



**zenon**  
by COPA-DATA

# zenon manual

## Message Control

v.8.10



© 2019 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. Technical data is only used for product description and are not guaranteed qualities in the legal sense. Subject to change, technical or otherwise.

# Contents

<b>1</b>	<b>Welcome to COPA-DATA help.....</b>	<b>5</b>
<b>2</b>	<b>Message Control.....</b>	<b>5</b>
<b>3</b>	<b>Compatibility with version 6.xx .....</b>	<b>6</b>
3.1	Compatibility with version 5.50 .....	8
<b>4</b>	<b>Requirements .....</b>	<b>9</b>
<b>5</b>	<b>Limitations.....</b>	<b>10</b>
<b>6</b>	<b>Supported AT commands .....</b>	<b>11</b>
<b>7</b>	<b>Configure Message Control.....</b>	<b>13</b>
7.1	General settings for sending.....	15
7.1.1	E-mail message via Outlook.....	16
7.1.2	E-mail message via SMTP .....	17
7.1.3	SMS message via GSM modem .....	19
7.1.4	SMS message via SMS gateway.....	22
7.1.5	Voice messages .....	23
7.1.6	zenon6.ini entries.....	31
7.2	Project-specific settings.....	44
7.3	Create a screen of type Message Control.....	46
7.4	User Administration .....	50
7.5	Functions .....	54
7.5.1	Screen switching to a screen of type Message Control .....	54
7.5.2	Save current queue .....	58
7.5.3	Group/class/area/equipment suppressed .....	59
7.5.4	Send a Message .....	64
7.5.5	Send Message: activate .....	85
7.5.6	Send Message: deactivate.....	85
7.6	Parameters for messages .....	85
7.6.1	Text from limit value and free text.....	87
<b>8</b>	<b>Message Control in the Runtime.....</b>	<b>91</b>
8.1	Connect screen of the type Message Control.....	93
8.2	Acknowledgement of messages.....	96
8.3	Voice messages in Runtime .....	100

8.4 Network .....	101
8.5 Voice message process .....	103
<b>9 Messages and error handling.....</b>	<b>106</b>
9.1 Check list .....	108
9.2 LOG entries .....	111
9.2.1 Miscellaneous messages.....	111
9.2.2 E-mail via SMTP/POP .....	118
9.2.3 Voice over IP .....	123
9.2.4 SMS .....	123
9.2.5 LOG entries - OpenSSL library .....	127

# 1 Welcome to COPA-DATA help

## ZENON VIDEO-TUTORIALS

You can find practical examples for project configuration with zenon in our YouTube channel ([https://www.copadata.com/tutorial\\_menu](https://www.copadata.com/tutorial_menu)). The tutorials are grouped according to topics and give an initial insight into working with different zenon modules. All tutorials are available in English.

## 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:

- ▶ EMail via Microsoft Outlook or via SMTP
- ▶ SMS via GSM modem or via a SMS gateway
- ▶ Voice message with audio file via modem or via voice over IP to telephone
- ▶ Voice message, text-to-speech via modem or via voice over IP to 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.

#### 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 *messag32.ini* but in file *zenon6.ini*

- ▶ Evaluating the limit value texts: Up to now the evaluation of compound texts in module Message Control differed from the evaluation of standard limit value 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 Runtime 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 106).
- ▶ **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 106) in the output window indicates this. After the conversion you must check the settings of the function.
- ▶ **Runtime changeable files:**

Because the user administration was changed, Runtime changeable files are no longer required 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 106) 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: *message32.ini*
- ▶ 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 assigned 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.

## 3.1 Compatibility with version 5.50

Direct transfer of content from version 5.50 to version 8.10 is not possible. In order to be able to convert project configurations of the Message Control module from version 5.50 to the current version, carry out the following steps:

1. First convert the projects to zenon version 6.51.  
All users and groups are updated.

**Note:** This interim step is absolutely necessary. Direct conversion of the project configuration from version 5.50 to the current version is not possible!

2. Then convert the project conversion from version 6.51 to the current zenon version.



3. In the current version, check the settings for the user and user groups. Carry out all necessary corrections in the current version if required.



### Attention

Note the compatibility notices in the Compatibility with version 6.xx (on page 6) chapter.

## 4 Requirements

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

Medium	Requirement
<b>E-mail message via Outlook (on page 16):</b>	Running Microsoft Outlook instance
<b>E-mail message via SMTP (on page 17):</b>	<p>The Runtime server executing the sending must be connected to a network.</p> <p>An e-mail account must be available. The servers with access to the account must be available.</p>
<b>SMS via GSM (on page 19):</b>	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.
<b>SMS message via SMS gateway (on page 22):</b>	Compatible suitable SMS server.
<b>Speech output: (on page 23)</b>	<p>Voice modem with DTMF functionality or voice over IP server.</p> <p>Text-to-speech engine.</p>
<b>Telephone: (on page 23)</b>	<p>Voice modem with DTMF functionality.</p> <p>If an ISDN adapter is used it must support the following functions.</p> <ul style="list-style-type: none"> <li>▶ Support of voice messages (LINEMEDIAMODE_AUTOMATEDVOICE)</li> <li>▶ Support of DTMF signals (for receiving the PIN code when acknowledging a message)</li> </ul>

Medium	Requirement
	<ul style="list-style-type: none"> <li>▶ Support of Telephony Application Programming Interface (TAPI)</li> </ul>
Voice over IP: (on page 23)	<p>Voice over IP Access.</p> <p>The following conditions must be met in order to use VoIP:</p> <ul style="list-style-type: none"> <li>▶ The VoIP provider must support the <b>SIP</b> and <b>RTP</b> protocols.</li> <li>▶ The corresponding ports must be open in the firewall: SIP (Default: 5060) RTP (Default: 4000) RTCP (Default: 4001)</li> </ul>

## PORTS FOR VOIP

Two protocols are used for VoIP:

- ▶ SIP: uses port for SIP port.  
**Note:** SIP-TLS is not currently supported. VoIP should therefore not be used in critical environments that require encryption.
- ▶ RTP: uses ports for RTP and RTCP.  
RTCP must always be 1 higher than RTP.

The ports for the Voice over IP dispatch method can be freely configured using the properties of the **Voice over IP** group.

Ranges:

- ▶ SIP: 0 - 65535
- ▶ RTP: 0 - 65534

Default values:

- ▶ **SIP port:** 5060
- ▶ **RTP port:** 4000
- ▶ **RTCP port:** 4001  
This port cannot be changed. It is always 1 higher than the **RTP port**.

## 5 Limitations

Under certain circumstances, the functionality of Message Control can be limited or not available.

## DTMF SIGNALS WITH CELLPHONES AND VOICE OVER IP

DTMF signals from some cellphones can be interpreted with misunderstandings by some VoIP providers.

**Reason:** Some cellphones only send very long DTMF signals. These can be sporadically interpreted as two short button presses by Voice over IP from the provider.

**Solution:** Try a different combination of cellphone and VoIP provider.

## DETECTING SIGNALS

DTMF signals from some cellphones can be interpreted with misunderstandings by some VoIP providers.

**Solution:** Try a different combination of cellphone and VoIP provider.

## WINDOWS CE

The **Message Control** module is not available under **Windows CE**.

# 6 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:  Note: With a SMSC number, set the country code too (00xy or +xy; xy stands for the	AT+CSCA="[SMSC Telephone number]"\r	\r\nOK\r\n	at the end

Goal	Command	Answer	Comment on the response.
corresponding figures).			
Check to see if the modem has switched to SMS-PDU mode:	<code>AT+CMGF?\r</code>	<code>\r\nOK\r\n</code>	at the end
Check to see if the modem has set the given SMS message center:	<code>AT+CSCA?\r</code>	<code>\r\nOK\r\n</code>	at the end
Send message to this number. Modem signalizes <i>Ready for PDU</i> if the command was accepted:	<code>AT+CMGS="[PDU length]"\r</code>	<code>\r\n&gt;[Space]</code>	
<i>PDU</i> for the SMS to be sent to the number specified previously with completion flag:	<code>[SMS-PDU][0x1A]</code>	<code>\r\nOK\r\n</code>	at the end
List SMSs from the memory with status set by flag:	<code>AT+CMGL=[Flag]\r</code>	<code>\r\nOK\r\n</code>	at the end
Delete all SMSs with status set by flag from the memory (regardless of index):	<code>AT+CMGD=0,[Flag]\r</code>	<code>\r\nOK\r\n</code>	at the end
Delete SMS with the given index from the memory:	<code>AT+CMGD=[Index]\r</code>	<code>\r\nOK\r\n</code>	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 SEARCH

Error messages contain the notice *ERROR* in the response. You can find details on troubleshooting and testing the connection in the Messages and error handling (on page 106) chapter.

## 7 Configure Message Control

Message Control is configured via:

- ▶ global properties (on page 15) of the media for sending
- ▶ project-specific properties (on page 44) for the message
- ▶ Functions (on page 54) which are executed in the Runtime for sending and management of messages

Messages can be sent and acknowledged as:

- ▶ E-email
- ▶ SMS
- ▶ Voice message

### CONFIGURING THE SENDING OF A MESSAGE

In order to configure the sending of a message:

1. Select the medium for dispatch in the properties group in **Message Control**.  
**Note:** You can find this properties group if you click on the current **Workspace** in the zenon Editor.
2. To do this, activate the **Sending mode active** property in the area of the properties group that represents the desired dispatch type.  
**Note:** The **Sending mode active** property can be activated for several types of dispatch.
3. Configure (on page 15) the medium.
4. Configure (on page 44) the project-specific properties.
5. Configure the users (on page 50) or user groups that are to be reached via the medium.
6. Create a Send message (on page 64) function and link the function:
  - a) with a button or
  - b) with the property **Execute function instantly** in case of limit values which are supposed to send this message in case of violations

For linking with a limit value, content that is dynamically generated in the Runtime can be sent. In order to use several **Send a Message** functions for a limit value, use a script that contains these functions and is linked to the limit value.

### 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
  - ▶ **Mail subject** 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
- ▶ Description and incorrect entry: is defined for language messages using:
  - ▶ Voice message via text to speech (on page 23)
  - or
  - ▶ Voice message by audio file (on page 23)
- ▶ Message text: defined for e-mails, SMS and voice messages in:
  - ▶ the Option (on page 71) **Constant Text**

## CONFIGURE MANAGEMENT OF THE MESSAGE QUEUE

In order to administrate messages in the Runtime:

1. create a screen of type Message Control (on page 46)
2. create and configure a function Screen switch (on page 54) for this screen.
3. link the function to a button

By doing so in the Runtime, the status of messages can be checked and the message queue can be monitored.

## MESSAGE CONTROL IN THE RUNTIME

In order to use Message Control in the 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 85): If this function is carried out Message Control is started. Messages are only processed from this point of time on.

When the Runtime is closed normally, 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 58) function. Message Control can be closed with the function Send Message: deactivate (on page 85) in the 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 64) 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 59) in the Runtime. Alarms and equipment can be activated or deactivated. Suppressed entries are logged in the CEL if the property **Log in CEL** is configured with *all confirmations* or *only negative confirmations*.

Activating or deactivating entries is carried out remanently in the network. If for instance for maintenance 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.

## 7.1 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 22)
- ▶ Voice message audiofile via modem (on page 25)
- ▶ Voice message text-to-speech via modem (on page 26)
- ▶ Voice over IP as audio file (on page 28)
- ▶ Voice over IP as text-to-speech (on page 29)

In order to configure the properties:

1. Highlight the **Workspace**
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 31).



## Attention

These properties of the **Message Control** group in the workspace are not project-specific. Changes to this area are only effective in the Runtime after it has been restarted.

### 7.1.1 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 valid account and 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 82) for acknowledgement. This period of time is defined with the function **Send message** (tab acknowledgement (on page 82)).



## Attention

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

- ▶ 32bit Outlook requires the 32-bit version of zenon.
- ▶ 64bit 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**.  
**Note:** You can find this property in the **Mail message (Outlook)** area of the **Message Control** property group.
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 44) the properties for **Project-specific settings**.
5. Configure Users (on page 50) and user groups.
6. Create required functions (on page 54) and connect them with buttons or events.

## BEHAVIOR 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.

### 7.1.2 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 82) for acknowledgement. This period of time is defined with the function **Send message** (tab acknowledgement (on page 82)).

## CONFIGURATION

1. Activate the sending mode via the property **Sending mode active**.  
**Note:** You can find this property in the **Mail message (SMTP)** area of the **Message Control** property group.
2. Configure the properties for:
  - ▶ **Incoming mail server**
  - ▶ **Outgoing mail server**
3. Configure (on page 44) the properties for project-specific settings.
4. Configure Users (on page 50) and user groups.
5. Create required functions (on page 54) and connect them with buttons or events.

## BEHAVIOR 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 that needs it and then closed again after the data exchange is complete. Whilst a message is sent, another thread can query the received post or also send a message. 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 **Keep read mail on server** property is deactivated - deleted from the server with the DELE command.  
Note for HTML emails: All HTML control characters are removed on receipt.
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.

### 7.1.3 SMS message via GSM modem

For sending a message as SMS via a GSM modem:

- ▶ the sending mode was 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**.  
**Note:** You can find this property in the **SMS message (GSM modem)** area of the **Message Control** property group.
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 44) the properties for **Project-specific settings**.
7. Configure Users (on page 50) and user groups.
8. Create required functions (on page 54) and connect them with buttons or events.

## BEHAVIOR 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
<b>Timeout for AT commands</b>	<i>10000 ms</i>	The modem has 10 seconds to respond to the transmission of an AT command.
<b>SMS polling interval</b>	<i>20000 ms</i>	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.
<b>COM timeouts</b>	Write: <i>5 seconds</i> read: <i>immediate return</i>	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.

Entry	Value	Description
COM state configuration	<ul style="list-style-type: none"> <li>▶ <b>Baud:</b> <i>9600</i></li> <li>▶ <i>Binary mode</i></li> <li>▶ <b>Parity check:</b> <i>Inactive</i></li> <li>▶ <b>Clear-to-Send</b> and <b>Data-Set-Ready:</b> <i>not controlled</i></li> <li>▶ <b>Data-Terminal-Ready</b> and <b>Request-To-Send Flow Control:</b> <i>activated</i></li> <li>▶ <b>Bit per byte:</b> <i>8</i></li> <li>▶ <b>Parity:</b> <i>None</i></li> <li>▶ <b>Stop Bit:</b> <i>1</i></li> </ul>	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	<p>The individual PDUs have a maximum size of 400 byte. During the sending of a SMS the individual PDUs are sent one after another.</p> <p>During receiving all SMS arrived since the last <b>polling interval</b> 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.</p>
Size of the total buffer for conversion and incoming SMS in byte	12288	<p>Since a SMS-PDU occupies a maximum of 400 bytes in the receive buffer within one <b>polling interval</b> of 20 seconds, a maximum of 30 SMS parts with maximum size can be received with this buffer.</p> <p>The maximum size of an SMS to be sent is 6144 Unicode characters.</p>

### 7.1.4 SMS message via SMS gateway

For sending a message as SMS via a SMS gateway:

- ▶ the sending mode was activated via the property **Sending mode active**
- ▶ a compatible SMS server must be present and configured

## PROCEDURE

### SMS SERVER CONFIGURATION

For the correct configuration of the SMS server, please contact the manufacturer.

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**.  
**Note:** You can find this property in the **SMS message (SMS gateway)** area of the **Message Control** property group.
2. Configure **Outbox folder**, **Inbox folder** and **Status folder**.
3. Configure the **First letter of the SMS file**.  
This starting letter is unique for the project. If a SMS gateway is used by various projects, for each project an unique prefix must be defined.
4. configure **First letter of the lock/semaphore file**: Sets the initial letter for the lock/semaphore files. This starting letter is unique for the project. If a SMS gateway is used by various projects, for each project an unique prefix must 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 cellphone 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 44) the properties for **Project-specific settings**.
  7. Configure Users (on page 50) and user groups.
  8. Create required functions (on page 54) and connect them with buttons or events.

### BEHAVIOR 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.

### SMART ALARMING

If the **Smart alarming** property has been activated, SMS messages are handled as follows:

1. All outgoing messages are checked.
2. If the messages starts with the text "**ALARM!**", the message is not sent in the usual format (*ID; message*), but in the format *ALARM: ALARM! ID; message*
3. With the sending of such a message, the recipient's mobile phone is automatically switched to very loud if this is supported by the telephone.

**Note:** Incoming messages are also checked for this character string.

### 7.1.5 Voice messages

Voice messages can be sent as:

- ▶ Audio file via modem (on page 25)
- ▶ Text-to-speech via modem (on page 26)

- ▶ Voice over IP as audio file (on page 28)
- ▶ Voice over IP as text-to-speech (on page 29)

You can read information about the process in the Runtime in the subchapters and in the **Voice messages in Runtime** (on page 100) chapter.

## MODEM

The following must be the case in order to send messages via a modem:

- ▶ a modem must be connected to the sending computer and must be configured
- ▶ The telephone of the recipient must support DTMF

## VOICE OVER IP

To send messages by means of Voice over IP, it must be correctly configured.

