CONTROL

When the function Send message (on page 54) is activated a copy of the alarm object is transmitted to the function. The message text is created and translated when the function is activated. If the function is activated via AML it will use a copy of the alarm object from the AML. So the content always corresponds to a current combination of process and alarm status.

MESSAGE QUEUE REMANENCE

If the Runtime is closed normally an image of the current message queue is created and saved. This image is reloaded during the start of Runtime and is being processed by Message Control. The data are saved in a folder called **`MessageRT_Image.bin`**. The following data are saved:

- Current messages in the queue

- Suppressed alarm groups
- Suppressed alarm class
- Suppressed areas
- Suppressed equipment

NETWORK

You can find further information on the behaviour in the network in the Network (on page 89)chapter.

ACKNOWLEDGING

Every message can unambiguously be identified by a GUID. According to the configured media and sending mode messages can also be acknowledged through other media. For more information about the acknowledgement of messages see chapter Message acknowledgement (on page 85).

LANGUAGE SWITCH

The language of message texts can completely be switched. To do so, the following will be analyzed:

- the area **`Constant Text`**
- the parameter `Dynamic Limit value Text`
- Limit value text of a limit value in the AML/CEL

The selectable columns in type Message Control are language switchable if they are configured correspondingly.

7.1 Connect screen of the type Message Control

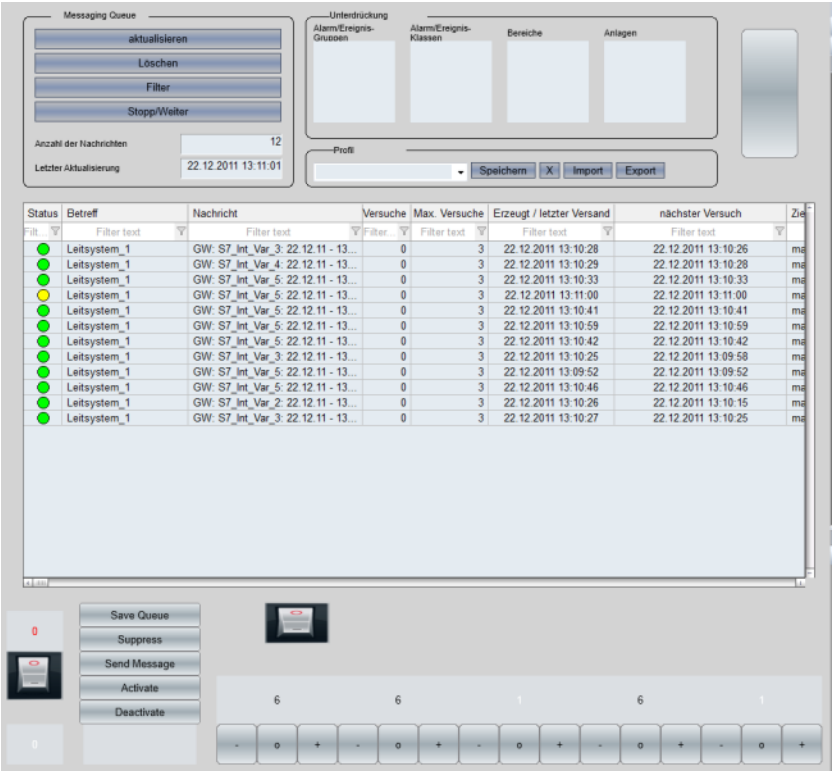
The screen of type `Message Control` makes a copy of the current message queue and control elements for analysis and filtering available in Runtime. The content of the elements displayed in the list field **`message list`** corresponds to the point of time of connecting the screen and is not automatically updated. The display can be updated anytime by clicking on the button **`Update`**. The button **`Stop/Continue`** allows for a cyclical update. A text field indicates the point of time of the last update.

The **`message list`** contains messages which

- are waiting to be sent
- are being sent
- have already been sent
- have been confirmed
- have been deleted

Suppressed groups, classes, areas and equipment whose alarms are not displayed in the list are displayed in separate list fields.

The columns of the **`message list`** can be configured individually (see Filter (on page 45)) and can be provided with individual column headings. These labelings are localizable.



Control elements	Description
Insert template	Opens the dialog for selecting a template for the screen type. Templates are shipped together with zenon and can also be created by the user. Templates add pre-defined control elements to pre-defined locations in the screen. Elements that are not necessary can also be removed individually once they have been created. Additional elements are selected from the drop-down list and palced in the screen. Elements can be moved in the screen and placed individually.
Buttons	Buttons for Runtime actions.
Refresh	Refreshes view. The current data are loaded onto the screen from the message queue. The point of time of the last update can be displayed with the field **Last update**.
Clear	Deletes selected entry from the list. It is only possible to delete messages which have not been sent yet.
Filter	Opens Dialog (on page 45) to configure the filters for the view.
Stop/continue	Starts and stops the cyclic update of the list.
Windows	Lists and fields which can be displayed in Runtime.
Number of messages	Number of currently existing messages.
Last Update	Point of time of last update.
Suppressed alarm/event groups	**Alarm/event groups** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Suppressed alarm/event classes	**Alarm/event classes** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Suppressed alarm/event areas	**Alarm/event areas** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Suppressed equipment	**Equipment** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Message waiting list	List of messages. Contains messages which have been sent, confirmed or deleted or remain to be sent.

	The messages are only displayed and cannot be edited anymore. Column headings can be named individually (on page 47) and are localizable by putting `@` before them.
Filter profiles	Profile administration
`Profile selection`	Opens the dialog for selecting a profile.
`Save`	Saves current setting as a profile.
`Clear`	Deletes profile.
`Import`	Opens dialog for importing profiles from a file.
`Export`	Opens dialog for exporting profiles from a file.

7.2 Acknowledgement of messages

Messages can be confirmed or rejected by the recipient. Depending on the medium messages must either mandatorily or optionally be acknowledged.

Medium	Mandatory acknowledging - –: no - +: yes
E-mail message via Outlook (on page 16)	–
E-mail message via SMTP (on page 17)	–
SMS message via GSM modem (on page 19)	–
SMS message via SMS gateway (on page 23)	–
Voice message audiofile via modem (on page 25)	+
Voice message text-to-speech via modem (on page 26)	+

If a message is not, incorrectly or rejectively acknowledged it is being sent to the next substitute person. If a substitute is defined, the message is not sent again. The status is logged in the CEL. During the acknowledging of messages linked functions can be carried out.

Information

Rules for interrupted calls:

- If a call is interrupted without confirmation or a decline and the interruption is not classified as a hardware error, then:
  - The call is not repeated for the same addressee
  - If substitutes or group members who have not yet been messaged are messaged
- If a call is interrupted by a hardware error, then the call for the same addressee is sent again according to the number of repetitions set in the `Max. number of attempts` property.

CONFIGURING THE CONFIRMATION

In order to configure the confirmation or rejection of the message:

1. In the function Send Message (on page 54) open the tab Acknowledgement of receipt (on page 71).
2. Activate the checkbox `Acknowledgement required` (always active for voice messages).
3. Define the waiting period (Timeout) within which the message must be acknowledged.
   After the waiting period has expired the message will be sent to the next person on the list.
4. Configure the codes for confirmation (PIN) or rejection (NA code) as follows:
   a) properties of the user:
   `Message Control user` -> `PIN code` and `NA code`

   or

   b) in the function via:
   checkbox `Use message-specific codes` -> `PIN code` and `NA code`

   If codes are configured both via properties and the function the codes of the function are valid.
   Hint: If codes are configured via the function it is recommended to make a note in Message text (on page 58) for the recipient to use the right code for activation or rejection.
5. Define the functions which are to be executed after receiving the code for:

a) message was confirmed
b) message was neither confirmed nor rejected
c) message was rejected

6. Close the dialog by clicking on the **OK** button.

ACKNOWLEDGEMENT IN RUNTIME

During sending every message which must be acknowledged receives an individual, explicit identification (GUID). All activities of the message are allocated by means of this identification. If a message must be acknowledged, the identification is automatically added to the message text.

In order to acknowledge a message, according to the medium the recipient has to send the message number and PIN in a defined format.

Thus it is also possible to acknowledge written messages through other sending modes than those configured. For this:

- the message must have been sent as e-mail or SMS
- the response must be as e-mail or SMS
- the response must contain the GUID
- the response must contain the code for acknowledgement
- the medium used for responding must be configured

E-ACKNOWLEDGING E-MAILS

An e-mail is acknowledged as follows:

- The response is being sent using the reply function of the e-mail program of the recipient.
- In the case of an e-mail, the subject must not be changed, but must be recognizable as a response by using two characters as prefix such as **RE:**
  The contents defined in Outlook in `Subject (ID)` or respectively for SMTP in `Subject for outgoing e-mails` must remain unaltered. These will identify and allocate the message during receipt in zenon.
- The response text must contain:
  - GUID

- semicolon (`;`) as separator
- PIN code or NA code

for example: `43d3c61d-ccc9-4c76-bc2c-61c2d12b0db3;0246`

- In Message Control e-mails for Message Control are identified by means of the subject.
  Attention: The response subject must be marked as response. To recognize this during receipt, the message is checked to see if it starts with two characters such as **RE:**
- Message Control analyzes GUID and code and subsequently executes the functions set in the function Send Message (on page 54).

Note: if the medium SMS was configured a message sent via e-mail can also be acknowledged via SMS.

ACKNOWLEDGING SMS

An SMS is acknowledged as follows:

- The response is being sent using the reply function of the user's telephone.
- The response text must contain:
  - GUID
  - semicolon (`;`) as separator
  - PIN code or NA code

for example: `43d3c61d-ccc9-4c76-bc2c-61c2d12b0db3;0246`

- Message Control analyzes GUID and code and subsequently executes the functions set in the function Send Message (on page 54).

Note: if the medium e-mail was configured a message sent via SMS can also be acknowledged via e-mail.

VOICE MESSAGE

Voice messages always must be acknowledged. A voice message is acknowledged as follows:

- The telephone of the recipient must support DTMF.
- The receiving modem of the equipment musst support DTMF.
- The message is read to the recipient with greeting and message text.

- The recipient acknowledges the message via key signals with `PIN code` or `NA code`.
- The recipient can:
  - have messages repeated:
    by pressing the #key on the telephone. The previously sent message (greeting, message or error message) is being repeated.
  - re-set sent code:
    by pressing the *key on the telephone. The code (PIN or NA) previously sent by the recipient is revoked. The message is replayed starting again with the greeting.
- Message Control analyzes the code and subsequently executes the functions set in the function Send Message (on page 54).

Note: Voice messages always must be acknowledged by telephone.

7.3 Network

In the network, the message queue is permanently being synchronized between server and standby server. If new messages are created at the server these are transferred to the standby. When the server or standby server are booting, the message queue is synchronized with the primary server or standby server. Since the server and standby server use different resources, messages can only be acknowledged at the sending computer.

Similarly, the list of suppressed elements (on page 49) is being synchronized and remanently saved.

The function Send message (on page 54) is only executed at the process-executing computer. Sending by the client is not supported.

USERS

Users are edited by means of the zenon user administration. For this reason, users can be edited at the server, standby server or client. Editing is also possible through web server and web client.

NO RESOURCE SHARING

The server and standby server must not share resources. Identical resources for different media at the same computer must also not be used. That means:

- E-mail (SMTP): The server and standby server use different email accounts.

- E-mail (Outlook): The server and standby server use different Outlook profiles.
- GSM: The server and standby server use different modems and SIM cards with different telephone numbers.
- SMS (SMS gateway): The server and standby server use different modems and different telephone numbers.
- Voice message: The server and standby server use different modems and different telephone numbers.
- Every computer:

uses a separate modem and telephone number respectively for:

- SMS via GSM
- SMS message via SMS gateway
- Voice message with audiofile
- Voice message text-to-speech

Uses different accounts for:

- Outlook
- E-mail message via SMTP

BEHAVIOUR IN CASE OF FAILURE

In the event of a server failure, the standby server will resend all messages in the message queue that must be acknowledged after having been upgraded to server. This may result in a duplicate sending of messages, but it ensures correct acknowledgement, since messages can only be acknowledged at the sending computer.