### Mandatory fields:

- ▶ **Server address**
- ▶ **User name**
- ▶ **Password**

Furthermore, the following must be the case:

- ▶ The telephone of the recipient must support DTMF
- ▶ The VoIP provider supports the **SIP** and **RTP** protocols
- ▶ The firewall has the corresponding ports open
  - ▶ SIP (Default: 5060)
  - ▶ RTP (Default: 4000)
  - ▶ RTCP (Default: 4001)

**Attention:** Not all technical possibilities of the protocol have been implemented in Message Control.

## AUDIO FILE

The following must be the case in order to send messages as an audio file:

- ▶ One audio file each must be saved in the Files/Multimedia node for:
  - ▶ Description
  - ▶ Acknowledgement
  - ▶ Misentry



## TEXT-TO-SPEECH

The following must be the case in order to send messages as text-to-speech:

- ▶ The telephone of the recipient must support DTMF
- ▶ A text-to-speech (TTS) engine must be installed.
- ▶ The corresponding language file must be installed

### 7.1.5.1 Voice message audiofile via modem

For the sending of a message as an audio file via a modem, the following must be the case:

- ▶ the sending mode was activated via the property **Voice modem (audio file)**
- ▶ One audio file each must be saved in the Files/Multimedia node for:
  - ▶ Description
  - ▶ Acknowledgement
  - ▶ Misentry
- ▶ a modem must be connected to the sending computer and must be configured
- ▶ The telephone of the recipient must support DTMF

**Attention:** The value for the project-specific **Waiting period between trials and timeout** property must be greater than the value for the **Waiting time** option in the **Send a Message** functions (**Receipt Confirmation** tab).

## PROCEDURE

### CONFIGURATION

1. Activate the sending mode via the property **Voice modem (audio file)**.
2. Configure the settings for **Timeout [min]** 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 44) the properties for **Project-specific settings**.
5. Configure Users (on page 50) and user groups.
6. Create required functions (on page 54) and connect them with buttons or events.
7. This method must be acknowledged.

## BEHAVIOR IN RUNTIME

1. A function calls up the dispatch method.
2. A call is initiated via the modem.
3. The connection is considered established as soon as the call to the counterparty is signaled. The greeting is played from this point. The greeting can be repeated using the #- key. Then the PIN must be entered. Only then is the message played back.
4. The message must be acknowledged (on page 96) 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 will be played back again

**Note:** Only once the line has been closed is the status of the message evaluated (confirmed, rejected or missing). As long as the line is open, it is possible to start over by pressing the star key \* and entering the code again. PIN can thus be changed to NA, for example.

### 7.1.5.2 Voice message text-to-speech via modem

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

- ▶ the sending mode was activated via the property **Voice modem (Text-to-Speech)**
- ▶ 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

**Attention:** The value for the project-specific **Waiting period between trials and timeout** property must be greater than the value for the **Waiting time** option in the **Send a Message** functions (**Receipt Confirmation** tab).

## 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 via the property **Voice modem (Text-to-Speech)**.
2. Configure the settings for **Timeout [min]** 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 the properties for text-to-speech:
  - ▶ **Voice**
  - ▶ **Speech rate**
  - ▶ **Volume**
5. Configure (on page 44) the properties for **Project-specific settings**.
6. Configure Users (on page 50) and user groups.
7. Create required functions (on page 54) and connect them with buttons or events.
8. This method must be acknowledged.

### BEHAVIOR IN RUNTIME

1. A function calls up the dispatch method.
2. A call is initiated via the modem.
3. The connection is considered established as soon as the call to the counterparty is signaled.  
The description is played back from this point. The greeting can be repeated using the #- key.  
Then the PIN must be entered. Only then is the text for the confirmation played back.
4. 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 an entry by pressing the star key (\*); In this case the description will be played back again

### 7.1.5.3 Voice over IP as audio file

The following must be the case for the sending of a voice message as an audio file via Voice over IP:

- ▶ the sending mode was activated via the property **Voice over IP (Audio file)**
- ▶ Voice over IP is configured in the Editor
- ▶ One audio file each must be saved in the Files/Multimedia node for:
  - ▶ Description
  - ▶ Acknowledgement
  - ▶ Misentry
- ▶ The telephone of the recipient must support DTMF
- ▶ The VoIP provider supports the **SIP** and **RTP** protocols
- ▶ The firewall has the corresponding ports open
  - ▶ SIP (Default: 5060)
  - ▶ RTP (Default: 4000)
  - ▶ RTCP (Default: 4001)

## PROCEDURE

### CONFIGURATION

1. Activate the sending mode via the property **Voice over IP (Audio file)**.
2. Configure the settings of the **Voice over IP** group:
  - ▶ **Server address**
  - ▶ **User name**
  - ▶ **Password**
3. Configure (on page 44) the properties for **Project-specific settings**.
4. Configure Users (on page 50) and user groups.
5. Create required functions (on page 54) and connect them with buttons or events.
6. This method must be acknowledged.

## BEHAVIOR IN RUNTIME

1. A function calls up the dispatch method.
2. A call is initiated via the VoIP server.
3. The connection is considered established as soon as the call to the counterparty is received. The description is played back from this point. The greeting can be repeated using the # button.
4. The description must be acknowledged (on page 96) by the recipient with the respective 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 an input by pressing the star key (\*)

The call is not ended automatically. It must be ended by the recipient.

Only once the line has been closed is the status of the message evaluated (confirmed, rejected or missing). As long as the line is open, it is possible to start over by pressing the \* key and entering the code again. PIN can thus be changed to NA, for example.

### Note:

- ▶ If the call is not accepted, the substitute person is called up if one has been configured. The call is canceled otherwise.
- ▶ The name of the audio file is shown in the list of messages in the **Message** column.

## 7.1.5.4 Voice over IP as text-to-speech

The following must be the case for sending the message via Text-to-Speech:

- ▶ the sending mode was activated via the property **Voice over IP (Text-to-Speech)**
- ▶ Voice over IP 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
- ▶ The VoIP provider supports the **SIP** and **RTP** protocols
- ▶ The firewall has the corresponding ports open
  - ▶ SIP (Default: 5060)
  - ▶ RTP (Default: 4000)

- ▶ RTCP (Default: 4001)

## 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 via the property **Voice over IP (Text-to-Speech)**.
2. Configure the properties of the **Voice over IP** group.
  - ▶ **Server address**
  - ▶ **User name**
  - ▶ **Password**
3. Configure the properties for text-to-speech:
  - ▶ **Voice**
  - ▶ **Speech rate**
  - ▶ **Volume**
4. Configure (on page 44) the properties for **Project-specific settings**.
5. Configure Users (on page 50) and user groups.
6. Create required functions (on page 54) and connect them with buttons or events.
7. This method must be acknowledged.

### BEHAVIOR IN RUNTIME

1. A function calls up the dispatch method.

2. A call is initiated via the VoIP server.
3. The connection is considered established as soon as the call to the counterparty is received. The description is played back from this point. The greeting can be repeated using the # button.
4. The description must be acknowledged or rejected by the recipient with the respective 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 an entry by pressing the star key (\*); In this case the description will be played back again

**Note:**

- ▶ If the call is not accepted, the substitute person is called up if one has been configured. The call is canceled otherwise.
- ▶ The message text is shown in the list of messages in the **Message** column.

## 7.1.6 zenon6.ini entries

Message Control is mainly configured via global (on page 15) and project-specific (on page 44) properties. In the **zenon6.ini** the global properties of the media are displayed in section **[Message Control]**.

**Note:** Changes should always be made using the properties of the zenon user interface.

Meaning of the INI entries:

Entry	Description
<b>[MESSAGE CONTROL]</b>	<p>Settings for the module Message Control.</p> <p>Recommendation: Configuration via the properties of the <b>Message Control</b> group in the workspace including subgroups and the properties of the <b>Project-specific settings</b> group in the <b>Message Control</b> node in the project tree.</p>
<b>GSM_SMS=</b>	<p>Activation of SMS via GMS as sending type.</p> <ul style="list-style-type: none"> <li>▶ 0: active</li> <li>▶ not 0: inactive</li> </ul>

Entry	Description
	<p>Default: 0</p> <p>The entry in Message32.ini corresponds to the entry <b>[GSM]</b> <b>On</b></p> <p>This is taken into account when importing/exporting the ini settings.</p> <p>It corresponds to the property <b>Sending mode active</b> in group <b>SMS message (GSM modem)</b> in the Editor.</p>
<b>GSM_SMS_COM=</b>	<p>COM port that is used for the connection to the modem.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the <b>Modem connection (serial)</b> property in the Editor.</p>
<b>GSM_SMS_PIN=</b>	<p>PIN code which is used for authentication towards the modem.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the <b>PIN code</b> property in the Editor.</p>
<b>GSM_SMS_SMSC=</b>	<p>Telephone number of the message center of the GSM provider.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the <b>Number of SMS center</b> property in the Editor.</p>
<b>GSM_BULK_DELETE=</b>	<p>Behavior when deleting an SMS:</p> <ul style="list-style-type: none"> <li>▶ 0: Messages are deleted with the <b>Index</b> method. SMSs that have been read are deleted individually</li> <li>▶ 1: Messages are deleted with the <b>Statusflag</b> method. All SMSs that have been read are deleted at the same time.</li> </ul> <p><b>Note:</b> This method is faster and more powerful, but is not supported by all modems.</p> <p>Default: 0</p> <p>It corresponds to the <b>Delete SMS with status flag syntax of AT+CMGD command</b> property in the Editor.</p>



Entry	Description
<b>Outlook=</b>	<p>Email notification via Outlook:</p> <ul style="list-style-type: none"> <li>▶ 0: inactive</li> <li>▶ 1: active</li> </ul> <p>Default: 0</p> <p>It corresponds to the property <b>Sending mode active</b> in group <b>Mail message (Outlook)</b> in the Editor.</p>
<b>Outlook_Profile=</b>	<p>Name of the Outlook profile which should be used for sending.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the <b>Profile</b> property in the Editor.</p>
<b>POP_APOP=</b>	<p>Controls authentication at the incoming mail server with messaging via SMTP/POP.</p> <ul style="list-style-type: none"> <li>▶ 0: User (<b>USER</b>) and password (<b>PASS</b>) are used for authentication.</li> <li>▶ 1: The <b>APOP</b> command is used for authentication instead of user and password.</li> </ul> <p>Default: 0</p> <p>It corresponds to the property <b>Use APOP for authentication</b> in group <b>Mail message (SMTP)</b> in the Editor.</p>
<b>POP_KEEP_MAILS=</b>	<p>Defines whether e-mails that have been read remain on the server:</p> <ul style="list-style-type: none"> <li>▶ 0: E-mails are deleted after they have been fetched from the server,</li> <li>▶ 1: E-mails remain after they have been fetched from the server.</li> </ul> <p>Default: 0</p> <p><b>Note:</b> As POP3 offers no possibility to filter mails when fetching them, always all mails which are in the incoming mail are fetched. If this property is active, the server's memory consumption increases and the performance can decrease when fetching mails.</p>

Entry	Description
	It corresponds to the <b>Keep read mail on server</b> property in the Editor.
<b>POP_PASSWORD=</b>	<p>Password for login at the incoming mail server (POP3). The password is saved encrypted and is only decrypted for authentication purposes.</p> <p>Must only be configured in the Editor.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the property <b>Password</b> in group <b>Incoming mail server</b> in the Editor.</p>
<b>POP_POLL_INTERVALL=</b>	<p>Minimum period between two POP3 requests in seconds. Value should not be below the POP3 server guideline.</p> <ul style="list-style-type: none"> <li>▶ Minimum: <i>10</i></li> <li>▶ Maximum: <i>4294967295</i></li> </ul> <p>Default: <i>60</i></p> <p>It corresponds to the <b>Minimum time between two requests (s)</b> property in the Editor.</p>
<b>POP_PORT=</b>	<p>Port in which the POP3 server is addressed.</p> <ul style="list-style-type: none"> <li>▶ Maximum value: <i>6553</i></li> </ul> <p>Default for:</p> <ul style="list-style-type: none"> <li>▶ Insecure connection: <i>110</i></li> <li>▶ Secure connection: <i>995</i></li> </ul> <p>It corresponds to the property <b>Port</b> in group <b>Incoming mail server</b> in the Editor.</p>
<b>POP_SECURITY=</b>	<p>Type of connection protection to the POP3 Server.</p> <ul style="list-style-type: none"> <li>▶ <i>0</i>, no security</li> <li>▶ <i>1</i>: SSLv2 and SSLv3</li> <li>▶ <i>2</i>: TLSv1</li> </ul> <p>Default: <i>0</i></p> <p>It corresponds to the property <b>Encryption</b> in group <b>Incoming mail server</b> in the Editor.</p>

Entry	Description
<b>POP_SERVER=</b>	<p>Address of the POP3 server.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the property <b>Server address</b> in group <b>Incoming mail server</b> in the Editor.</p>
<b>POP_USER=</b>	<p>User name for the incoming server.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the property <b>User name</b> in group <b>Incoming mail server</b> in the Editor.</p>
<b>SMSGateway=</b>	<p>SMS notification via SMS gateway:</p> <ul style="list-style-type: none"> <li>▶ 0: inactive</li> <li>▶ 1: active</li> </ul> <p>Default: 0</p> <p>It corresponds to the property <b>Sending mode active</b> in group <b>SMS message (SMS gateway)</b> in the Editor.</p>
<b>SMSGateway_Inbox=</b>	<p>Incoming folder of the SMS server for sending SMS.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the <b>Inbox folder</b> property in the Editor.</p>
<b>SMSGateway_OriginId=</b>	<p>Sender identification for sending SMS.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the <b>Sender ID</b> property in the Editor.</p>
<b>SMSGateway_Outbox=</b>	<p>Outgoing folder of the SMS server for sending SMS.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the <b>Outbox folder</b> property in the Editor.</p>
<b>SMSGateway_Prefix=</b>	<p>First letter of the SMS files. Must be unique for each project.</p> <p>Length: 1 characters</p> <p>Default: <i>F</i></p>

Entry	Description
	It corresponds to the <b>First letter of the SMS file</b> property in the Editor.
<b>SMSGateway_SemaphorPrefix=</b>	<p>First letter of the lock file.</p> <p>Length: 1 characters</p> <p>Default:S</p> <p>It corresponds to the <b>First letter of the lock/semaphore file</b> property in the Editor.</p>
<b>SMSGateway_SmartAlarm=</b>	<p>Smart alarming is used. Must be unique for each project.</p> <ul style="list-style-type: none"> <li>▶ 0: inactive</li> <li>▶ 1: active</li> </ul> <p>Default: 0</p> <p>It corresponds to the <b>Smart alarming</b> property in the Editor.</p>
<b>SMSGateway_Statusbox=</b>	<p>Folder for the status message at SMS sending.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the <b>Status folder</b> property in the Editor.</p>
<b>SMSGateway_TimeOut=</b>	<p>Timeout for outgoing messages in minutes. Defines after what period of time a message is interpreted as "not send successfully".</p> <p>Default: 60</p> <p>It corresponds to the <b>Timeout</b> property in the Editor.</p>
<b>SMTP_AUTH=</b>	<p>Authentication at the outgoing server.</p> <ul style="list-style-type: none"> <li>▶ 0, no security</li> <li>▶ 1: log in to the POP3 Server before sending</li> <li>▶ 2: SMTP AUTH with signing in to the SMTP server before dispatch</li> </ul> <p>Default: 0</p> <p>It corresponds to the property <b>Authentication at outgoing mail server</b> in group <b>Outgoing mail server</b> in the Editor.</p>

Entry	Description
<b>SMTP_OTHER_CREDS=</b>	<p>Defines whether the outgoing server uses different log in data than the incoming server.</p> <ul style="list-style-type: none"> <li>▶ 0: inactive</li> <li>▶ 1: active</li> </ul> <p>Default: 0</p> <p>Corresponds to the property in the <b>Outgoing mail server</b> group in the Editor.</p>
<b>SMTP_PASSWORD=</b>	<p>Hex dump of the encrypted password for authentication at the outgoing server.</p> <p>Default: <i>empty</i></p> <p>Corresponds to the property in the <b>Outgoing mail server</b> group in the Editor.</p>
<b>SMTP_OTHER_CREDS=</b>	<p>Sign-in data for SMTP server.</p> <ul style="list-style-type: none"> <li>▶ 0: Login data for the incoming mail server is also used for the outgoing mail server.</li> <li>▶ 1: For the authentication at the outgoing mail server different login data than at the incoming mail server are used.</li> </ul> <p>Default: 0</p> <p>It corresponds to the property <b>Use different log in data as at the incoming mail server</b> in group <b>Outgoing mail server</b> in the Editor.</p>
<b>SMTP_OUT_ADDR=</b>	<p>Address for outgoing e-mails.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the property <b>Address for sent mails</b> in group <b>Outgoing mail server</b> in the Editor.</p>
<b>SMTP_POP_MAIL=</b>	<p>Email notification via SMTP/POP:</p> <ul style="list-style-type: none"> <li>▶ 1: active</li> <li>▶ 0: inactive</li> </ul> <p>Default: 0</p> <p>It corresponds to the property <b>Sending mode active</b> in</p>