After a failure resulting in a reboot of the computer GSM modems probably will have to be re-initialized and their COM ports checked.

8. Messages and error handling

Messages, notes and errors can be logged and displayed with the Diagnose Viewer in a separate module `[Message Control]`. Message details can be found in section LOG entries (on page 96).

NETWORK

The network traffic can be read with Wireshark. In case of an unsecured transmission all the commands sent to the server as well as the responses are visible in plain text in Capture. In case of encrypted connections the SSL / TLS Handshake can be checked, however, the exchange of user data cannot be checked.

SERVER

In order to check if a server supports certain commands, a Telnet session can be established with the server.

In Windows 7, the Telnet client can be installed as follows:

1. Open system control.
2. Open **Programs and functions**.
3. Open **Activate or deactivate Windows functions**.
4. Activate checkbox in front of **Telnet client**.
5. Close dialog by clicking **OK**.
6. The Telnet client is being installed.

In the command line the connection can only be opened with the command **telnet [server name or IP address] [port e.g. 25 or 110]**.

MAIL SERVER

The connection to the mail server can be checked with a mail-client program (e.g. Outlook). If an unencrypted connection is used the analysis with Wireshark can display the supported SMTP and POP3 commands.

DNS

The name resolution can be checked with a program such as **nslookup**. **nslookup** can both resolve names to IP addresses and IP addresses to names. To do so, a valid DNS server has to be registered in the IP configuration of the computer.

GSM

For the analysis in case of problems with GSM it is possible for instance to use the following:

- Serial Port Monitoring:
  for supervising the communication between zenon and the GSM modem.
- HyperTerminal:
  allows to send AT commands to the modem.
  Microsoft Hyperterminal or Putty (open source) for example.
  Note: Hyperterminal was only supplied by Microsoft up to Windows XP. Can be copied manually for other operating systems.

AT COMMANDS

For troubleshooting with AT commands:

1. Ensure that the driver suitable for the operating system is installed. To do this, use the preferred driver of the modem manufacturer from the driver CD or the homepage of the manufacturer.
2. Check to see if the driver has been installed properly and that the modem is ready for operation.
3. Use a hyperterminal to connect to the COM port of the modem. Microsoft Hyperterminal or Putty (open source) for example.
4. Type in the AT command `AT` in the main window of the terminal.
   The modem should respond with `OK`.
5. Test the AT commands that have created an error message in the LOG file. Note any possible preparations for the modem, such as the creation of a message in the queue.
6. Check to see that the responses of the modem correspond to those in the list of supported AT commands (on page 11).

8.1 Check list

In case of problems check:

Problem	Possible cause	Solution
SMTP mail		
Outgoing mails are not being sent, incoming mails not received.	The sending mode is not configured correctly. LOG messages like "inactive sending mode", "server not available", "user cannot be found", "SMTP / POP 3 authentication failed" are indicating this.	▸ Is the `Sending mode active` property activated? ▸ Is the user who is supposed to receive the mail set as „Message Control User" (property `Message Control user` activated)? ▸ Was an e-mail address configured for the user? ▸ Are the user data transferred to the Runtime after a change (property `RT changeable data`)? ▸ Is the configuration of the sending mode correct? Compare to the data of the e-mail provider.
	The server is registered correctly, but is not available (error messages when connecting).	▸ Does the name resolution work? ▸ Is the server available (Ping)? ▸ Does the firewall allow for the connection with the configured port? Both the computer firewall and the domain firewall can block this. Check with Telnet.
	The server does not support a command (error message with "Command not supported" or similar in the LOG file).	▸ Check if the server supports the command (e.g. with Thunderbird or Telnet). If possible select a different configuration (e.g. SMTP AUTH commands can be bypassed with "`register at the incoming mail server before sending e-mails`". APOP can be deactivated and replaced by user and password.
	The server requires an encrypted connection. Since the mail is sent in unencrypted form the server disconnects.	▸ In this case the analysis with Wireshark shows that the server sends "STARTTLS". Since this command is not supported the connection should be protected via SSL or TLS.
	The server does not support the selected protection mechanism	▸ Select other protection configuration or plain text transmission.

	(error messages when establishing an encrypted connection).	

SMS-GSM		
zenon cannot establish a connection with the modem.	▸ Wrong COM port (file-not-found error when opening the COM port). ▸ COM port busy (e.g. by the configuration software of the manufacturer, access-denied error when opening the COM port). ▸ Error when configuring the COM port/modem not responding. ▸ Modem does not support the SMS-PDU mode (AT+CMGG error message or PDU mode error message in LOG). ▸ Probably the modem requires a PIN / is the PIN incorrect? (PIN error messages in LOG) ▸ Too many attempts to log on with an incorrect PIN? (PUK error message in LOG) In this case the connection between zenon and the modem is only possible again once the PUK has been entered manually and thus the lock been lifted.	▸ Check COM port and modem configuration. ▸ Check PIN, if necessary release manually with the PUK.
Connection is established, but sending and receiving do not work.	▸ Has a connection first been established between the modem and the GSM network? In most cases, the connection with the network is only established after accessing the manufacturer software!	▸ Connect modem for the first time with manufacturer software with the GSM network. ▸ Put modem in different location. ▸ Check telephone number.

	▸ Despite this the modem is not connected with the network (check with manufacturer software): Look for a location with better reception. ▸ The SMSC telephone number entered is not correct (load the SMSC with HyperTerminal from the modem and write it down before entering it into zenon / making first connection (also via "check modem!!").) ▸ Look at error message in LOG	

8.2 Log entries

Entries in the LOG file of the diagnose viewer.

MISCELLANEOUS MESSAGES

Level	Entry	Description
ERRORS	Error while initializing the profile list.	An error has occurred during the identifying of existing Outlook profiles.
logLe_MSG	Logging on to default mapi profile %s	If no profile was selected, the sending mode identifies the first profile and logs on with it.
ERRORS	The default profile could not be determined.	An error has occurred during the identifying of the 1. available Outlook profile.
ERRORS	Logon to mapi profile(%s) failed. Error code = %d	The login on the indicated profile was not possible.
MSG	Logging on to mapi profile %s	Log-on on profile [name].
ERRORS	An error occured while initialising the MAPI.	An error has occurred during the initializing of the MAPI.
MSG	Loggin off of MAPI profile %s.	Log-off from profile [name].
ERRORS	An error occured while uninitialising the MAPI.	An error has occurred during the deinitializing of the MAPI.
ERRORS	The Message '%s' could not be send, no connection to Outlook present: make sure it's running and a valid profile has been set.	The message [name] could not be sent since there is no Outlook connection. Make sure that an Outlook instance is running and a valid profile was registered.
DEBUG	Sending mail to Benutzer '%s' using address: %s.	A message for the user [name] is prepared and [sent] to the address.
DEBUG	Current Message %Nachricht Inhalt%	The current message (content, settings, etc.) is traced in the LOG.
DEBUG	Mail to Benutzer '%s' was successfully added to the outbox.	The message to user [name] was successfully transferred to the Outlook outbox.
ERRORS	Mail to Benutzer '%s' failed.	The message to user [name] could not be transferred to the Outlook outbox.
ERRORS	Error while using MAPI to create and transfer a mail into the Outlook-outbox.	An error has occurred during using MAPI in creating and transferring a message into the Outlook outbox.
ERRORS	Error while using MAPI to hook into the fnevNewMail-Event of Outlook.	An error has occurred during the creation of the fnevNewMail-Event hook.

ERRORS	An error occured while initializing TAPI.	An error has occurred during the intialization of TAPI.
ERRORS	An error occured while trying to open the line %s.	An error has occurred during the opening of the line [].
DEBUG	The line %s has been opened.	The line [] has successfully been opened.
ERRORS	No line selected.	For this sending mode no line was defined.
DEBUG	The line %s has been closed.	The line [] has successfully been closed.
ERRORS	An error occured while closing the line %s.	An error has occurred during the closing of the line [].
DEBUG	The current call be terminated now.	A call is active, it is closed now.
DEBUG	The is no call in progress, nothing to terminate.	No call is active, nothing can be closed.
DEBUG	Attaching to current call.	The sending mode attaches to the active call.
ERRORS	The current call is not our call,can't attach!	The current call is not the correct call and is not used.
ERRORS	The call status of the current call returns an error!	The status of the current call is "error".
DEBUG	The #-sign was entered: Repeating the last message.	# was entered: the last message is repeated.
DEBUG	The *-sign was entered: Clearing the current key.	A * was entered, the current entered code is being reset.
ERRORS	The incoming message does not contain a seperator (';')for the unique message ID: %s	The layout of the incoming message is invalid. The separator is missing.
DEEPDEBUG	Project %s signed on.	The project is logged on to Message Control.
DEEPDEBUG	Project %s signed off.	The project is logged off from the engine.
ERRORS	An attempt was made to add a message %s to the engine queue, but the message its project is not signed on with the engine!	A project which is not logged on has tried to transfer a message [content] to Message Control.
DEEPDEBUG	The message %s is added to the Queue.	The message with Guid [number] was added to the queue.
ERRORS	An attempt was made to add	A project which is not logged on has tried to transfer

	messages to the engine queue, but owner project %s is not signed on with the engine!	several messages to Message Control.
ERRORS	An attempt was made to remove a message %s from the engine queue, but owner project %s is not signed on with the engine!	A project which is not logged on has tried to remove a message [Guid] from Message Control.
ERRORS	An attempt was made to remove messages from the engine queue, but the owner project %s is not signed on with the engine!	A project which is not logged on has tried to remove several messages from Message Control.
DEEPDEBUG	All messages for project %s will be removed.	All messages of project [name] are removed from Message Control.
ERRORS	And old function 'Edit Benutzer database' was executed, but this function is no longer required/supported, execution interupted!	A function "**`display/edit user database`**" was executed. This function is only contained in versions up to zenon 6.51. They do no longer exist from zenon 7 on.
ERRORS	Line is busy -> try again…	The line is busy, the call will be repeated later on again.
ERRORS	Sending message to Benutzer %s."	Current message is sent to user [number/address]
ERRORS	Current Message:	Current message [text/file] is being sent.
DEBUG	Connected	A connection has been established.
DEBUG	Valid Acknowlegde-code entered.	A valid PIN code for confirming the message was entered.
DEBUG	Valid NegAcknowlegde-code entered.	A valid NA code for rejecting the message was entered.
DEBUG	Invalid code '%s' entered.	An invalid code was entered.
ERRORS	Inbox folder %s invalid!.	The configured inbox folder does not exist.
ERRORS	Outbox folder %s invalid!.	The configured outbox folder does not exist.
ERRORS	Statusbox folder %s invalid!.	The configured statusbox folder does not exist.
ERRORS	Prefix %s invalid!.	The configured prefix is invalid. It has to be exactly 1 character long.
ERRORS	Semaphor prefix %s invalid!.	The configured semaphor prefix is invalid. It has to be exactly 1 character long.
DEBUG	Send SMS to Benutzer %s.	An SMS is being sent to the user [address].

DEBUG	SMS to Benutzer %s transmitted to the outbox.	An SMS was transferred to the user [address] in the outbox.
ERRORS	SMS to Benutzer %s could not be placed in the outbox.	An SMS to the user [address] could not be transferred to the outbox.
WARNINGS	Index exceeded 99999, starting from 0.	Maximum index for file name reached, starts again with 0.
DEEPDEBUG	Last index set to %d.	The last applied index was set to [value].
ERRORS	Could not set last index set to %d.	The index could not be set to [value].
ERRORS	Inbox SMS-File for semaphor file: %s; is not found! Deleted the SEMA file!	Though a semaphor file was found in the inbox no corresponding SMS file was found.
ERRORS	StatusBox Status-File for semaphor file: %s; is not found! Deleted SEMA file!	Though a semaphor file was found in the statusbox no corresponding status file was found.
DEEPDEBUG	MessageChangedNotify received for Message %s ignored (no longer Mainstation)!	The project received a changed notification for message with GUID [number] from Message Control, however, it is no longer the process-executing instance.
DEEPDEBUG	Sending changed notification for message %s to SB.	The message object with GUID [number] has changed, the "new" object is sent to the standby server.
DEEPDEBUG	A changed notification of message(%s) was received, the old message object will be updated!	The project received a change telegram for the message with the GUID [number]. The local (previous) object is updated.
ERRORS	Couldn't send changed notification for Message %s to SB.	The attempt to send a change telegram for message with GUID [number] to the standby server has failed.
DEEPDEBUG	Sending remove notification for message %s to SB.	The message object with GUID [number] was removed; a remove telegram is sent to the standby server.
DEEPDEBUG	A remove notification of message(%s) was received, the message object will be removed from the list. (%s)	The project received a remove telegram for message with GUID [number]. The local (previous) object is deleted.
ERRORS	Couldn't send remove notification for Message %s to SB.	The attempt to send a remove telegram for message with GUID [number] to the standby server has failed.
DEEPDEBUG	Sending added notification for message %s to SB.	The message object with GUID [number] was created. The "new" object is also sent to the standby server.
DEEPDEBUG	An addded notification of message(%s) was received, the message object will be added to the	The project received an add telegram for message with GUID [number]. The new object is added.