Entry	Description
	group <b>Outgoing mail server</b> in the Editor.
<b>SMTP_PORT=</b>	<p>Defines the used port at the SMTP Server.</p> <p>Maximum: 65535</p> <p>Default: 25</p> <p>It corresponds to the property <b>Port</b> in group <b>Outgoing mail server</b> in the Editor.</p>
<b>SMTP_SECURITY=</b>	<p>Type of encryption for connection to the SMTP Server.</p> <ul style="list-style-type: none"> <li>▶ 0, no security</li> <li>▶ 1: SSLv2 and SSLv3</li> <li>▶ 2: TLSv1</li> </ul> <p>Default: 0</p> <p>It corresponds to the property <b>Encryption</b> in group <b>Outgoing mail server</b> in the Editor.</p>
<b>SMTP_SERVER=</b>	<p>The SMTP Server entered by the user.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the property <b>Server address</b> in group <b>Outgoing mail server</b> in the Editor.</p>
<b>SMTP_SRV_IS_POP=</b>	<p>This entry defines whether the POP3 Server is used as SMTP Server.</p> <ul style="list-style-type: none"> <li>▶ 1: POP 3 is SMTP</li> <li>▶ 0: POP3 and SMTP are different Servers</li> </ul> <p>Default: 0</p> <p>It corresponds to the property <b>Use incoming mail server for outgoing mails</b> in group <b>Outgoing mail server</b> in the Editor.</p>
<b>SMTP_SUBJECT=</b>	<p>Subject for outgoing e-mails and for detecting whether an incoming e-mail at the server is relevant for the sending type.</p> <p>Incoming e-mails must have this text as subject in order to be processed by the system. E-mails which do not contain this subject are neither passed on to Message Control nor</p>

Entry	Description
	<p>deleted from the server.</p> <p>Default: <b>MsgCtrl_Alert</b>:</p> <p>It corresponds to the property <b>Mail subject</b> in group <b>Outgoing mail server</b> in the Editor.</p>
<b>SMTP_USER=</b>	<p>User name saved for the outgoing server.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the property <b>User name</b> in group <b>Outgoing mail server</b> in the Editor.</p>
<b>SMTP_USER_IS_ADDR=</b>	<p>Defines whether the user name for authentication towards the outgoing server is used as sender address for outgoing mails. Is only used if the <b>SMTP_AUTH=</b> entry is not 0.</p> <ul style="list-style-type: none"> <li>▶ 1: active</li> <li>▶ 0: inactive</li> </ul> <p>Default: 0</p> <p>It corresponds to the property <b>Use user name as address for sent mails</b> in group <b>Outgoing mail server</b> in the Editor.</p>
<b>Speech=</b>	<p>Text-to-Speech notification via modem:</p> <ul style="list-style-type: none"> <li>▶ 0: inactive</li> <li>▶ 1: active</li> </ul> <p>Default: 0</p> <p>It corresponds to the property <b>Voice modem (Text-to-Speech)</b> in group <b>Voice message</b> in the Editor.</p>
<b>Speech_Name=</b>	<p>Selection of speech and language for text-to-speech.</p> <p>Ensure that the correct voices for the Editor have been selected:</p> <ul style="list-style-type: none"> <li>▶ 32-bit Editor: C:\Windows\SysWOW64\Speech\SpeechUX\sapi.cpl</li> <li>▶ 64-bit Editor: C:\Windows\System32\Speech\SpeechUX\sapi.cpl</li> </ul>

Entry	Description
	<p>Default: <i>empty</i></p> <p>It corresponds to the property <b>Voice</b> in group <b>Text-to-Speech</b> in the Editor.</p>
<b>Speech_Rate=</b>	<p>Speech speed.</p> <ul style="list-style-type: none"> <li>▶ Minimum: <i>-10</i></li> <li>▶ Maximum: <i>10</i></li> </ul> <p>Default: <i>0</i></p> <p>It corresponds to the property <b>Speech rate</b> in group <b>Text-to-Speech</b> in the Editor.</p>
<b>Speech_Volume=</b>	<p>Speech volume. Number equals the percent value of the maximum value of the selected speech.</p> <ul style="list-style-type: none"> <li>▶ Maximum: <i>100</i></li> <li>▶ Minimum: <i>0</i> Number taken from the operating system without change.</li> </ul> <p>Default: <i>0</i></p> <p>It corresponds to the property <b>Volume</b> in group <b>Text-to-Speech</b> in the Editor.</p>
<b>Subject=</b>	<p>Unique ID which incoming e-mails must contain in order to be processed in Message Control.</p> <p>Default: <b>MsgCtrl_Alert:</b></p> <p>It corresponds to the property <b>Subject (ID)</b> in group <b>Mail message (Outlook)</b> in the Editor.</p>
<b>Telephone=</b>	<p>Notification by means of audio file via modem:</p> <ul style="list-style-type: none"> <li>▶ <i>0</i>: inactive</li> <li>▶ <i>1</i>: active</li> </ul> <p>Default: <i>0</i></p> <p>It corresponds to the property <b>Voice modem (audio file)</b> in group <b>Voice message</b> in the Editor.</p>
<b>Telephone_IgnoreDisconnect=</b>	<p>Behavior in the event of a loss of connection::</p> <ul style="list-style-type: none"> <li>▶ <i>1</i>: A disconnection (e.g. recipient ends call) is ignored and the message is played back</li> </ul>



Entry	Description
	<p>completely before the line is closed.</p> <ul style="list-style-type: none"> <li>▶ 0: Message is aborted when the connection breaks.</li> </ul> <p>Default: 0</p> <p>It corresponds to the property <b>Ignore disconnect</b> in group <b>Voice message</b> in the Editor.</p>
Telephone_Line=	<p>Entry of the modem to be used. Must already be configured on the computer.</p> <p>It corresponds to the property <b>Line name</b> in group <b>Phone settings</b> in the Editor.</p>
Telephone_LineId=	<p>Automatically created device ID which identifies the selected modem. Serves - for several modems with the same name - the purpose of distinguishing the devices.</p> <p><b>Attention:</b> For information only. Must not be changed here.</p> <p>It corresponds to the property <b>Line ID</b> in group <b>Phone settings</b> in the Editor.</p>
Telephone_Timeout=	<p>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.</p> <p>Default: 1</p> <p>It corresponds to the property <b>Timeout [min]</b> in group <b>Voice message</b> in the Editor.</p>
Telephone_WelcomeMessageCount=	<p>Number of repetitions for the welcome text.</p> <p>Default: 5</p> <p>It corresponds to the property <b>Repeat welcome text</b> in group <b>Voice message</b> in the Editor.</p>
VOIP_AUDIO=	<p>Voice message as audio file by means of Voice over IP:</p> <ul style="list-style-type: none"> <li>▶ 0: inactive</li> <li>▶ 1: <b>Voice over IP (Audio file)</b></li> </ul> <p>Default: 0</p>

Entry	Description
	<p>It corresponds to the property <b>Voice over IP (Audio file)</b> in group <b>Voice message</b> in the Editor.</p> <p><b>Note:</b></p> <p>The following conditions must be met in order to use VoIP:</p> <ul style="list-style-type: none"> <li>▶ The VoIP provider must support the <b>SIP</b> and <b>RTP</b> protocols.</li> <li>▶ The corresponding ports must be open in the firewall: SIP (Default: 5060) RTP (Default: 4000) RTCP (Default: 4001)</li> </ul>
<b>VOIP_DOMAIN=</b>	<p>Server address of the VoIP provider.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the property <b>Server address</b> in group <b>Voice over IP</b> in the Editor.</p>
<b>VOIP_PASSWORD=</b>	<p>Password for VoIP access.</p> <p>Is saved in encrypted form and must only be changed in the user interface.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the property <b>Password</b> in group <b>Voice over IP</b> in the Editor.</p>
<b>VOIP_RTP=</b>	<p>Number of the RTP port for VoIP.</p> <p>Default: 4000</p> <p>It corresponds to the property <b>RTP port</b> in group <b>Voice over IP</b> in the Editor.</p>
<b>VOIP_SIP=</b>	<p>Number of the SIP port for VoIP.</p> <p>Default: 5060</p> <p>It corresponds to the property <b>SIP port</b> in group <b>Voice over IP</b> in the Editor.</p>
<b>VOIP_TTS=</b>	Voice message as text-to-speech by means of voice over

Entry	Description
	<p>IP:</p> <ul style="list-style-type: none"> <li>▶ 0: inactive</li> <li>▶ 1: <b>Voice over IP (Text-to-Speech)</b></li> </ul> <p>Default: 0</p> <p>It corresponds to the property <b>Voice over IP (Text-to-Speech)</b> in group <b>Voice message</b> in the Editor.</p> <p><b>Note:</b></p> <p>The following conditions must be met in order to use VoIP:</p> <ul style="list-style-type: none"> <li>▶ The VoIP provider must support the <b>SIP</b> and <b>RTP</b> protocols.</li> <li>▶ The corresponding ports must be open in the firewall: <ul style="list-style-type: none"> <li>SIP (Default: 5060)</li> <li>RTP (Default: 4000)</li> <li>RTCP (Default: 4001)</li> </ul> </li> </ul>
<b>VOIP_USER=</b>	<p>User name for VoIP access.</p> <p>Default: <i>empty</i></p> <p>It corresponds to the property <b>User name</b> in group <b>Voice over IP</b> in the Editor.</p>



### Information

Some properties can accept the values of other properties. You always save the value that was entered last. The value of the ini entries therefore does not always need to correspond to the values of the properties displayed in the Editor. The following properties are affected:

- ▶ **SMTP\_SERVER=** -> **Server address**
- ▶ **SMTP\_USER=** -> **User name**
- ▶ **SMTP\_PASSWORD=** -> **Password**
- ▶ **SMTP\_OUT\_ADDR=** -> **Address for sent mails**

▶

## 7.2 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 the Runtime, the Message Control module 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 85): 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 85) in the 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:
  - ▶ *Confirm all*:  
All messages
  - ▶ *No confirmations*:  
No messages
  - ▶ *Negative confirmations only*:  
Only messages that are rejected or not responded to (including repetitions)
    - ▶ *Positive confirmations only*:  
Only messages that have already 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 **Maximum number of attempts** property.

## CONFIGURATION OF THE MESSAGE

The message 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

The dynamic part of the subject was set up according to the following rules:

- ▶ A parameter (on page 85) can:
  - ▶ Be a key from the language table, such as **@MyText**;
  - ▶ Define a certain variable that is called up on execution of the function by means of AML or a limit value breach by means of its name: for example **%Var1**;  
**Caution:** The variable whose limit value has been breached is not dynamically determined for the subject!
  - ▶ Be a compiled entry in the language table: for example **%@Var2+MultipleText**
- ▶ **\$**: marks text that contains the parameter.
- ▶ **@**: marks language switching
- ▶ **%**: marks variables
- ▶ **%%**: marks **limit value text parameters** (on page 87) for variables  
 A variable can be stated between the two percentage marks.  
 If no variable is given, the value relates to the main variable.
- ▶ Text parts are separated from one another with a semi colon (;).

- ▶ Messages end with a semicolon (;).

Parameter	In the Runtime	Action
key: <b>@MyText</b>	A check is made to see if an entry in @MyText is present in the language table for the current language.	<ul style="list-style-type: none"> <li>▶ <i>Existing</i>: entry is added to the message.</li> <li>▶ <i>non-existent</i>: According to the settings of the project properties <b>Show untranslated keywords</b> either <b>@MyText</b> or <b>MyText</b> is added to the message.</li> </ul>
Variable: <b>%Var1%</b>	A check is made if the variable exists and the value can be read.	<ul style="list-style-type: none"> <li>▶ <i>Existing</i>: Value of the variable is taken and added to the text as string.</li> <li>▶ <i>Not existing/not readable</i>: The text <b>xxx</b> is added to the message.</li> </ul>
Compound entry: <b>@StringTable+%var1Text</b>	A check is made if: <ul style="list-style-type: none"> <li>▶ 1. the variable exists and the value can be read. Value is attached to the prefix text ("MultipleText") as string.</li> <li>▶ 2. an entry in the language table exists for the text</li> </ul>	<ul style="list-style-type: none"> <li>▶ Variable and text exist: text is added to the message. Example: Value of the variable is 33. The language table is then checked for <b>@StringTable33</b> and the corresponding text is added.</li> <li>▶ <i>non-existent</i>: According to the settings of the project properties <b>Show untranslated keywords</b> either <b>@MyText</b> or <b>MyText</b> is added to the message.</li> </ul>

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

## 7.3 Create a screen of type Message Control

The *message control* screen is for managing the message queue of a project in the Runtime.

### ENGINEERING

Two procedures are available to create a screen:

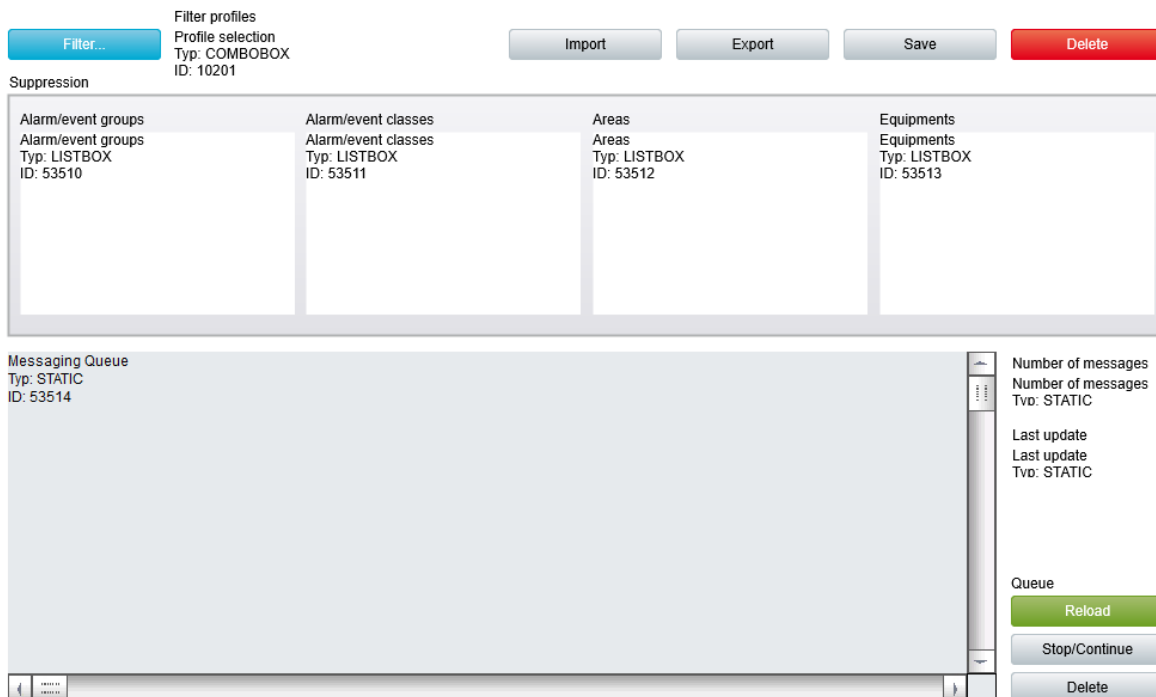
- ▶ The use of the screen creation dialog
- ▶ The creation of a screen using the properties

Steps to create the screen using the properties if the screen creation dialog has been deactivated in the menu bar under **Tools, Settings** and **Use assistant**:

1. Create a new screen.

To do this, select the **New screen** command in the tool bar or in the context menu of the **Screens** node.

2. Change the properties of the screen:
  - a) Name the screen in the **Name** property.
  - b) Select *Message Control* in the **Screen type** property.
  - c) Select the desired frame in the **Frame** property.
3. Configure the content of the screen:
  - a) Select the **Elements (screen type)** menu item from the menu bar.
  - b) Select *Insert template* in the drop-down list.  
The dialog to select pre-defined layouts is opened. Certain control elements are inserted into the screen at predefined positions.
  - c) Remove elements that are not required from the screen.
  - d) If necessary, select additional elements in the **Elements** drop-down list. Place these at the desired position in the screen.
4. Create a screen switch function.



If you want to edit the list directly using the monitor, activate the Multi-Touch functionality. You can find detailed information in relation to this in the Configure interactions chapter.

Longer texts can also be displayed in the Runtime over several lines using the **Automatic word wrap** property.

In the Editor, go to **Representation** in the properties of the respective list properties and activate the checkbox of the **Automatic word wrap** property.

The line height must be amended manually.

## INSERT TEMPLATE

Control elements	Description
<b>Insert template</b>	<p>Opens the dialog for selecting a template for the screen type.</p> <p>Templates are shipped together with zenon and can also be created by the user.</p> <p>Templates add pre-defined control elements to pre-defined position 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 placed in the zenon screen. Elements can be moved on the screen and arranged individually.</p>

## WINDOW

Lists and fields which can be displayed in the Runtime.

Control elements	Description
<b>Message queue</b>	<p>List of messages.</p> <p>Contains messages which have been sent, confirmed or deleted or remain to be sent.</p> <p>The messages are only displayed and cannot be edited anymore.</p> <p>Column headings can be named individually (on page 56) and are localizable by putting @ before them.</p>
<b>Number of messages</b>	Number of currently existing messages.
<b>Last Update</b>	Point of time of last update.
<b>Suppressed alarm/event groups</b>	<b>Alarm/event groups</b> for which messages were suppressed and the alarms of which are thus not displayed in the message queue.
<b>Suppressed alarm/event classes</b>	<b>Alarm/event classes</b> for which messages were suppressed and the alarms of which are thus not displayed in the



Control elements	Description
	message queue.
Suppressed alarm/event areas	<b>Alarm/event areas</b> for which messages were suppressed and the alarms of which are thus not displayed in the message queue.
Suppressed pieces of equipment	<b>Equipment</b> for which messages were suppressed and the alarms of which are thus not displayed in the message queue.