	list. (%s)	
ERRORS	Couldn't send added notification for message %s to SB.	The attempt to send an add telegram for the message with GUID [number] to the standby server has failed.
DEEPDEBUG	Sending MessageCtrl-StateChanged to SB.	A MessageControl-StateChange telegram is sent to the standby server.
ERRORS	Couldn't send MessageCtrl-StateChanged notification to SB.	The attempt to send a MessageControl-StateChange telegram for message with GUID [number] to the standby server has failed.
DEEPDEBUG	An update of the message control state was received. The local state will be updated!	The project received a MessageControl-StateChange telegram for message with GUID [number]. The new object is added.
DEBUG	Sending MessageDeleteRequest for %d messages to Mainstation.	A delete request for [number] messages is sent to the process-executing computer.
DEBUG	MessageDeleteRequest Response: %d messages were deleted from the queue.	A delete response was received: [Number] messages were removed from the queue.
DEBUG	Couldn't send MessageDeleteRequest notification to Mainstation.	The attempt to send a delete request telegram to the process-executing computer has failed.
DEBUG	Requesting File alignment from mainstation.	An alignment request is sent to the process-executing computer.
DEBUG	File alignment request could not be send to mainstation.	The attempt to send an alignment request telegram to the process-executing computer has failed.
DEEPDEBUG	A getqueue request was received, sending queue.	An alignment request was received. The current queue and the current status are being sent.
DEEPDEBUG	A delete message(s) request was received!	A delete request telegram was received.
DEEPDEBUG	SB state: %s(%u)	The status of the standby server has changed and is now %s (%d).
DEEPDEBUG	Switch to server:%s	The current instance is upgraded to server.
DEEPDEBUG	Switch to standby:%s	The current instance is downgraded to standby server.
DEEPDEBUG	An ChangeSuppressionRequest was received. The local suppression-state will be updated!	A ChangeSuppression-Request was received. The local suppression status is updated.
DEEPDEBUG	Messaging TimeOut watchdog activated for Project '%s'	For the project [name] the TimeOut supervision was started.

MSG	MessageControl has been activated.	Message Control was activated.
MSG	MessageControl has been deactivated.	Message Control was deactivated.
DEEPDEBUG	Messaging-Queue is being saved.	The current message queue is saved.
ERRORS	Error occured while saving Messaging-Queue.	An error has occurred during the saving of the message queue.
WARNINGS	A Send-Message function was executed for a Group which does not contain any Message Control users!	A sending-message function was executed for a group which is empty or does not contain activated users for Message Control.
DEEPDEBUG	The Message (%s) was removed from the project queue.	The message with Guid [number] was removed from the queue.
ERRORS	Update failure: An existing instance of message '(%s)' could not be found in the local map!	The message with Guid [number] could not be updated since it was not found.
ERRORS	Messaging canceled: The Benutzer (%s) which was defined as the destination for the message %s could not be found!	The selected target user could not be found, the sending is cancelled.
DEBUG	The next Benutzer (%s) for the group (%s) will be notified, previous Benutzer was:%s	The next user [name] from user group [name] is notified, the previous user was user [name].
ERRORS	Messaging canceled: The next Benutzer for the Group (%s) could not be determined, has the group been deleted or empty?	The next user for user group [name] could not be identified. Has the group been deleted or is it empty? The sending is cancelled.
DEBUG	Messaging canceled: The end of the Benutzer list for the Group (%s) has been reached.	The end of user group [name] was reached, the sending is cancelled.
ERRORS	The message %s should not be acknowledged!	The attempt was made to confirm a message [ID] though this message cannot or does not have to be confirmed.
MSG	The incoming message for %s contains the correct Ack-Code.	The incoming message [ID] contains a valid PIN code for confirming the message.
MSG	The incoming message for %s contains the correct NegAck-Code.	The incoming message [ID] contains a valid NA code for rejecting the message.
MSG	The incoming message for %s does	The incoming message [ID] does not contain a valid

	not contain a valid code.	code.
ERRORS	The Benutzer (%s) which was the defined destination for the outgoing message %s could not be found!	The user [name] selected as target person for message [ID] could not be found!
DEBUG	The Message %s was successfully sent to %s!	The message [ID] was successfully sent to [recipient].
DEBUG	The Message %s was NOT successfully sent to %s!	The message [ID] could not be sent to [recipient].
DEBUG	The Message %s was not successfully sent to %s, and will be send again!	The message [ID] could not be sent to [recipient] and will be sent again.
DEBUG	The sending of message %s was completed.	The sending process for message [ID] has been completed.
ERRORS	msgNotify_TimeOut received for message %s, which is no longer present in the queue!	A TimeOut notification for message [ID] was received from Message Control though this message is no longer present in the queue.
ERRORS	msgNotify_Finished received for message %s, which is no longer present in the queue!	A finished notification for message [ID] was received from Message Control though this message is no longer present in the queue.
DEBUG	The message %s should have been acknowledged by now, handling TimeOut.	The message [ID] should already have been confirmed, TimeOut is being processed.
DEBUG	The message %s should have been Sent by now, handling hardware TimeOut.	The message [ID] should already have been sent, TimeOut is being processed.
MSG	Selected dispatcher '%s' is busy->try again...	The selected sending mode [type] is busy: Another attempt will be made later.
ERRORS	Selected dispatcher '%s' could not send message %s.	The selected sending mode [type] could not send the message [ID].
DEEPDEBUG	Firing MessageCreated	MessageCreated is being fired.
DEEPDEBUG	MessageCreated fired	MessageCreated was fired.
DEEPDEBUG	Firing MessageSent	MessageSent is being fired.
DEEPDEBUG	MessageSent fired	MessageSent was fired.
DEEPDEBUG	Firing MessageSendError	MessageSendError is being fired.
DEEPDEBUG	MessageSendError fired	MessageSendError was fired.

DEEPDEBUG	Firing MessageAcknowledged	MessageAcknowledged is being fired.
DEEPDEBUG	MessageAcknowledged fired	MessageAcknowledged was fired.
DEEPDEBUG	Firing MessageNegAcknowledged	MessageNegAcknowledged is being fired.
DEEPDEBUG	MessageNegAcknowledged fired	MessageNegAcknowledged was fired.
DEEPDEBUG	Firing MessageAcknowledgeTimeOut	MessageAcknowledgeTimeOut is being fired.
DEEPDEBUG	MessageAcknowledgeTimeOut fired	MessageAcknowledgeTimeOut was fired.
DEEPDEBUG	Firing MessageDeleted	MessageDeleted is being fired.
DEEPDEBUG	MessageDeleted fired	MessageDeleted was fired.
DEEPDEBUG	Firing MessageIn: (Identfier:'%s',Message:'%s')	Message is being fired, Id value is %s, MessageTExt is %s.
DEEPDEBUG	MessageIn fired: (Identfier:'%s',Message:'%s')	Message was fired, Id value is %s, MessageTExt is %s.
DEEPDEBUG	Current message count is (%d)	There are [number] messages in the queue. (Entry is created when creating and deleting messages.)
ERRORS	Can't find a Benutzer with id %s	User [name] could not be found. (Is being created when incoming messages are processed, probably the user was deleted.)

E-MAIL VIA SMTP/POP

Level	Entry	Description
DEBUG	Sent E-Mail successfully to [Zieladresse]	An outgoing e-mail has successfully been transferred to the outgoing mail server (SMTP). This does not mean that the e-mail has reached the recipient!
DEBUG	Received E-Mail successfully from [Absenderadresse]	An e-mail was successfully retrieved from the incoming mail server (POP3).
DEBUG	Creating an encrypted [SMTP / POP3] Connection using [SSLv2 and SSLv3 / TLSv1]	An encrypted connection to one of the servers is being established. Which protection mechanism is being used and to which server (protocol information: SMTP for outgoing, POP 3 for incoming) the connection is being established is contained in the LOG message.
DEBUG	Creating an unencrypted [SMTP / POP3] Connection	An unencrypted connection to one of the servers is being established.
DEBUG	Connection to [SMTP / POP3] Server	A connection to one of the servers has successfully

	established successfully	been established.
DEBUG	Connection to [SMTP / POP3] Server closed	A connection to one of the servers has been closed.
DEBUG	Authenticating to SMTP Server using SMTP AUTH [PLAIN / LOGIN]	The authentication at the outgoing mail server is being carried out with the indicated mechanism.
DEBUG	Authenticating to SMTP Server using POP3 Server Connection	A connection to the incoming mail server is being established in order to authenticate at the outgoing mail server.
DEBUG	Connection to [SMTP / POP3] Server [Servername] Port [Port] ready to go	The connection to and authentication at the indicated server have been succesfully carried out. The connection is ready for data transmission.
DEBUG	Authenticating to POP3 Server using [APOP command / BENUTZER and PASS commands]	The authentication at the incoming mail server is being carried out with the indicated mechanism.
WARNING	No Subject for the E-Mail specified	No subject was specified for an outgoing mail. Though this is not an error a mail without subject may cause a spam filter to block this mail.
WARNING	No Text for the E-Mail specified	No text was included for an outgoing mail. Though this is not an error a mail without text may cause a spam filter to block this mail.
WARNING	The Server did not respond an OK Code to EHLO. Response: [Antwort des SMTP Server]	The outgoing mail server responded with an error code to the EHLO command (response contained in the LOG message). The SMTP server does not support extensions. Extensions are used for SMTP authentication. This is not an error because SMTP authentication is not mandatory and the concerned functions carry out an error detection.
Error	No Destination E-Mail-Address specified	In case of an outgoing e-mail no target address was entered.
Error	Creating SSL/TLS Method failed. Errorcode: [Code]	The SSL/TLS method required for establishing a secure connection could not be created.
Error	Creating Connection failed. Errorcode: [Code]	The object for a connection (both secure and plain text) could not be created.
Error	Reading SSL Structure failed. Errorcode: [Code]	Reading the SSL/TLS structure of a secure connection failed.
Error	Opening Connection failed. Errorcode: [Code]	A connection (both secure and plain text) could not be established.

Error	The Server did not respond an OK Code to HELO. Response: [response from the SMTP server]	The outgoing mail server neither responded to the **EHLO** nor **HELO** command with an OK code. (The response is attached to the LOG message.) The server is considered not to be available.
Error	The Server did not send a Waiting-For-Data-Response to the AUTH LOGIN [command / Benutzer data]: [response from the SMTP server]	The outgoing mail server has not sent the expected response to the **AUTH LOGIN** command of the corresponding user names. (The response is attached to the LOG message.)
Error	The Server did not send an OK-Response to the AUTH LOGIN password data: [response from the SMTP server]	The SMTP server has not confirmed the **AUTH LOGIN** password data with OK. The login has failed.
Error	The Server did not send a Waiting-For-Data-Response to the AUTH PLAIN command: [response from the SMTP server]	The outgoing mail server has not sent the expected response to the **AUTH PLAIN** command. (The response is attached to the LOG message.)
Error	The Server did not send an OK-Response to the AUTH PLAIN data: [response from the SMTP server]	The SMTP server has not confirmed the **AUTH PLAIN** data with OK. The login has failed. (The response is attached to the LOG message.)
Error	SMTP AUTH should be used, but the Server neither supports AUTH LOGIN nor the minimum implementation AUTH PLAIN	SMTP authentication should be used, however the server does not have the necessary commands for an **EHLO** response listed in the list of supported commands (**AUTH LOGIN** and **AUTH PLAIN**, the latter is the minimum implementation in accordance with RFC 2554).
Error	[SMTP / POP3] not connected	The attempt was made to send data to the outgoing mail server or to retrieve data from the incoming mail server, although the server is not connected.
Error	The Server did not send an OK-Response to the MAIL command: [response from the SMTP server]	The outgoing mail server has not confirmed the **MAIL** command (sending of a new mail) with OK. (The response is attached to the LOG message.)