## LIST FUNCTIONS

Control elements to control lists.

Control elements	Description
Filter	Opens Dialog (on page 54) to configure the filters for the view.
Delete	Deletes selected entry from the list. It is only possible to delete messages which have not been sent yet.
Refresh	Refreshes view. The current data is loaded onto the screen from the message queue.  The point of time of the last update can be displayed with the field <b>Last update</b> .
Stop/Continue	Starts and stops the cyclic update of the list.

## FILTER PROFILES

Profile administration

Control elements	Description
Profile selection	Opens the dialog for selecting a profile.
Save	Saves current setting as a profile.
Delete	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 configured via the Filter (on page 54) for the function screen switching in the Editor, in the Runtime via the control element **Filter**. The columns of the list can be provided with individual labels and are localizable.

## 7.4 User Administration

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

Users and user groups are referenced by names.



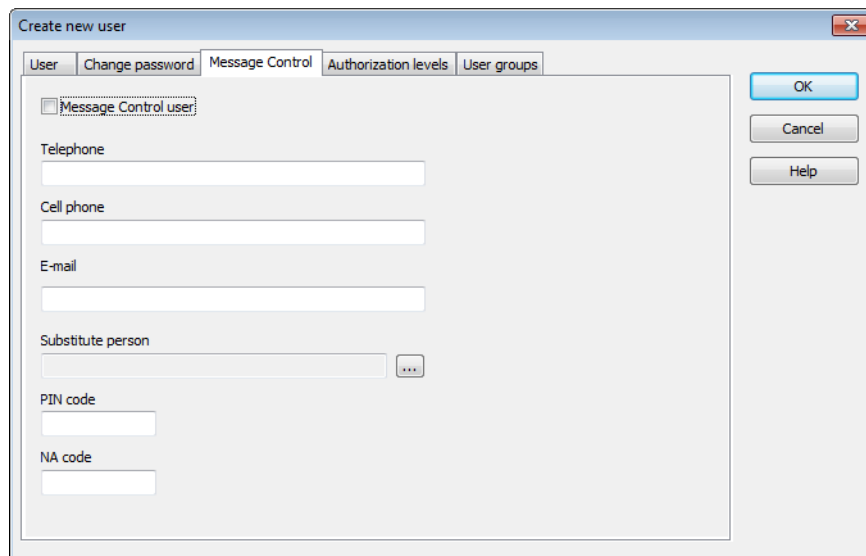
### Attention

Ensure that the required contact data for sending has been configured for each user.

If a user or user group is to be notified in the Runtime and the data required for the sending type is not available for a user, the sending is canceled. Another attempt to send is not made.

For groups, this means: If, for example, the fourth user of a list cannot be configured correctly, the first three users of the list are notified. The process is canceled with the fourth user. They and all subsequent users on the list do not get any more notifications. There is also no attempt to notify the first three users again.

## USER



Parameter	Description
<b>Message Control User</b>	<i>Active:</i> The user is used by the module Message Control.
<b>Telephone</b>	<p>Number of the voice-compatible telephone device of the user. Used for text to speech.</p> <p>Enter numbers. In addition, the following are permitted:</p> <ul style="list-style-type: none"> <li>▶ The prefix + as an abbreviation for 00 of the international area code is permitted.</li> <li>▶ The following separators are also permitted in AD user administration: Minus (-), slash (/) and <b>space</b></li> </ul> <p>Note: When communicating between AD and Message Control, separators are ignored as soon as the data from the AD is mapped to a zenon object.</p>
<b>Cell phone</b>	<p>Cellphone number of the user. Used for messages via GSM and SMS (text messages).</p> <p>Enter numbers. In addition, the following are permitted:</p> <ul style="list-style-type: none"> <li>▶ The prefix + as an abbreviation for 00 of the international area code is permitted.</li> <li>▶ The following separators are also permitted in AD user administration: Minus (-), slash (/) and <b>space</b></li> </ul> <p>Note: When communicating between AD and Message Control, separators are ignored as soon as the data from the AD is mapped to a zenon object.</p>

Parameter	Description
Email	E-mail address of the user
Substitute person	If a user has not been reached or they do not accept the message, a substitute person can be given. Click the ... button and the dialog opens to select an user. Only users who have been activated as <b>Message Control</b> users are offered for selection.
PIN code	PIN code with which the user confirms the receipt of the message. The code consists of a four-digit number between 0000 and 9999.
NA code	<p>PIN code with which the user rejects the receipt of the message (not available). The message is then sent to the next user in the list.</p> <p>If there is no other user entered in the list, the message is entered as <i>"not successfully acknowledged"</i>. The function assigned to this is executed. In addition, a <i>"rejected by"</i> CEL entry is created in each case.</p> <p>The code consists of a four-digit number between 0000 and 9999.</p> <p><b>Note:</b> You can find further information on the assignment of functions in the Confirmation of receipt - confirmation of receipt settings (on page 82) chapter.</p>

## CLOSE DIALOG

Option	Description
OK	<b>Applies all changes in all tabs and closes the dialog.</b>
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.



### Attention

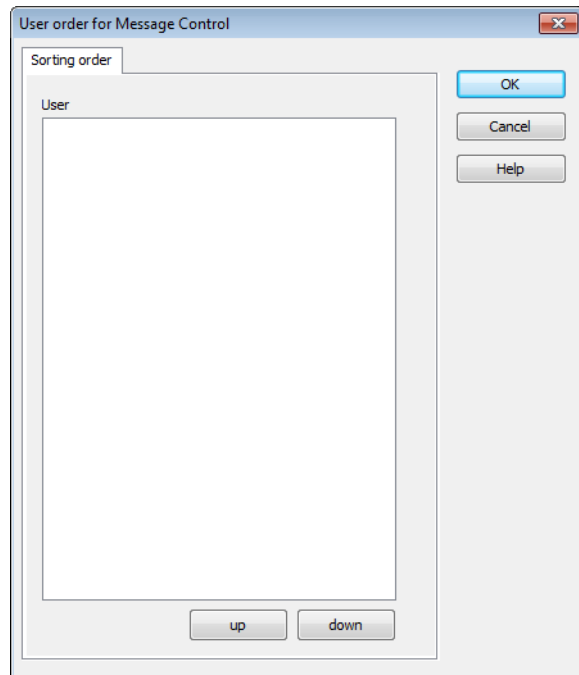
The acknowledgment codes for PIN (confirmation) and NA (rejection) must differ and should not be too similar.

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

If an unknown code is received, a 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.



Parameter	Description
<b>User</b>	List of all available users.
<b>Up</b>	Moves selected user up one place.
<b>Down</b>	Moves selected user down one place.
<b>OK</b>	Applies settings and closes the dialog.
<b>Cancel</b>	Discards all changes and closes the dialog.
<b>Help</b>	Opens online help.

In the 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.

## 7.5 Functions

Functions control the use of Message Control in the Runtime.

With this:

- ▶ Message control is activated (on page 85) and deactivated (on page 85)
- ▶ messages are sent (on page 64)
- ▶ The Message Control screen will be displayed (on page 54) in order to manage the message queue
- ▶ the message queue is saved (on page 58)
- ▶ elements for messages are disabled (on page 59)

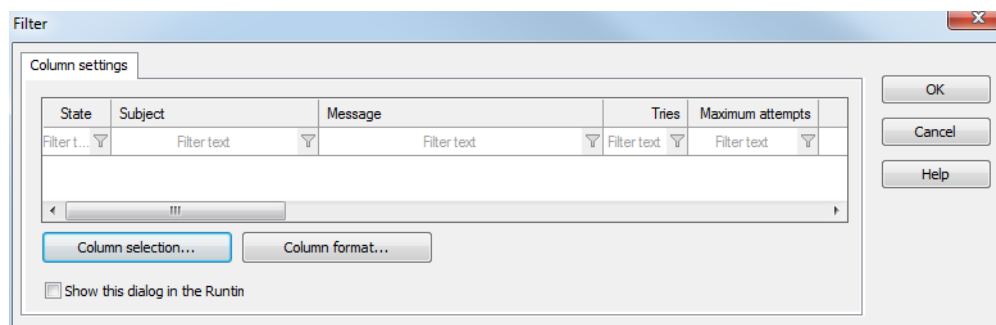
### 7.5.1 Screen switching to a screen of type Message Control

With a switch to a Message Control screen, you call up the screen in Runtime in order to monitor 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 46)
5. the dialog for configuring the column settings (on page 54) 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 in the Runtime

#### 7.5.1.1 Column settings

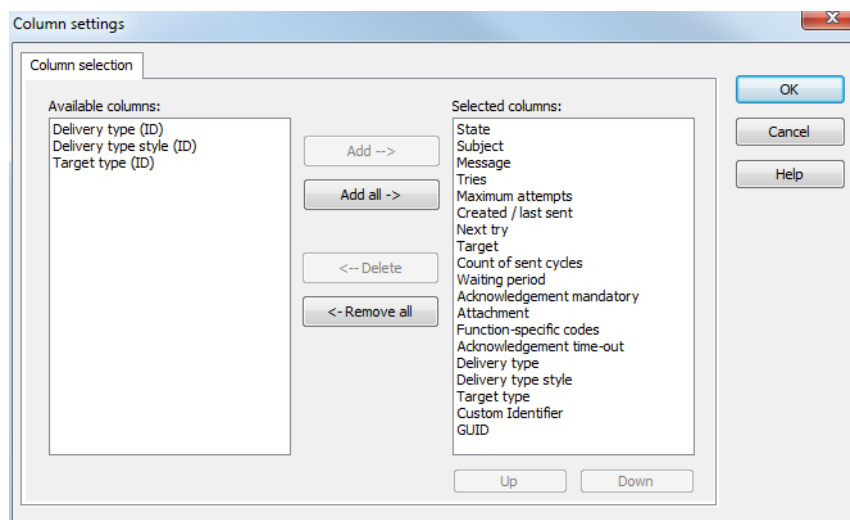
Settings for filtering messages



Parameter	Description
List field	Display of the configured columns.
Column selection	Opens the dialog for selecting the 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.

### 7.5.1.1.1 Column selection

Definition of the columns displayed in the Runtime.



Option	Function
Available columns	List of columns that can be displayed in the table.
Selected columns	Columns that are displayed in the table.
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

Option	Function
	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.

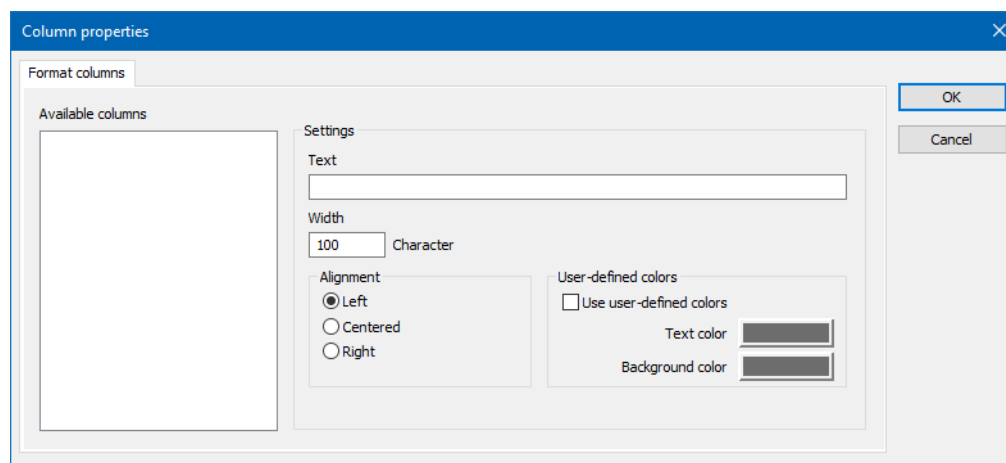
## CLOSE DIALOG

Options	Description
OK	<b>Applies settings and closes the dialog.</b>
Cancel	Discards all changes and closes the dialog.
Help	Opens online help.

### 7.5.1.1.2 Column format

Formatting of columns.

Configuration of the properties of the columns for configurable lists. The settings have an effect on the respective list in the Editor or - when configuring screen switching - in Runtime.





## AVAILABLE COLUMNS

Option	Description
Available columns	List of the available columns via <b>Column selection</b> . The highlighted column is configured via the options in the <b>Settings</b> area.

## SETTINGS

Option	Description
Settings	Settings for selected column.
Labeling	<p>Name for column title.</p> <p>The column title is online language switchable. To do this, the @ character must be entered in front of the name.</p>
Width	<p>Width of the column in characters.</p> <p>Calculation: Number time average character width of the selected font.</p>
Alignment	<p>Alignment. Selection by means of radio buttons.</p> <p>Possible settings:</p> <ul style="list-style-type: none"> <li>▶ <b>Left:</b> Text is justified on the left edge of the column.</li> <li>▶ <b>Centered:</b> Text is displayed centered in the column.</li> <li>▶ <b>Right:</b> Text is justified on the right edge of the column.</li> </ul>
User-defined colors	<p>Properties in order to define user-defined colors for text and background. The settings have an effect on the Editor and Runtime.</p> <p><b>Note:</b></p> <ul style="list-style-type: none"> <li>▶ These settings are only available for configurable lists.</li> <li>▶ In addition, the respective focus in the list can be signaled in the Runtime by means of different text and background colors. These are configured using the project properties.</li> </ul>
User defined colors	<i>Active:</i> User-defined colors are used.

Option	Description
<b>Text color</b>	Color for text display. Clicking on the color opens the color palette to select a color.
<b>Background color</b>	Color for the display of the cell background. Clicking on the color opens the color palette to select a color.
<b>Lock column filter in the Runtime</b>	<ul style="list-style-type: none"> <li>▶ <i>Active:</i> The filter for this column cannot be changed in the Runtime.</li> </ul> <p><b>Note:</b> Only available for:</p> <ul style="list-style-type: none"> <li>▶ Batch Control</li> <li>▶ Extended Trend</li> <li>▶ Filter screens</li> <li>▶ Message Control</li> <li>▶ Recipegroup Manager</li> <li>▶ Shift Management</li> <li>▶ Context List</li> </ul>

#### CLOSE DIALOG

Option	Description
<b>OK</b>	<b>Applies all changes in all tabs and closes the dialog.</b>
<b>Cancel</b>	Discards all changes in all tabs and closes the dialog.
<b>Help</b>	Opens online help.

### 7.5.2 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 the Runtime is closed. To engineer the function:

1. Create a new function.
2. Select the **Save current queue** function in the **Message Control** group.
3. Link the function with a button in order to be able to access it in the Runtime

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

### 7.5.3 Group/class/area/equipment suppressed

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 engineering takes place in the Editor and can be released for changes in the Runtime.

If a **Limit Values** 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.

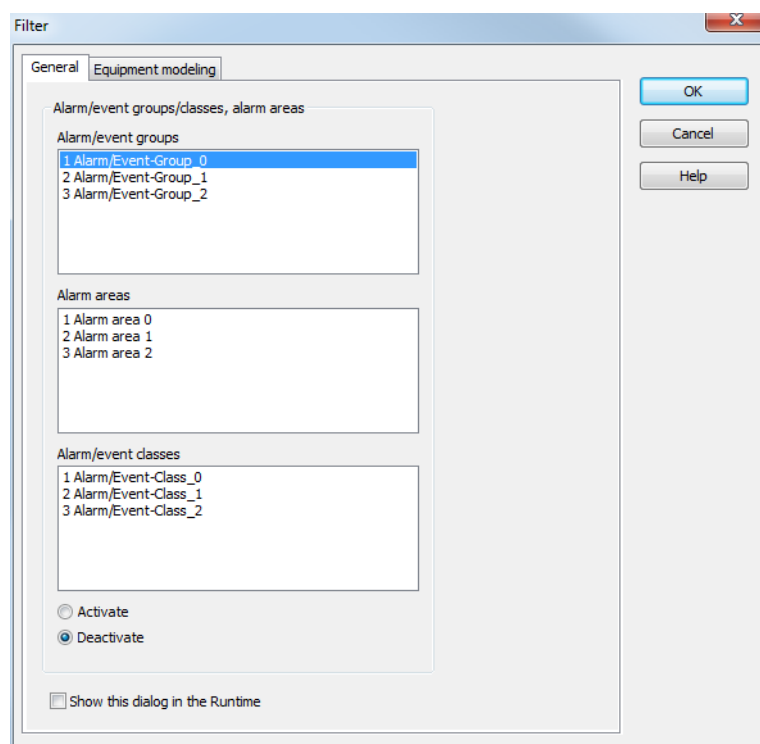
Suppressed entries are logged in the CEL if the property **Log in CEL** is configured with *all confirmations* or *only negative confirmations*.

## ENGINEERING OF A FUNCTION

To configure the function:

1. Create a new function
2. in group **Message Control** select the function **Group/class/area/equipment suppressed**
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 in the Runtime

### 7.5.3.1 General



Parameter	Description
<b>Alarm/event groups</b>	List of configured alarm/event groups.
<b>Alarm Areas</b>	List of configured alarm areas.
<b>Alarm/event classes</b>	List of configured alarm/event classes.
<b>Activate</b>	<i>Active:</i> When activating the function the selected elements are added to the list of suppressed elements.
<b>Deactivate</b>	<i>Active:</i> When activating the function the selected elements are removed from the list of suppressed elements.
<b>Show dialog in the Runtime</b>	<p>Checkbox to select whether this dialog is shown in Runtime:</p> <ul style="list-style-type: none"> <li>▶ <i>active:</i> This dialog is called up during operation in the Runtime on the current computer. In the network, this dialog is called up on the computer that executes the function. As a result, changes to existing parameter settings of an zenon Editor configuration are possible during execution in zenon Runtime.</li> <li>▶ <i>Inactive:</i> This dialog is not shown in the Runtime during operation. The function or the command is immediately executed with the project configuration</li> </ul>

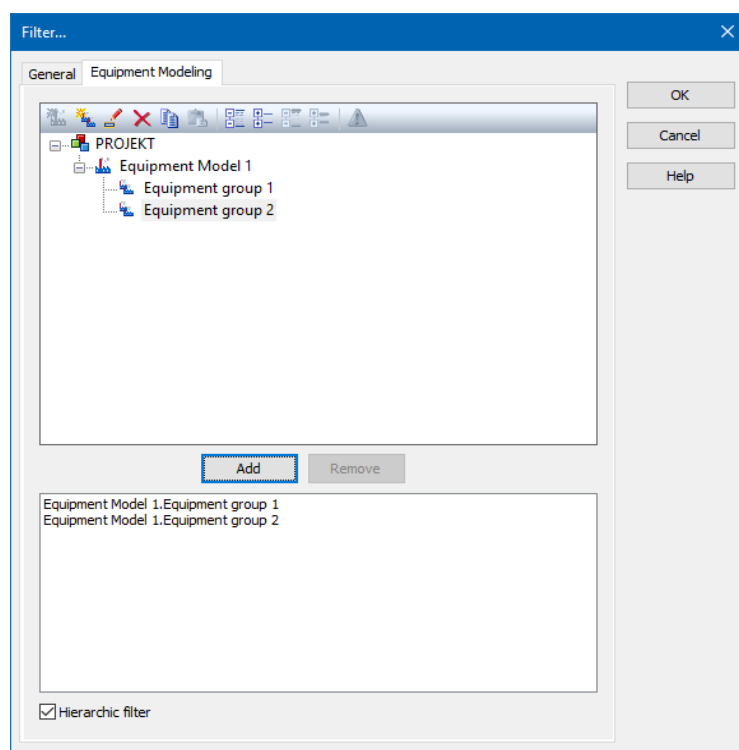
Parameter	Description
	created in the Editor.

## CLOSE DIALOG

Option	Description
OK	<b>Applies all changes in all tabs and closes the dialog.</b>
Cancel	Discards all changes in all tabs and closes the 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.

## 7.5.3.2 Equipment Modeling



Option	Description
Toolbar	Symbols to: <ul style="list-style-type: none"> <li>▶ Edit local equipment models</li> <li>▶ Expand or collapse the display</li> </ul>

Option	Description
	<ul style="list-style-type: none"> <li>► Display of information</li> </ul>
<b>List of equipment models</b>	<p>provides models and groups for selection The list separates the display into equipment models from the global project and from local projects.</p> <p>Local equipment models can be created, edited or deleted.</p> <p><b>Note:</b> Equipment models from the global project cannot be displayed if there are models with the same name from the local project. Affected models are displayed by clicking on the warning symbol (triangle with exclamation mark). For details, see the <b>Equipment modeling</b> manual, <b>Editing local equipment models</b> chapter.</p>
<b>Add</b>	Adds the selected groups to the filter list.
<b>Remove</b>	Removes all selected groups from the filter list.
<b>Filter list</b>	Shows all equipment groups that are to be filtered.
<b>Hierarchic filter</b>	<p>Checkbox for the activation of the hierarchical filtering of the equipment model</p> <ul style="list-style-type: none"> <li>► <i>active</i>: Variables that are linked to a subhierarchy of the selected equipment group are taken into account when filtering and are contained in the display in the Runtime.</li> <li>► <i>Inactive</i>: When filtering, only variables that are linked to the selected equipment group are taken into account. Default: <i>active</i></li> </ul>

## CLOSE DIALOG

Option	Description
<b>OK</b>	Applies settings and closes the dialog.
<b>Cancel</b>	<p>Discards the selection and closes the dialog.</p> <p><b>Attention:</b> Any changes that have been made to the structure of local equipment models are retained.</p>
<b>Help</b>	Opens online help.

## CONFIGURATION

New models and groups can be created for the active project and existing ones can be deleted.

### ADD MODEL TO PROJECT

To add a new model:

1. Click on the project.
2. In the toolbar select **New Equipment model**

### REMOVE MODEL

To remove an existing model:

1. Click on the model.
2. Select **Remove** in the toolbar.

### ADD GROUPS TO THE MODEL

To add a group to a model:

1. 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 toolbar.
2. Select an equipment group or level.
3. Add the new group to the list in the lower area of the dialog with the **Add** button.

Note:

- ▶ Subgroups are not automatically added.
- ▶ it is possible to link as many groups as you want.

### REMOVE GROUP FROM THE MODEL

To delete a group from a model:

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

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

### 7.5.4 Send a Message

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

- ▶ an alarm (function for group, class or area)
- ▶ a limit value (property **Limit Values/Execute function instantly**),
- ▶ a response matrix (function)
- ▶ a time control (via button or PFS)



#### Attention

Ensure that the required contact data for sending has been configured for each user.

If a user or user group is to be notified in the Runtime and the data required for the sending type is not available for a user, the sending is canceled. Another attempt to send is not made.

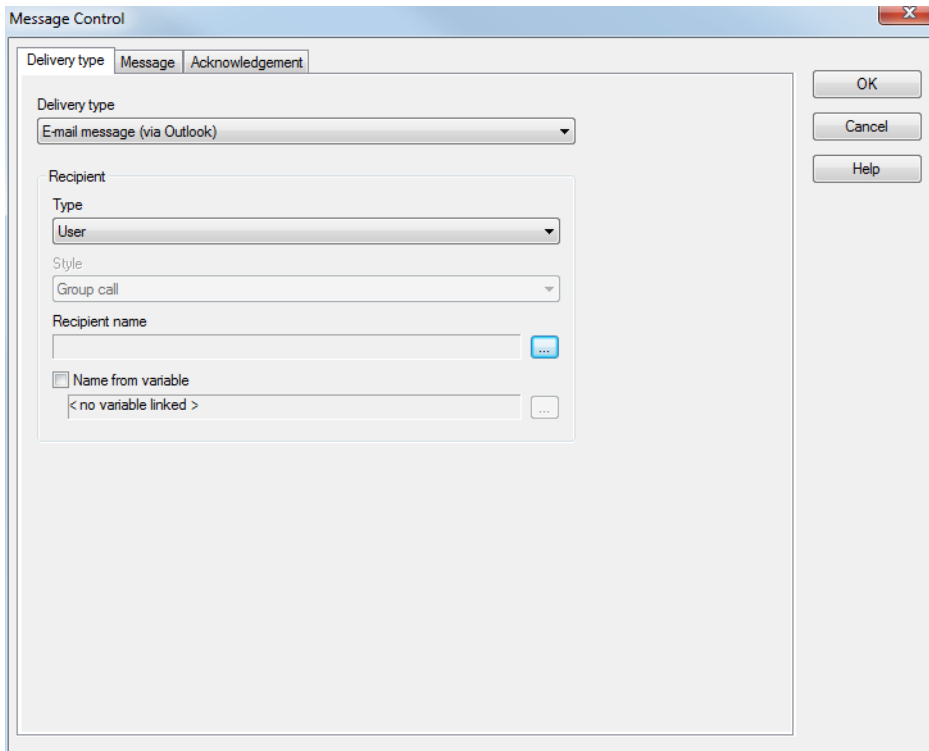
For groups, this means: If, for example, the fourth user of a list cannot be configured correctly, the first three users of the list are notified. The process is canceled with the fourth user. They and all subsequent users on the list do not get any more notifications. There is also no attempt to notify the first three users again.

To configure the function:

1. Create a new function.
2. Select the **Send a Message** function in the **Message Control** group.



3. The dialog for configuration is opened:



4. Configure
  - ▶ Sending mode (on page 66)
  - ▶ Message (on page 71)
  - ▶ Acknowledgment of receipt (on page 82)

5. Close the dialog by clicking on **OK**.

**Note:** The entries are validated in the tabs by clicking on **OK**. If incorrect configurations are discovered, the user is informed by means of a message box. The user can then change the configuration or continue with the saving of the incorrect configuration.

6. Link the function.

For linking with a limit value or reaction matrix, content that is dynamically generated in the Runtime can be sent. In order to use several **Send a Message** functions for a limit value, use a script that contains these functions and is linked to the limit value.

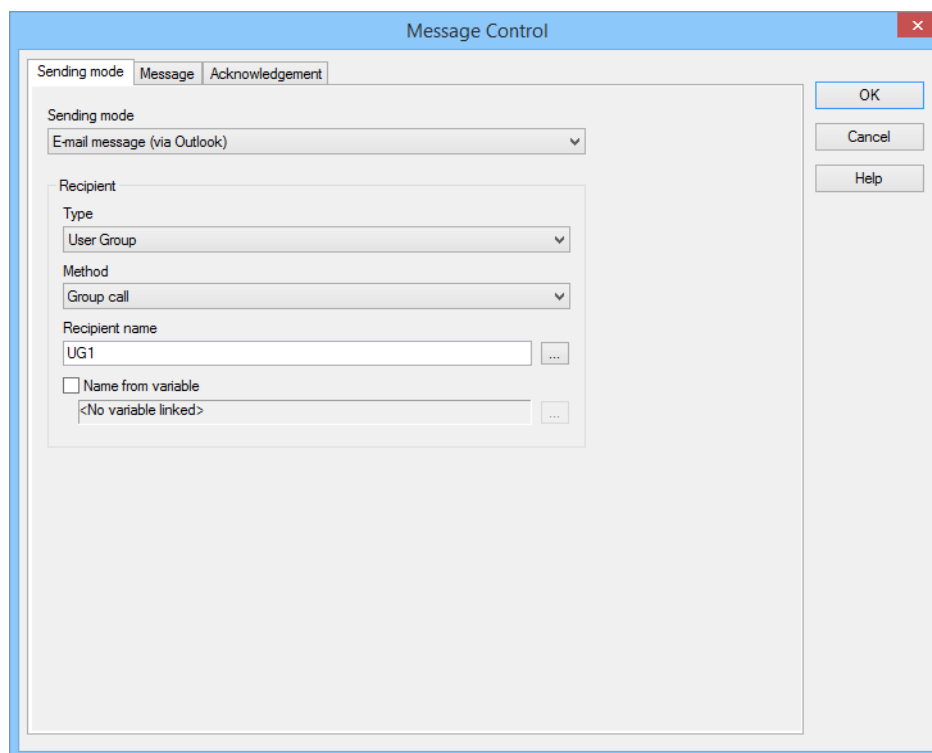


### Information

In the Runtime the function **Send a Message** is always performed at the computer executing the process. This is the computer that executes the function in Runtime.

### 7.5.4.1 Sending mode

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



Parameter	Description
<b>Sending mode</b>	<p>Selection of the sending mode from the drop-down list: The following are available depending on configuration:</p> <ul style="list-style-type: none"> <li>▶ E-mail message via Outlook</li> <li>▶ E-mail message via SMTP</li> <li>▶ Voice message audiofile via modem</li> <li>▶ Voice message Text-to-Speech via modem</li> <li>▶ SMS message via SMS server</li> <li>▶ SMS message via GSM modem</li> <li>▶ Voice over IP as audio file</li> <li>▶ Voice over IP configured as Text-to-Speech</li> </ul> <p><b>Note:</b> In the drop-down list, only media whose <b>Sending mode active</b> property has been activated in the Message Control Properties (on page 15) of the workspace are shown.</p>
<b>Recipient</b>	Recipient settings:

Parameter	Description
<b>Type</b>	<p>Configuration of recipient pattern from the drop-down list. In doing so, the dialog to select a user or a user group is opened.</p> <p>The following are available:</p> <ul style="list-style-type: none"> <li>▶ <i>User group:</i> Sending to a group. The <b>Method</b> option stipulates how group members are notified.</li> <li>▶ <i>User:</i> Dispatch to an individual user.</li> <li>▶ <i>User group with shift filter:</i> Selection of a user group for the shift management. Users linked to a shift can be notified. <b>Recommendation:</b> Select <i>infinite successive call</i> as a <b>method</b>.</li> </ul> <p>Configured users or user groups can be amended using the <b>Recipient name</b> option.</p>
<b>Method</b>	<p>Only available if, for the <b>Type</b> <i>User group</i> or <i>User group with shift filter</i> option has been selected.</p> <p>Selection of sending method from drop-down list. Possible methods:</p> <ul style="list-style-type: none"> <li>▶ <i>Group call:</i> All users of the group will receive the message parallel. (Notice: In versions prior to zenon 7.00, this method corresponded to the <b>group sending</b> dispatch type.)</li> <li>▶ <i>Following call:</i> The message is delivered to the users according to the defined sequence (on page 50) 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.</li> <li>▶ <i>Endless following call:</i> In principle corresponds to the <i>Following call</i>, 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.</li> </ul> <p><b>Note:</b> This setting is independent of the <b>Maximum number of attempts</b> property.</p>

Parameter	Description
<b>Recipient name</b>	Only available if the option <b>Name from variable</b> is deactivated.  Opens dialog for selection of a recipient or a recipient group - depending on the selection in the option <b>Type</b> .
<b>Name from variable</b>	<i>Active:</i> The variable is read in Runtime and a user or user group from the user administration is read off by means of its value. Click on button ... in order to open the dialog for selecting a variable.
<b>OK</b>	Applies all changes in all tabs and closes the dialog.  Dialog can only be closed if all required options have been configured.
<b>Cancel</b>	Discards all changes in all tabs and closes the dialog.
<b>Help</b>	Opens online help.

#### 7.5.4.1.1 Send message to user group with shift filter

Users that have been assigned to a shift can be notified of alarms and events automatically using the **Message Control** module.

Notification is configured by means of the two modules **shift management** and **Message Control**.

**Note:** Notification can only be used for configured shifts. Linking to shift models is not possible.

### CONFIGURATION

To notify users:

1. Link the desired user to the shift.
2. Link the user to a **user group**.  
Ensure that, for users who need to be notified:
  - ▶ The **Message Control user** property has been activated
  - ▶ The **Message Control group** property has been activated for the user group
3. Stipulate, for the user group, the sequence in which the users are to be notified. Use the **User order** property for this. The users are notified in the stipulated order.
4. For **Message Control**, configure the **recipient** in the **Send a Message** function:
  - a) Open the function and switch to the **Dispatch Type** (on page 64) tab.
  - b) Select *User group with shift filter* in the **Type** option.  
The **User group selection** dialog is opened.

- c) Select the user group configured for shift management as the recipient of the message.

**Note:** To change the group later, select the **Recipient name** option.

- d) Select a **method**.

**Note:** If *group call* is selected, all recipients are notified at the same time, and consecutively with *following call* and *infinite successive call*.

5. Configure the further options of the function.

## PROCEDURE IN RUNTIME

Procedure when triggering the **Send a Message** function:

1. The function looks, in the set user group, for the first user who is configured for **Message Control**.
2. A check is carried out to see whether this user is included one of the shifts that is currently active.
3. If a user is included in an active shift, they are notified.
4. If the user is not included, they cannot be contacted or they reject the message, the next user in the user group is searched for and compared to the current shift.
5. This process is carried out until a user confirms the message or the group has been gone through in full.

The precise procedure depends on the configured method.

## METHODS FOR USER GROUP WITH SHIFT FILTER

The behavior in the Runtime depends on which method has been selected for the user group with shift filter:

Style	Procedure
<i>Group call:</i>	<ul style="list-style-type: none"> <li>▶ The message goes to all users of the selected type who are linked in a shift for which the function execution is active. All users found are notified at the same time.</li> <li>▶ If a user does not confirm the message, a message is sent to their substitute if they are linked in a shift.</li> </ul>
<i>Following call:</i>	<ul style="list-style-type: none"> <li>▶ An attempt is made to find the first user of the group who is linked to the shift.</li> <li>▶ This user is notified.</li> <li>▶ If they cannot be contacted, the next user is searched for and notified.</li> <li>▶ This is run through once until the end of the group has been</li> </ul>

Style	Procedure
	<p>reached.</p> <ul style="list-style-type: none"> <li>▶ If no shift is still active during the process or no relevant users are linked in the active shifts, the notification is ended. A corresponding entry is made in the CEL.</li> </ul>
<i>Endless following call:</i>	<ul style="list-style-type: none"> <li>▶ An attempt is made to find the first user of the group who is linked to the shift.</li> <li>▶ This user is notified.</li> <li>▶ If they cannot be contacted, the next user is searched for and notified.</li> <li>▶ This is run through cyclically until a user is reached.</li> <li>▶ If no shift is still active during the process or no relevant users are linked in the active shifts, the notification is ended. A corresponding entry is made in the CEL.</li> </ul>

**Note:**

- ▶ Only users who are linked in a shift for which the execution of the function is active are notified.
- ▶ Depending on the configuration, messages can also be sent to users who are no longer active in any shift.  
Example: The type of sending (such as telephone) needs some time for sending. If, at the time of generating the message, there are already further messages in the queue, the new message is sent with a delay. If the shift has been switched in the mean time, the message is nevertheless sent to the user of the previous shift.
- ▶ *Following call* and *Endless following call*: No further messages are created if no user of the selected group is active in the active shift.  
Example: The message sending with endless subsequent call starts during **Shift 1**.  
None of the recipients acknowledges the message. A shift-free time of 5 seconds is configured between **Shift 1** and the subsequent **Shift 2**.  
If the timeout for acknowledgment runs out in precisely these 5 minutes, there is no recipient available at this point in time. The sending of the message is aborted. No messages are sent to their user, even after the start of **Shift 2**.