SMS

Level	Entry	Description
ERROR	[Telefonnummer] is not a valid SMSC telephone number!	The inserted telephone number of the short message center is invalid. Only the contained characters (probably + as prefix,

		otherwise only numbers) will be checked, however, it is not being checked if the telephone number actually exists or if this really is a short message center!
ERROR	[Telefonnummer] is not a valid telephone number!	The inserted recipient telephone number is invalid. Only the contained characters (probably + as prefix, otherwise only numbers) will be checked, however, it is not being checked if the telephone number actually exists or if this really is a short message center!
ERROR	Error on opening the port [COM-Port]: [Fehlernummer in HEX]	The indicated serial port could not be opened. The indicated error number can be looked up in the MSDN Library and will give clues on the probable causes of the error.
ERROR	Error on configuring the port [COM-Porp] [state / timeouts / buffers]: [Fehlernummer in HEX]	The indicated COM port could not be configured. State, timeouts and buffers are configured. The indicated error number can be looked up in the MSDN Library.
ERROR	The total input buffer ([Zahl] bytes) is too small for the [Zahl] bytes received from the modem.	The input buffer is too small to record all data sent by the modem. The input buffer is created with a size of 12 KB. For receiving an SMS with maximum size approximately 400 bytes are required. This error will occur if within a polling interval (see below) SMS data of more than 12 KB are incoming.
ERROR	The modem returned an error to the [AT-Kommando] command: [Fehlermeldung]	The modem responded with an error message to an AT command. The error message is attached to the LOG message. An AT error code may be contained in the error message. If an authentication via PIN code is required this is transferred with the command AT+CPIN. If an incorrect code is entered the modem will respond with an error to this request.
ERROR	The modem did not respond to the [AT-Kommando] command.	A timeout occurred while waiting for the response of the modem for an AT command.
ERROR	The modem did not switch in SMS PDU mode.	Though the modem has confirmed the AT command for activating the SMS-PDU mode with OK this has not been activated.
ERROR	The modem did not apply the SMSC telephone number.	Though the modem has confirmed the AT command for setting the telephone number of the short message center with OK the telephone number has not been saved as short message center.
ERROR	Error: not connected	The attempt was made to send, receive or delete an

		SMS while no connection with the modem could previously be established.
ERROR	The modem returned an error to the transmission of the PDU: [Fehlermeldung]	The modem responded with an error message to the transmission of an SMS-PDU. The error message is attached to the LOG message. An AT error code may be contained in the error message.
ERROR	The modem did not respond to transmission of the PDU.	While waiting for the response of the modem to the transmission of an SMS-PDU a timeout occurred.
ERROR	Error on writing data: [Fehlernummer in HEX]	An error occurred during the sending of data via the serial port. The error number can be looked up in the MSDN Library.
ERROR	Not all bytes have been sent.	Not all data were sent although the function for sending data via the serial port did not report an error.
ERROR	Error on reading data: [Fehlernummer in HEX]	An error occurred during receiving data via the serial port. The error number can be looked up in the MSDN Library.
ERROR	The source SMSC telephone number [Telefonnummer] contains the not supported character [Zeichen]	The telephone number of the short message centre of the sender received during the SMS receipt contains an invalid character.
ERROR	The source telephone number [Telefonnummer] contains the not supported character [Zeichen]	The telephone number of the sender received during the SMS receipt contains an invalid character.
ERROR	The syntax of the timestamp [Zeitstempel] is incorrect.	The time stamp of the GSM network received during the SMS receipt does not correspond to the expected format.
ERROR	The byte [Byte als HEX] could not be converted.	A byte contained in the PDU could not be converted into a number by a HEX string.
ERROR	Insufficient buffer to convert SMS text [Text]	The buffer for converting outgoing SMS (12 KB at 6144 characters) is insufficient.
ERROR	The format flag [Byte als HEX] is not supported.	The formatting of an incoming SMS is not supported.
ERROR	Could not convert [Byte als HEX] from GSM 7-Bit-Alphabet. Baselanguage: [Byte als HEX], Extensionlanguage [Byte als HEX]	A character contained in the 7-bit-SMS data section could not be decoded. The employed alphabets basis table and extension table are attached as language identifiers (see Standard 3GPP TS 23.038 V10.0.0 (2011-03).)
ERROR	Insufficient data to decode	An SMS was received in several parts, the individual

	multilanguage fragmented SMS!	parts using different alphabets. However, the information on the alphabets needed for conversion is not complete.
ERROR	The input byte stream is not a valid unicode text.	A received Unicode SMS does not contain valid Unicode text (only an even byte number is valid, since there are 2 bytes per character in Unicode).
ERROR	Insufficient Buffer to decode PDU	The 12 KB buffer is too small for decoding a PDU.
ERROR	The PDU is not a valid byte stream.	A received PDU is not a valid hex dump of a byte stream.
ERROR	The PDU length does not match the SMS text length	The number of bytes contained in a received PDU does not match the length specification in the SMS control data.
ERROR	The Modem requests the PUK.	The modem requests the Personal Unlock Key (PUK) for authentication. This is not used by Message Control and, as a consequence, the attempt to establish a connection with the modem is cancelled. After an incorrect PIN code was entered several times the SIM card in the modem requests the PUK for authentication, subsequently requiring to reset the PIN code. **Attention:** Incorrect entering of the PUK may render the SIM card useless!
ERROR	The Modem responded with an unknown PIN Status: [Antwort des Modems]	The PIN status check of the modem delivered an unexpected result. The response of the modem to the check of the PIN status is attached to the LOG message.
ERROR	The Modem requests a PIN and the PIN-Input is empty.	The modem requires an authentication with a PIN code, however, the user did not enter any.
ERROR	The Modem did not respond with READY-State after PIN-Authentication: [String]	Although the transmission of the PIN code was confirmed with OK by the modem, the PIN status of the modem was not set on “READY” (no PIN input required anymore). The response of the modem to the check of the PIN status is attached to the LOG message.
DEBUG	Port [COM-Port] opened successfully	The indicated serial port was opened successfully.
DEBUG	Modem configured successfully for SMS-PDU-Mode and SMSC [Telefonnummer]	The modem was successfully configured for the SMS-PDU mode and the indicated telephone number of the short message center.
DEBUG	Connection to modem closed.	The connection to the modem and the serial port was