**EXAMPLE WITH FOLLOWING CALL:**

A message is sent to a group.

- ▶ The group contains the users **User 1, User 2, User 3, User 4, User 5** and **User 6**, in exactly this order.
- ▶ The **Send a Message** function is configured with the *following call* method.
- ▶ At the time of the initial sending, a shift is active with **User 4, User 5** and **User 6**.

- ▶ The function compares the users of the group to the users in the shift.
- ▶ **User 4** is notified first.  
If they do not react, **User 5** is notified and then **User 6**.
- ▶ However, at this time, the original shift has already ended and the next shift has started with **User 1**, **User 2** and **User 3**.
- ▶ Because all users are arranged in the group before **User 6**, no further message is now sent.

**Recommendation:** Select *infinite successive call* as a **method** . Then all users are contacted until one reacts. In our example, **User 1** would be notified after a shift change to the new shift.

### 7.5.4.2 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 66).

Messages are individually configured:

- ▶ E-mail via Outlook and SMS via GSM or SMS gateway (on page 71)
- ▶ E-mail message via SMTP (on page 74)
- ▶ Voice message by audio file (on page 78)
- ▶ Voice message via text to speech (on page 80)

You can find details on configuration of the parameters for messages in the **Parameters for messages** (on page 85) chapter.



#### Information

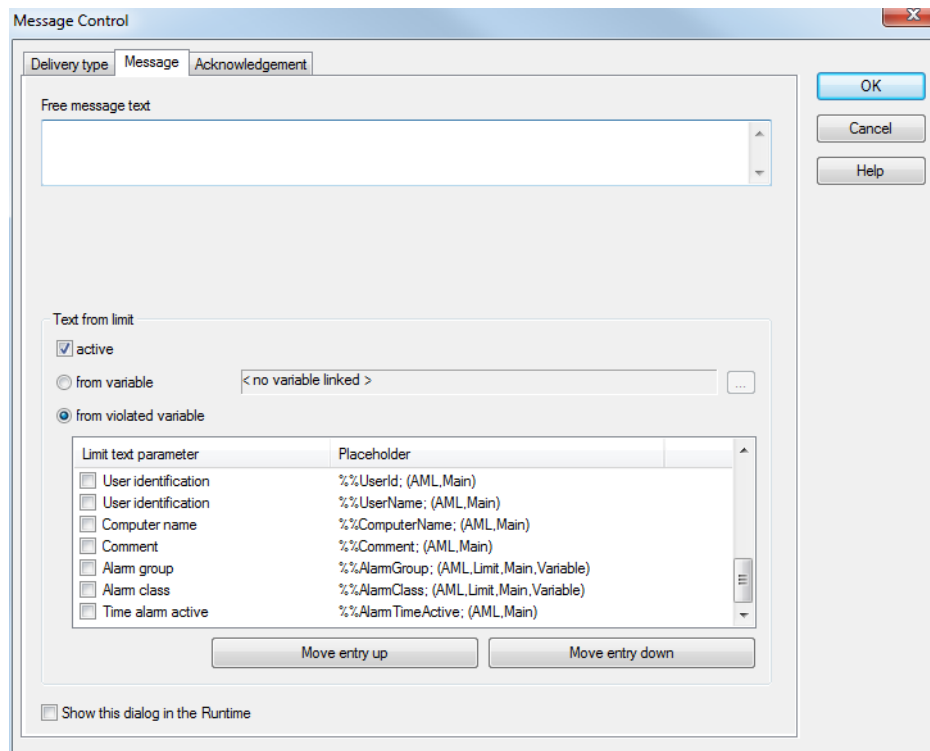
If parameters or values of additional variables are included in a message, these variables are not automatically logged on at the driver. If a variable is not logged on at the point of time the function is being carried out it must be logged on during the activity. Under certain circumstances (e.g. in 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.

#### 7.5.4.2.1 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



Parameter	Description
<b>Free message text</b>	<p>Integral part of the message.</p> <p>This text is searched for placeholders:</p> <ul style="list-style-type: none"> <li>► <b>\$:</b> indicates "formatted" text. The text is subsequently parsed for parameters.</li> <li>► Text components are considered parameters if they are separated from the rest of the text with a semicolon (;) and end with a semicolon (;).</li> <li>► A Parameter (on page 85) can be: <ul style="list-style-type: none"> <li>- key of the language table</li> <li>- name of a variable</li> <li>- compound entry in the language table</li> </ul> </li> </ul> <p>Parameters can be copied from the List (on page 87) of <b>limit value text parameters</b>. You can read more about the subject in the <b>Parameters for messages</b> (on page 85) chapter.</p>
<b>Text from limit value</b>	Properties for additional parameters from limit values.
<b>active</b>	<p><i>Active:</i> Adds additional parameters from the limit value to the constant text. Configuration via options:</p>



Parameter	Description
	<ul style="list-style-type: none"> <li>▶ <b>from variable</b></li> <li>▶ <b>from penultimate variable</b></li> <li>▶ <b>list limit value text parameters</b></li> </ul> <p><b>Attention:</b> If this option is active only limit value violations can trigger this function.</p>
<b>from variable</b>	Parameters are taken over from the defined variable. Click on button ... in order to open the dialog for selecting a variable.
<b>from penultimate variable</b>	Parameters are taken over from the variable whose limit value was violated.
<b>list limit value text parameters</b>	<p>Selecting parameters from list by activating the corresponding checkbox. Sequence is defined by drag &amp; drop with the mouse or the buttons <b>move entry up</b> and <b>move entry down</b>.</p> <p><b>Split placeholders:</b> Displays placeholders which can be used in free message text. If the option <b>from variable</b> or <b>from penultimate variable</b> or the variable selection is changed the example entry will be adjusted. Placeholders can be copied.</p> <p>Structure: <b>%Variable%Parameter; (origin)</b></p> <ul style="list-style-type: none"> <li>▶ <b>%</b>: percentage sign as prefix for variables and parameters</li> <li>▶ <b>Variable</b>: Variable. If no variable is indicated the following parameter refers to the main variable.</li> <li>▶ <b>Parameter</b>: Parameter</li> <li>▶ <b>;;</b>: semicolon ends string.</li> <li>▶ <b>Origin</b>: 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</li> </ul> <p>For details, see the <b>Parameters for messages</b> (on page 85) and <b>Parameter limit value text</b> (on page 87) chapters.</p> <p>If invalid parameters are used a corresponding error message is displayed in the message: <b>&lt;Invalid parameter: 'Parameter'&gt;.</b></p>

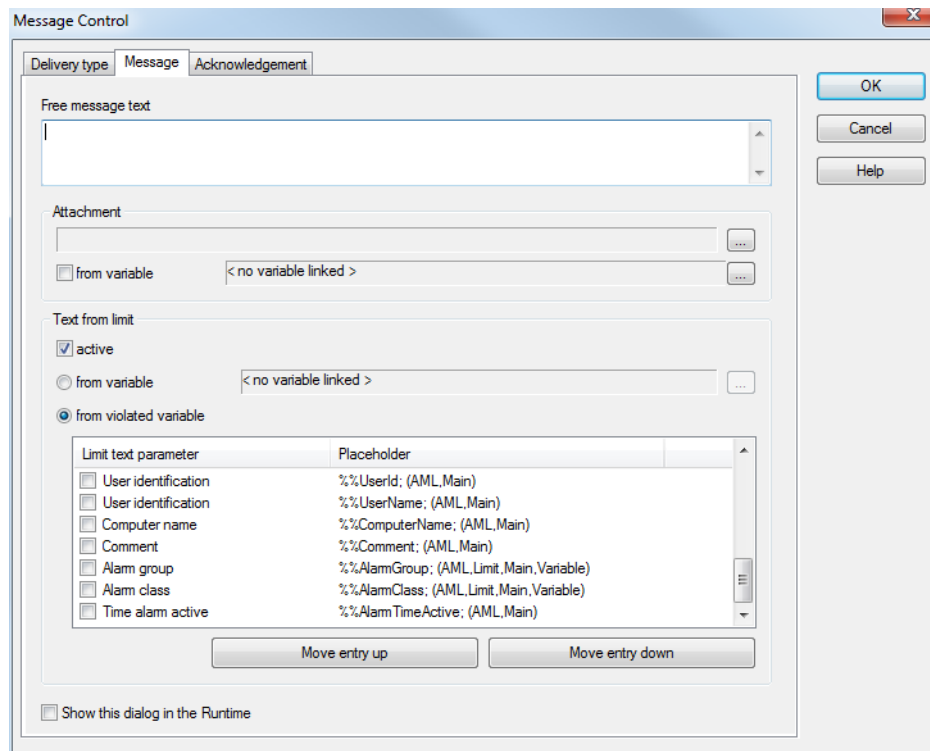
Parameter	Description
Move selected entry up	Places selected element one step higher.
Move selected entry down	Places selected element one step lower.
Show this dialog in the Runtime	<p><i>Active:</i> Opens this dialog in the Runtime before executing the function.</p> <p><b>Attention - Behavior in the Runtime:</b></p> <ul style="list-style-type: none"> <li>▶ In the 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.</li> <li>▶ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out.</li> <li>▶ If the dialog is edited and closed with OK the changes will be saved and the function carried out.</li> <li>▶ If the dialog is closed with <b>cancel</b> no message is being sent. This process is documented in the CEL provided for the property <b>Log in CEL</b> <i>all confirmations</i> or <i>only negative confirmations</i> was selected.</li> </ul>
OK	<p>Applies all changes in all tabs and closes the dialog.</p> <p>Dialog can only be closed if all required options have been configured.</p>
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

### 7.5.4.2.2 E-mail message via SMTP

#### E-MAIL MESSAGE VIA SMTP

Configuration of the message text for:

► E-mail message via SMTP



Parameter	Description
<b>Free message text</b>	<p>Integral part of the message.</p> <p>This text is searched for placeholders:</p> <ul style="list-style-type: none"> <li>► <b>\$:</b> indicates "formatted" text. The text is subsequently parsed for parameters.</li> <li>► Text components are considered parameters if they are separated from the rest of the text with a semicolon (;) and end with a semicolon (;).</li> <li>► A Parameter (on page 85) can be: <ul style="list-style-type: none"> <li>- key of the language table</li> <li>- name of a variable</li> <li>- compound entry in the language table</li> </ul> </li> </ul> <p>Parameters can be copied from the List (on page 87) of <b>limit value text parameters</b>. You can read more about the subject in the <b>Parameters for messages</b> (on page 85) chapter.</p>
<b>Appendix</b>	<p>Configuration of an appendix to the e-mail. Any desired file can be sent as an attachment.</p> <p>Clicking on the ... button opens the dialog to select a file with the presetting to the <b>Files/graphics</b> node.</p>

Parameter	Description
	<p>If the attachment is in another folder:</p> <ul style="list-style-type: none"> <li>▶ Activate the <b>Direct file selection</b> checkbox.</li> <li>▶ Click on the ... button for the file selection.</li> <li>▶ Select the desired folder and the desired file.</li> </ul> <p>When sending the message the selected file is added as attachment to the e-mail.</p> <p><b>Note:</b> To send files from other folders, select the <b>From variable</b> option.</p>
<b>from variable</b>	<p>Sending of attachments using a variable. Any desired file can thus be sent from any desired path:</p> <ul style="list-style-type: none"> <li>▶ <i>Active:</i> Select a string variable. Click on button ... in order to open the dialog for selecting a variable.</li> </ul> <p>The value of the string variables is read in the Runtime and the content is analyzed. The path to the appendix is set according to the analysis:</p> <ul style="list-style-type: none"> <li>▶ Absolute path: The path is used unchanged.</li> <li>▶ No absolute path: The value of the string variables is attached to the path to the zenon graphics folder.</li> </ul> <p>When setting the path, a check is not carried out to see which expansion the file has.</p>
<b>Text from limit value</b>	Properties for additional parameters from limit values.
<b>active</b>	<p><i>Active:</i> Adds additional parameters from the limit value to the constant text. Configuration via options:</p> <ul style="list-style-type: none"> <li>▶ <b>from variable</b></li> <li>▶ <b>from penultimate variable</b></li> <li>▶ <b>list limit value text parameters</b></li> </ul> <p><b>Attention:</b> If this option is active only limit value violations can trigger this function.</p>
<b>from variable</b>	Parameters are taken over from the defined variable. Click on button ... in order to open the dialog for selecting a variable.
<b>from penultimate variable</b>	Parameters are taken over from the variable whose limit value was

Parameter	Description
	violated.
<b>list limit value text parameters</b>	<p>Selecting parameters from list by activating the corresponding checkbox. Sequence is defined by drag &amp; drop with the mouse or the buttons <b>move entry up</b> and <b>move entry down</b>.</p> <p><b><u>Split placeholders:</u></b> Displays placeholders which can be used in free message text. If the option <b>from variable</b> or <b>from penultimate variable</b> or the variable selection is changed the example entry will be adjusted. Placeholders can be copied.</p> <p>Structure: <b>%Variable%Parameter; (origin)</b></p> <ul style="list-style-type: none"> <li>▶ <b>%</b>: percentage sign as prefix for variables and parameters</li> <li>▶ <b>Variable</b>: Variable. If no variable is indicated the following parameter refers to the main variable.</li> <li>▶ <b>Parameter</b>: Parameter</li> <li>▶ <b>;</b>: semicolon ends string.</li> <li>▶ <b>Origin</b>: 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</li> </ul> <p>For details, see the <b>Parameters for messages</b> (on page 85) and <b>Parameter limit value text</b> (on page 87) chapters.</p> <p>If invalid parameters are used a corresponding error message is displayed in the message: <b>&lt;Invalid parameter: 'Parameter'&gt;.</b></p>
<b>Move selected entry up</b>	Places selected element one step higher.
<b>Move selected entry down</b>	Places selected element one step lower.
<b>Show this dialog in the Runtime</b>	<p>Active: Opens this dialog in the Runtime before executing the function.</p> <p><b>Attention - Behavior in the Runtime:</b></p> <ul style="list-style-type: none"> <li>▶ In the 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.</li> </ul>

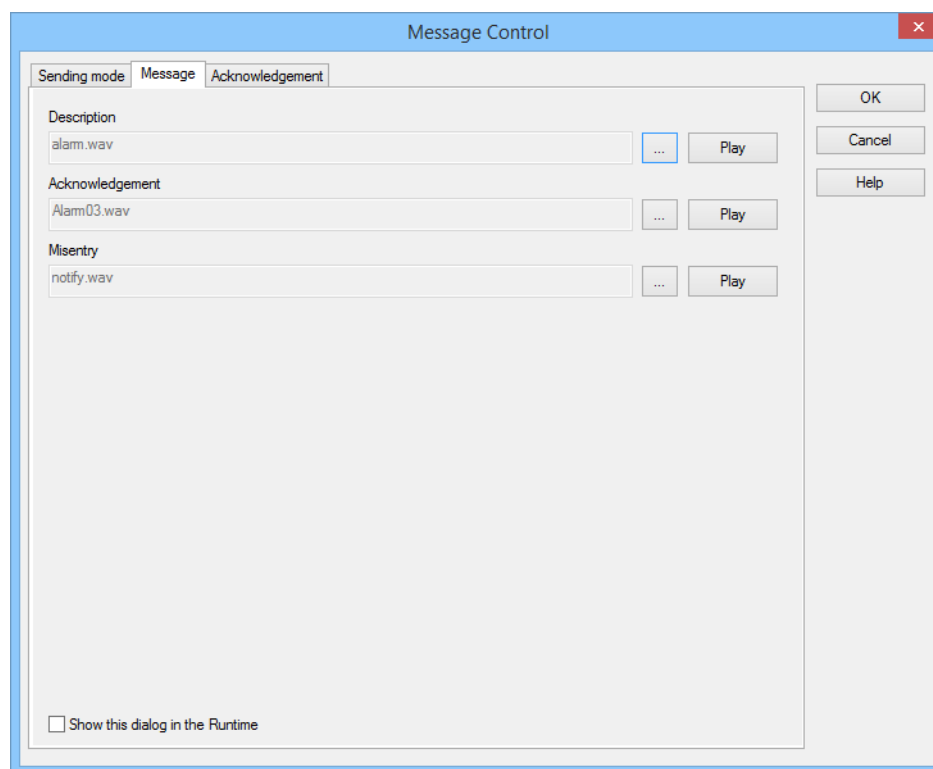
Parameter	Description
	<ul style="list-style-type: none"> <li>▶ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out.</li> <li>▶ If the dialog is edited and closed with OK the changes will be saved and the function carried out.</li> <li>▶ If the dialog is closed with <b>cancel</b> no message is being sent. This process is documented in the CEL provided for the property entries into CEL <i>all confirmations</i> or <i>only negative confirmations</i> was selected.</li> </ul>
<b>OK</b>	<p>Applies all changes in all tabs and closes the dialog.</p> <p>Dialog can only be closed if all required options have been configured.</p>
<b>Cancel</b>	Discards all changes in all tabs and closes the dialog.
<b>Help</b>	Opens online help.

### 7.5.4.2.3 Voice message by audio file

Configuration of the message text for voice messages via audio file using:

- ▶ Modem  
or

► VoIP



Parameter	Description
<b>Description</b>	<p>Select the file that contains the description of the message.</p> <p>Click on button ... in order to open the dialog for selecting a file. This has to be deposited in the node File/Multimedia.</p> <p>The file can be played for testing with the button <b>Play</b>.</p>
<b>Acknowledgement</b>	<p>Select the file that contains the text for the message if the <b>description</b> is confirmed positively.</p> <p>Click on button ... in order to open the dialog for selecting a file. This has to be deposited in the node File/Multimedia.</p> <p>The file can be played for testing with the button <b>Play</b>.</p>
<b>Misentry</b>	<p>Select the file containing the text in case of failure entries by the recipient.</p> <p>Click on button ... in order to open the dialog for selecting a file. This has to be deposited in the node File/Multimedia.</p> <p>The file can be played for testing with the button <b>Play</b>.</p>
<b>Show this dialog in the</b>	<i>Active:</i> Opens this dialog in the Runtime before executing the

Parameter	Description
Runtime	<p>function.</p> <p><b>Attention - Behavior in the Runtime:</b></p> <ul style="list-style-type: none"> <li>▶ In the 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.</li> <li>▶ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out.</li> <li>▶ If the dialog is edited and closed with OK the changes will be saved and the function carried out.</li> <li>▶ If the dialog is closed with <b>cancel</b> no message is being sent. This process is documented in the CEL provided for the property entries into CEL <i>all confirmations</i> or <i>only negative confirmations</i> was selected.</li> </ul>
OK	<p>Applies all changes in all tabs and closes the dialog.</p> <p><b>Note:</b> The entries are validated in the tabs by clicking on <b>OK</b>. If incorrect configurations are discovered, the user is informed by means of a message box. The user can then change the configuration or continue with the saving of the incorrect configuration.</p>
Cancel	Discards all changes in all tabs and closes the dialog.
Help	Opens online help.

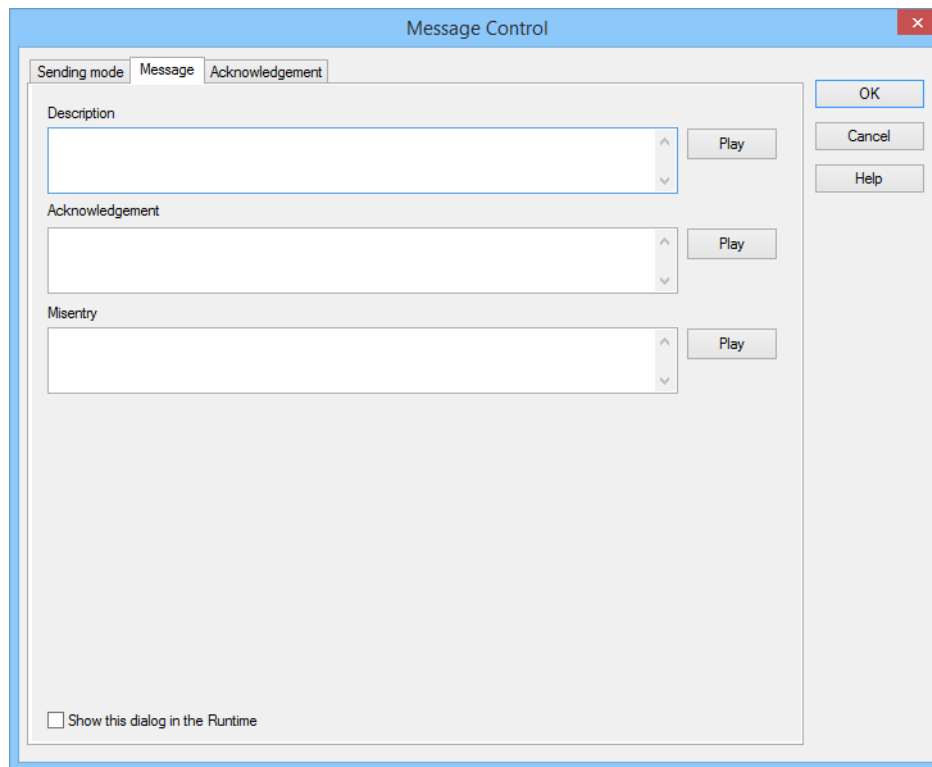
#### 7.5.4.2.4 Voice message via text to speech

Configuration of the message text for voice messages via text-to-speech using:

- ▶ Modem  
or



## ► VoIP



Parameter	Description
<b>Description</b>	<p>Enter the text to be displayed as message via the Text-to-Speech engine.</p> <p>The text can be entered as free message text. It can also contain dynamic components. These are configured in the same way as dynamic texts (on page 85).</p> <p>The text can be played for testing with the button <b>Play</b>.</p>
<b>Acknowledgment</b>	<p>Enter the text to be displayed as a message if, on positive confirmation of the <b>description</b>, playback is to be via the Text-to-Speech engine.</p> <p>The text can be entered as free message text. It can also contain dynamic components. These are configured in the same way as dynamic texts (on page 85).</p> <p>The text can be played for testing with the button <b>Play</b>.</p>
<b>Misentry</b>	<p>Enter the text that is to be played back via the Text-to-Speech engine in the event of incorrect entries.</p> <p>The text can be entered as free message text. It can also contain dynamic components. These are configured in the same way as</p>

Parameter	Description
	dynamic texts (on page 85). The text can be played for testing with the button <b>Play</b> .
<b>Show this dialog in the Runtime</b>	<ul style="list-style-type: none"> <li>▶ <i>Active</i>: Opens this dialog in the Runtime before executing the function.</li> </ul> <p><b>Attention: Behavior in the Runtime</b></p> <p>Note the following when calling up this dialog in the Runtime:</p> <ul style="list-style-type: none"> <li>▶ In the 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.</li> <li>▶ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out.</li> <li>▶ If the dialog is edited and closed with OK the changes will be saved and the function carried out.</li> <li>▶ If the dialog is closed with <b>cancel</b> no message is being sent. This process is documented in the CEL if <i>all confirmations</i> or <i>only negative confirmations</i> was selected for the <b>Log in CEL</b> property (<b>Project-specific settings</b>group).</li> </ul>
<b>OK</b>	<p>Applies all changes in all tabs and closes the dialog.</p> <p><b>Note:</b> The entries are validated in the tabs by clicking on <b>OK</b>. If incorrect configurations are discovered, the user is informed by means of a message box. The user can then change the configuration or continue with the saving of the incorrect configuration.</p>
<b>Cancel</b>	Discards all changes in all tabs and closes the dialog.
<b>Help</b>	Opens online help.

### 7.5.4.3 Acknowledgment of receipt

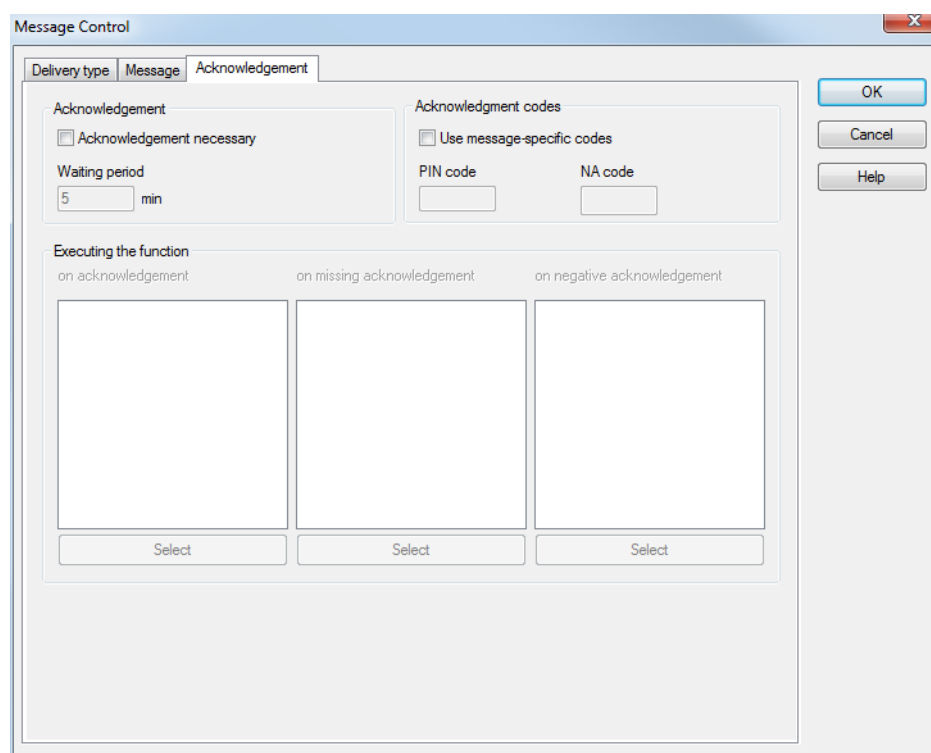
In this tab the required confirmations and the response to them are defined.



## Attention

Synchronously-transmitted messages (voice message via text-to-speech (on page 80) and voice message via audio file (on page 78)) can only be confirmed synchronously via the respective transmission medium. The use of other transmission methods for confirmation (such as email or SMS) is not possible. The message must be acknowledged immediately via the same connection which is still open.

## ACKNOWLEDGMENT SETTINGS



Parameter	Description
<b>Acknowledgment of receipt</b>	Settings for acknowledgment of receipt.
<b>Acknowledgment necessary</b>	<p><i>Active:</i> The message is only removed from the message queue if, within the <b>waiting period</b>, an acknowledgment of receipt has arrived or the <b>waiting period</b> has run out.</p> <p>For voice messages this option is always active and cannot be deactivated.</p>
<b>Waiting period</b>	Period of time in minutes during which the receipt has to be acknowledged before the message is forwarded to a substitute person.

Parameter	Description
	Default: <b>5 minutes</b>
<b>Confirmation codes</b>	Settings for confirmation codes.
<b>Use message-specific codes</b>	<p><i>Inactive:</i> The codes defined in the properties of the user administration for <b>Message Control user/PIN code</b> and <b>Message Control user/NA code</b> are used.</p> <p><i>Active:</i> The defined codes in the options <b>PIN code</b> and <b>NA code</b> are used for confirmation.</p>
<b>PIN code</b>	Individual code for confirming the message. The code consists of a four-digit number between 0000 and 9999.
<b>NA code</b>	Individual code for rejecting the message. The code consists of a four-digit number between 0000 and 9999.
<b>Function execution</b>	Selection of functions to be executed after a message has been confirmed, rejected or not acknowledged at all.
<b>In case of confirmation</b>	Functions which are executed if the message is confirmed. A click on the <b>Select</b> button opens the function selection dialog.
<b>In case of lack of confirmation</b>	Functions which are executed if there is no response to the message. A click on the <b>Select</b> button opens the function selection dialog.
<b>In case of negative confirmation</b>	Functions which are executed if the receipt of message is rejected. A click on the <b>Select</b> button opens the function selection dialog.
<b>OK</b>	Applies all changes in all tabs and closes the dialog.  Dialog can only be closed if all required options have been configured.
<b>Cancel</b>	Discards all changes in all tabs and closes the dialog.
<b>Help</b>	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.

### 7.5.5 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.

### 7.5.6 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

## 7.6 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 71)
- ▶ **E-mail message via SMTP** (on page 74)
- ▶ **Voice message via text-to-speech** (on page 80) via modem or VoIP

- ▶ **Dynamic part of the subject** property:

## CREATION OF A MESSAGE

Syntax for the structure of a message:

- ▶ A parameter (on page 85) can:
  - ▶ Be a key from the language table, such as **@MyText**;
  - ▶ Define a certain variable that is called up on execution of the function by means of AML or a limit value breach by means of its name: for example **%Var1**;  
**Caution:** The variable whose limit value has been breached is not dynamically determined for the subject!
  - ▶ Be a compiled entry in the language table: for example **%@Var2+MultipleText**
- ▶ **\$**: marks text that contains the parameter.
- ▶ **@**: marks language switching
- ▶ **%**: marks variables
- ▶ **%%**: marks **limit value text parameters** (on page 87) for variables  
 A variable can be stated between the two percentage marks.  
 If no variable is given, the value relates to the main variable.
- ▶ Text parts are separated from one another with a semi colon (;).
- ▶ 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

Parameter	In the Runtime	Action
key: <b>@MyText</b>	A check is made to see if an entry in @MyText is present in the language table for the current language.	<ul style="list-style-type: none"> <li>▶ <i>Existing</i>: entry is added to the message.</li> <li>▶ <i>non-existent</i>: According to the settings of the project properties <b>Show untranslated keywords</b> either <b>@MyText</b> or <b>MyText</b> is added to the message.</li> </ul>

Parameter	In the Runtime	Action
Variable: <b>%Var1%</b>	A check is made if the variable exists and the value can be read.	<ul style="list-style-type: none"> <li>▶ <i>Existing</i>: Value of the variable is taken and added to the text as string.</li> <li>▶ <i>Not existing/not readable</i>: The text <b>xxx</b> is added to the message.</li> </ul>
Compound entry: <b>@StringTable+%var1 Text</b>	A check is made if: <ul style="list-style-type: none"> <li>▶ 1. the variable exists and the value can be read. Value is attached to the prefix text ("MultipleText") as string.</li> <li>▶ 2. an entry in the language table exists for the text</li> </ul>	<ul style="list-style-type: none"> <li>▶ Variable and text exist: text is added to the message. Example: Value of the variable is 33. The language table is then checked for <b>@StringTable33</b> and the corresponding text is added.</li> <li>▶ <i>non-existent</i>: According to the settings of the project properties <b>Show untranslated keywords</b> either <b>@MyText</b> or <b>MyText</b> is added to the message.</li> </ul>

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

### 7.6.1 Text from limit value and free text

Messages can be configured with free text. This can also contain texts from limit values and dynamic text elements. Not all dispatch types in **Message Control** support texts from limit values. However all texts with dynamic elements can be added to.

## CONFIGURATION OF TEXT FROM LIMIT VALUE

To link limit values:

1. In the **Message** tab, activate the **Text from limit value active** option (Message Control function **Send a Message**).  
**Note:** This option is not available for all dispatch types. In this case, configure the option **Free message text** with dynamic text elements. (See **Configuration of free message text** section.)
2. Activate the option **from violated variable**.
3. Activate the checkboxes for the desired values.  
For parameters, see **List of limit value text parameters**.
4. Order the selected parameters by means of Drag&Drop or with the corresponding buttons.
5. Ensure that the limit value violation triggers the function.

## CONFIGURATION OF FREE MESSAGE TEXT

### STRUCTURE

**\$FREETEXT;%VARIABLE%TEXT;**

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

- ▶ Start sign: \$  
Parameters of "*free text*" are thus taken into account.
- ▶ Free text:  
is added between semicolons (;)
- ▶ Prefix for **Limit value text parameters**:  
2 percentage signs (%% or % variable name %).  
  
If no variable is indicated 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 violated variable plus text
- ▶ %%LimitText: Limit value text of the violated variable plus text
- ▶ %Master%VariableStatus: Status of the **Master** variables plus text

#### Hint

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

### LIST LIMIT VALUE TEXT PARAMETERS

Parameter	Main variable - Activating via limit value	Main variable - Activating via AML	Additional variable AML/limit value
<b>%%Name</b> (Variable name:)	Name of the variable, corresponds to property <b>Name</b> .	Name of the variable, corresponds to property <b>Name</b> .	Name of the variable, corresponds to property <b>Name</b> .
<b>%%Identification</b>	Description of the variable, corresponds	Description of the variable, corresponds	Description of the variable, corresponds



Parameter	Main variable - Activating via limit value	Main variable - Activating via AML	Additional variable AML/limit value
(Identification)	to property <b>Identification.</b>	to property <b>Identification.</b>	to property <b>Identification.</b>
<b>%%LimitText</b> (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 <b>Limit value text.</b>
<b>%%MessageCreateTimeStamp</b> (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).
<b>%%ResourceLabel</b> (Resource Label Identification)	Text for the resource label identification of the variable, corresponds to property <b>Resources label.</b>	Text for the resource label identification of the variable, corresponds to property <b>Resources label.</b>	Text for the resource label identification of the variable, corresponds to property <b>Resources label.</b>
<b>%%LimitTimeReceived</b> (Time alarm is generated)	Time stamp of the alarm object.	Time stamp of the alarm object.	-
<b>%%</b> (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).
<b>%%VariableStatus</b> (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).
<b>%%VariableTimeStamp</b> (Time stamp of the variable)	Time stamp of the variable (when activating the function).	Time stamp of the variable (when activating the function).	Time stamp of the variable (when activating the function).
<b>%%MessageTimeAckno</b>	Available time for	Available time for	Available time for

Parameter	Main variable - Activating via limit value	Main variable - Activating via AML	Additional variable AML/limit value
<b>wledge</b> (Remaining time for confirmation)	acknowledgement in minutes. Only available if acknowledgement of receipt required (on page 82) was activated.	acknowledgement in minutes. Only available if acknowledgement of receipt required (on page 82) was activated.	acknowledgement in minutes. Only available if acknowledgement of receipt required (on page 82) was activated.
<b>%%Address</b> (Address)	Address.	Address.	Address.
<b>%%Unit</b> Measuring unit:	Technical unit according to <b>Measuring unit</b> property.	Technical unit according to <b>Measuring unit</b> property.	Technical unit according to <b>Measuring unit</b> property.
<b>%%AlarmArea</b> (Alarm area)	Alarm area of the main variable.	Alarm area of the main variable.	Alarm area of the additional variable.
<b>%%LimitTimeAcknowledge</b> (Time alarm is acknowledged)	--	Time stamp of the alarm object.	--
<b>%%UserId</b> (User identification)	--	User identification of the user who started the action. Conforms to property <b>User name</b> .	--
<b>%%UserName</b> (User name)	--	Complete name of the user who started the action. Conforms to property <b>Complete name</b> .	--
<b>%%ComputerName</b> (Computer name)	--	Name of the computer on which the action was started.	--
<b>%%Comment</b>	--	Comment of the alarm object.	--

Parameter	Main variable - Activating via limit value	Main variable - Activating via AML	Additional variable AML/limit value
(Comment)			
<b>%%AlarmGroup</b> (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).
<b>%%AlarmClass</b> (Alarm/event class)	Alarm/event class of the alarm object (as text).	Alarm/event class of the alarm object (as text).	Alarm/event class of the alarm object (as text).
<b>%%AlarmTimeActive</b> (Time alarm is pending)	--	Period of time the alarm has been pending.	--

## ERROR TREATMENT

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.