		closed.
DEBUG	SMS successfully sent to [Telefonnummer]	An SMS was successfully sent to the indicated telephone number.
DEBUG	Successfully received SMS from [Telefonnummer]	An SMS was successfully received from the indicated telephone number.
DEBUG	The Modem does not need a PIN.	The modem does not require a PIN authentication.
DEBUG	Authentication with the PIN succeeded.	Authentication with the configured PIN code succeeded.

OPENSSL LIBRARY

Error codes and the error messages from the OpenSSL Library in format „**`Errorcode: [Code]. Errormessage: [String]`**".

The server responses have the following structure:

- POP3 responses only start with „OK“ if no error has occurred.
- SMTP server responses start with a three-digit status code:
  - 2xx = OK
  - 3xx = waiting for data
  - 4x and 5 xx = error

Level	Entry	Description
ERROR	No Destination E-Mail-Address specified	In case of an outgoing mail no target address was entered.
ERROR	Creating SSL/TLS Method failed. Errorcode: [Code]. Errormessage: [String]	The SSL/TLS method required for establishing a secure connection could not be created.
ERROR	Creating Connection failed. Errorcode: [Code]. Errormessage: [String]	The object for a connection (both secure and plain text) could not be created.
ERROR	Reading SSL Structure failed. Errorcode: [Code]. Errormessage: [String]	Reading the SSL/TLS structure of a secure connection failed.
ERROR	Opening Connection failed. Errorcode: [Code]. Errormessage: [String]	A connection (both secure and plain text) could not be established.
ERROR	The Server did not respond an OK Code to HELO. Response: [Antwort des SMTP Servers]	The outgoing mail server neither responded to the EHLO nor HELO command with an OK code (response is attached to the LOG message). The server is considered not to be available.
ERROR	The Server did not send a Waiting-For-Data-Response to the AUTH LOGIN [command / Benutzer data]: [Antwort des SMTP Server]	The outgoing mail server has not sent the expected response to the AUTH LOGIN command of the corresponding user names. The response is attached to the LOG message.
ERROR	The Server did not send an OK-Response to the AUTH LOGIN password data: [Antwort des SMTP Server]	The SMTP server has not confirmed the AUTH LOGIN password data with OK. The login has failed.
ERROR	The Server did not send a Waiting-For-Data-Response to the AUTH PLAIN command: [Antwort des SMTP Server]	The outgoing mail server has not sent the expected response to the AUTH PLAIN command. The response is attached to the LOG message. .
ERROR	The Server did not send an OK-Response to the AUTH PLAIN data: [Antwort des SMTP Server]	The SMTP server has not confirmed the AUTH PLAIN data with OK. The login has failed. The response is attached to the LOG message.
ERROR	SMTP AUTH should be used, but the Server neither supports AUTH LOGIN nor the minimum implementation AUTH PLAIN	SMTP authentication should be used, however, during the EHLO response the server has not listed the required commands in the list of supported extension commands. The following commands are required: AUTH LOGIN and AUTH PLAIN, the latter is the minimum

		implementation according to RFC 2554.
ERROR	[SMTP / POP3] not connected	The attempt was made to send data to the outgoing mail server or to retrieve data from the incoming mail server, although the server is not connected.
ERROR	The Server did not send an OK-Response to the MAIL command: [Antwort des SMTP Server]	The outgoing mail server has not confirmed the MAIL command (sending of a new mail) with OK. The response is attached to the LOG message.
ERROR	The Server did not send an OK-Response to the RCPT command: [Antwort des SMTP Server]	The outgoing mail server has not confirmed the RCPT command (adds a recipient) with OK. The response is attached to the LOG message.
ERROR	The Server did not send a Waiting-For-Data-Response to the DATA command: [Antwort des SMTP Server]	The outgoing mail server has not sent the expected response to the DATA command (mail data is being transferred). The response is attached to the LOG message.
ERROR	The Server did not send an OK-Respone to the transmitted Data: [Antwort des SMTP Server]	The outgoing mail server has not confirmed the transmitted mail data with OK. The response is attached to the LOG message.
ERROR	APOP Authentication is required but the Server did not send APOP Data	For authentication at the incoming mail server the APOP command is required, however, the incoming mail server has not sent the required data.
ERROR	Creating APOP Digest failed. Errorcode: [Code]. Errormessage: [String]	The MD5 hash for APOP authentication could not be created.
ERROR	The Server did not send an OK-Response to the [POP3-Kommando] command: [Antwort des POP3 Server]	The incoming mail server has not confirmed a command with OK. The response is attached to the LOG message. The following commands might be affected: APOP, USER, PASS, STAT, RETR, DELE. **Note:** If the DELE command still contains the reference „Could not delete mail“ the e-mail was received though could not be deleted. Since the e-mail was successfully received it is being forwarded to Message Control.
ERROR	The OK-Response to the STAT command did not have the expected Format: [Antwort des POP3 Server]	The response of the incoming mail server to the STAT command did not correspond to the format definition according to RFC 1939. Format: **OK<space>[number of mails in mailbox]<space>[total size of all mails in mailbox in bytes]**

ERROR	[Sending / Receiving] Data failed. Errorcode: [Code]. Errormessage: [String]	Sending or receiving data via an OpenSSL network connection has failed.