## 8 Message Control in the Runtime

In the Runtime Message Control is either started automatically (property **Active during Runtime start** active) or with the function Send Message: activate (on page 85).

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 96) is being analyzed
- ▶ the functions used for acknowledgement are executed



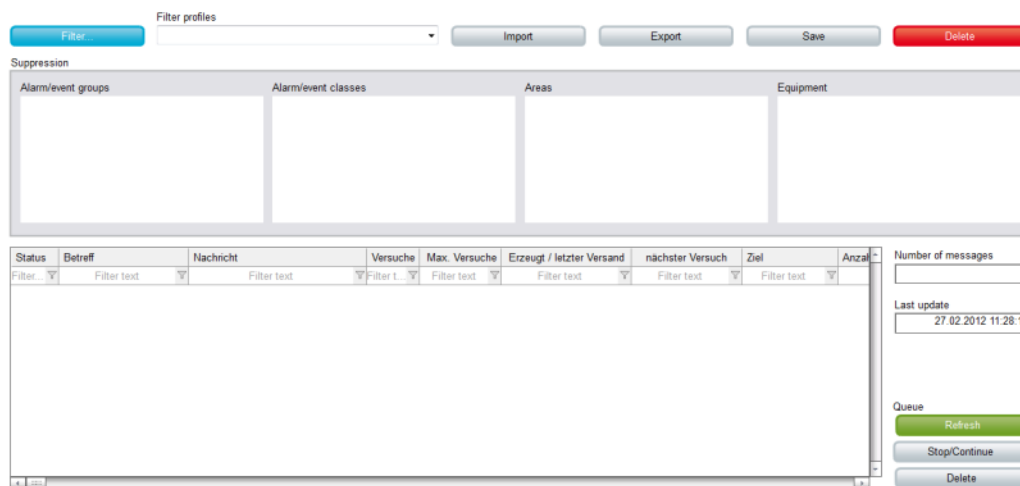
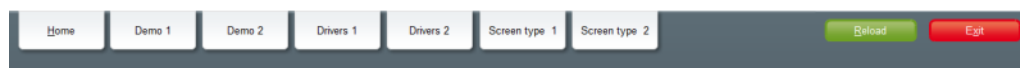
## Attention

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 also received (messages with a different subject), these may possibly be set to **"read"**.

## PROCEDURE IN THE RUNTIME

The message queue can be managed with a Message Control screen (on page 93).

Message Control is being closed when closing the Runtime or by the function Send Message: deactivate (on page 85).

## TIME STAMP IN ALARM MESSAGE LIST AND MESSAGE CONTROL

When the function Send message (on page 64) 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 - REMANANCE

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 behavior in the network in the Network (on page 101)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 acknowledgment of messages, see the Message Acknowledgment (on page 96) chapter.

## LANGUAGE SWITCHING

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.

## 8.1 Connect screen of the type Message Control

The *message control* screen makes a copy of the current message queue and control elements for analysis and filtering available in the Runtime. The content of the elements displayed in the **message list** list field corresponds to the point of time when the screen is connected 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.

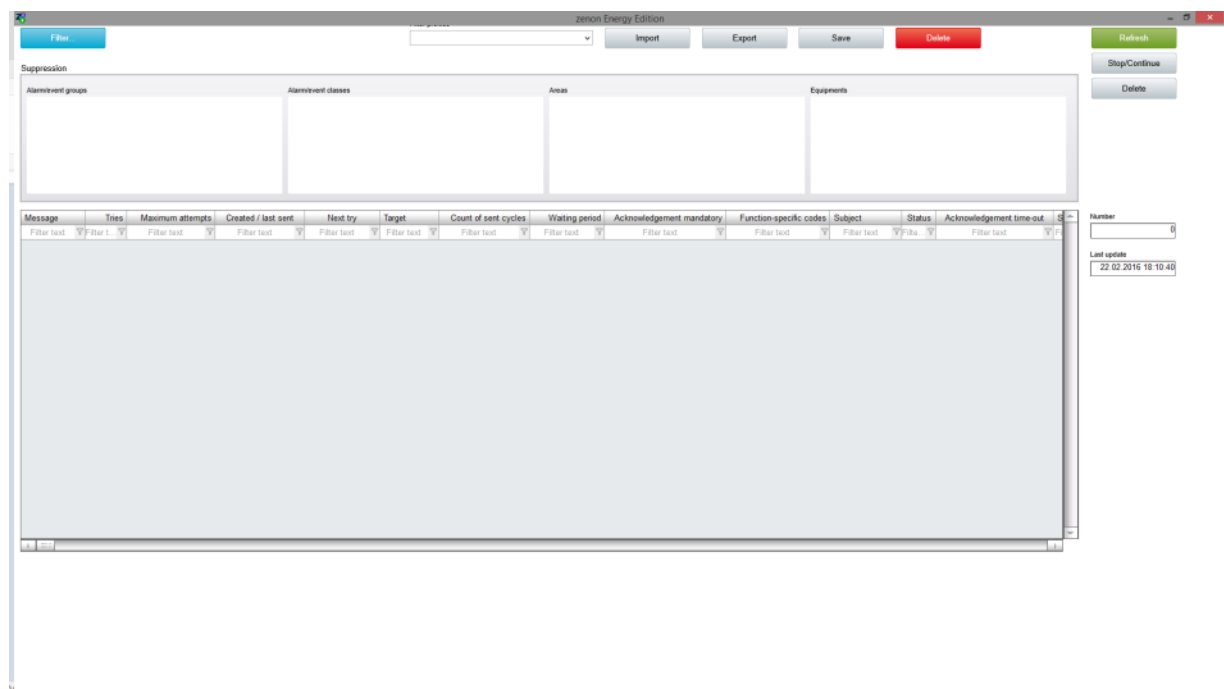
The **message list** contains messages with this status:

- ▶ Waiting for dispatch
- ▶ Is currently being sent
- ▶ Has already been sent, but not yet confirmed
- ▶ There was an error while sending

All sent messages can be checked in the CEL.

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 54)) and can be provided with individual column headings. These labelings are localizable.



## INSERT TEMPLATE

Control elements	Description
<b>Insert template</b>	<p>Opens the dialog for selecting a template for the screen type.</p> <p>Templates are shipped together with zenon and can also be created by the user.</p> <p>Templates add pre-defined control elements to pre-defined position 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</p>

Control elements	Description
	and placed in the zenon screen. Elements can be moved on the screen and arranged individually.

## WINDOW

Lists and fields which can be displayed in the Runtime.

Control elements	Description
<b>Message queue</b>	<p>List of messages.</p> <p>Contains messages which have been sent, confirmed or deleted or remain to be sent.</p> <p>The messages are only displayed and cannot be edited anymore.</p> <p>Column headings can be named individually (on page 56) and are localizable by putting @ before them.</p>
<b>Number of messages</b>	Number of currently existing messages.
<b>Last Update</b>	Point of time of last update.
<b>Suppressed alarm/event groups</b>	<b>Alarm/event groups</b> for which messages were suppressed and the alarms of which are thus not displayed in the message queue.
<b>Suppressed alarm/event classes</b>	<b>Alarm/event classes</b> for which messages were suppressed and the alarms of which are thus not displayed in the message queue.
<b>Suppressed alarm/event areas</b>	<b>Alarm/event areas</b> for which messages were suppressed and the alarms of which are thus not displayed in the message queue.
<b>Suppressed pieces of equipment</b>	<b>Equipment</b> for which messages were suppressed and the alarms of which are thus not displayed in the message queue.

## LIST FUNCTIONS

Control elements to control lists.

Control elements	Description
<b>Filter</b>	Opens Dialog (on page 54) to configure the filters for the

Control elements	Description
	view.
<b>Delete</b>	Deletes selected entry from the list. It is only possible to delete messages which have not been sent yet.
<b>Refresh</b>	Refreshes view. The current data is loaded onto the screen from the message queue.  The point of time of the last update can be displayed with the field <b>Last update</b> .
<b>Stop/Continue</b>	Starts and stops the cyclic update of the list.

## FILTER PROFILES

Profile administration

Control elements	Description
<b>Profile selection</b>	Opens the dialog for selecting a profile.
<b>Save</b>	Saves current setting as a profile.
<b>Delete</b>	Deletes profile.
<b>Import</b>	Opens dialog for importing profiles from a file.
<b>Export</b>	Opens dialog for exporting profiles from a file.

## 8.2 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	To acknowledge -: no +: yes
E-mail message via Outlook (on page 16)	-
E-mail message via SMTP (on page 17)	-



Medium	To acknowledge -: no +: yes
SMS message via GSM modem (on page 19)	-
SMS message via SMS gateway (on page 22)	-
Voice message by audio file (on page 25)	+
Voice message via text to speech (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 **Maximum 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 64) open the tab Acknowledgement of receipt (on page 82).
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**

The code consists of a four-digit number between 0000 and 9999.

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 71) 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 THE 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. The following must be the case 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.
- ▶ The original subject must not be changed in the email. This means: The contents defined in Outlook in **Subject (ID)** or for SMTP in **Mail subject** must remain unchanged. These will identify and allocate the message during receipt in zenon.
- ▶ However the subject must have at least three characters placed in front of it as a response. For example **RE:** or **AW:**.

- ▶ The response text must contain:
  - ▶ GUID: At the start of the message. There must be no other character, including spaces, in front.
  - ▶ Semi-colon (;): Separator.
  - ▶ Text: Contains 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 do this, the message is checked on receipt to see if it starts with at least three characters such as **RE:** or **AW:**.
- ▶ Message Control analyzes GUID and code and subsequently executes the functions set in the function Send Message (on page 64).

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

### ACKNOWLEDGING SMS

A 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: At the start of the message. There must be no other character, including spaces, in front.
    - ▶ Semi-colon (;): Separator.
    - ▶ Text: Contains 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 64).



#### Information

SMSs to Smartphones can also be received and acknowledged with the **Notifier App by zenon**. To do this, the app sends the response as an SMS in the format **Alarm-ID;PIN** or **Alarm-ID;NA**.

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 must support DTMF.
- ▶ The message is read to the recipient with the text of the **description**
- ▶ 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 (description, confirmation or error message) is 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 playback of the message starts with the **description** again.
- ▶ Message Control analyzes the code and subsequently executes the functions set in the function Send Message (on page 64).

**Note:** Voice messages always must be acknowledged by telephone.

## 8.3 Voice messages in Runtime

The procedure for the sending of voice messages is as follows:

1. A function calls up the desired dispatch method.
2. The call is initiated by means of the VoIP-Server or the modem.

**Note:** In the list of messages, the name of the audio file is displayed in the **Message** column; with sending by audio file, the message text is displayed with Text-to-Speech.

3. Establish connection: The connection is considered established as soon as the call to the counterparty is signaled.  
When a call is taken, it is not possible to distinguish whether the call was taken by a person or an answering machine.

Another attempt is made if no connection can be established.

The number of attempts corresponds to the value in the **Project-specific settings/Maximum number of attempts** property. The time between the attempts corresponds to the value in the **Project-specific settings/Waiting period between trials and timeout** property.

**Note:** If a message is forwarded to another user, this is considered a new message. The counter is automatically set back to 1.

**Attention:** The value for the project-specific **Waiting period between trials and timeout** property must be greater than the value for the **Waiting time** option in the **Send a Message** functions (**Receipt Confirmation** tab).

4. If the connection cannot be established or the message is rejected with an incorrect PIN, the substitute person is notified.  
If no substitute person is configured, the call is canceled.
5. A welcome message is played back as soon as the call is taken.  
This message can be requested again by the recipient with the **#** key code.
6. The recipient reacts to the message with a confirmation or rejection. To do this, it enters the corresponding code:
  - ▶ Confirmation: **PIN code**
  - ▶ Rejection: **NA code**
7. If receipt confirmation has been saved for the message functions, these are executed accordingly.

## CODE ENTRY

The code is entered by DTFM. The following characters are interpreted:

- ▶ **#**: Repeat current message.
- ▶ **\***: Delete entered character and start process again with welcome message.
- ▶ Numbers (0 - 9): Entry of the 4-digit code

Only the first four sounds are interpreted when entering the code. All other DTMF key tones are ignored for PIN and NA, but lead to the message being repeated.

The entry can be corrected at any time with the **\*** key. In doing so, all characters entered are deleted and the process is started again with the welcome message being played back.

## 8.4 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 59) is being synchronized and remanently saved.

The function Send message (on page 64) is only executed at the process-executing computer. Sending by the client is not supported.

### USER

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 on the zenon web client is also possible.

### 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 files and folders.
- ▶ 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

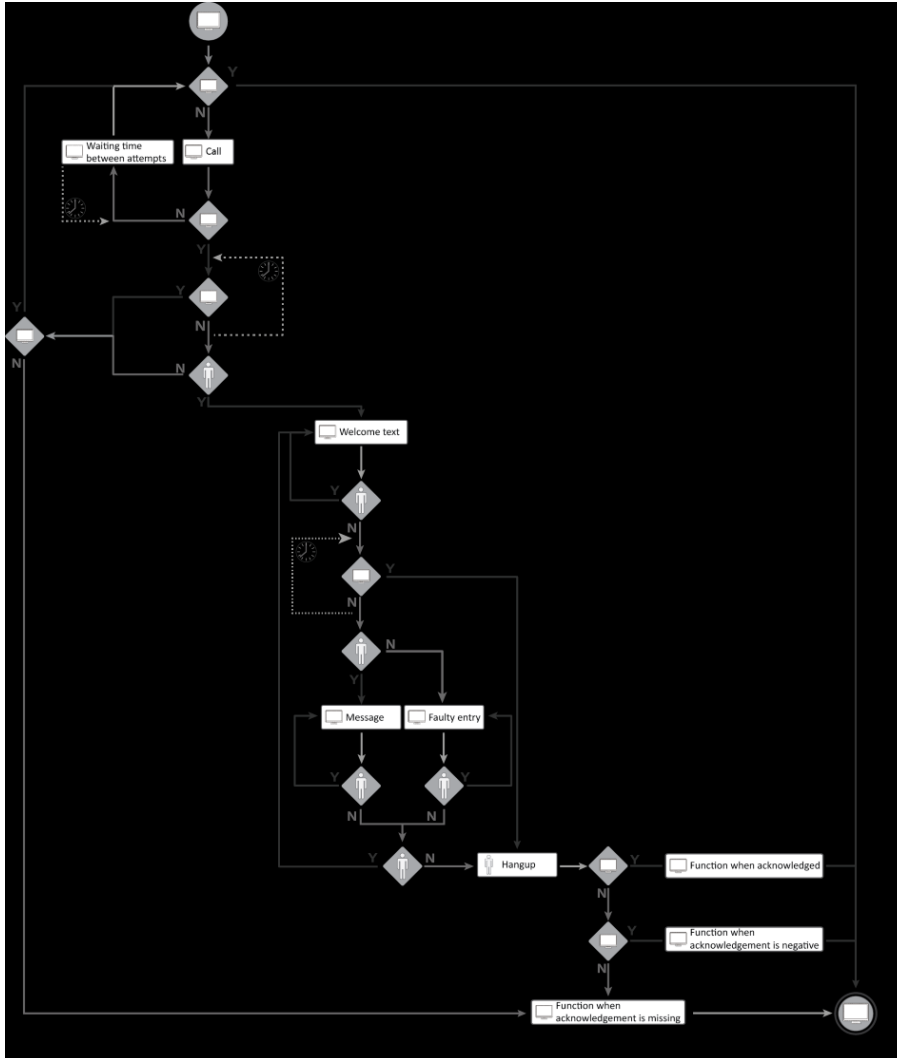
### BEHAVIOR IN CASE OF FAILURE

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.

## 8.5 Voice message process

The process when sending a voice message and when receiving the confirmation corresponds to the following diagram:



### Key:

- ▶ Y: Yes
- ▶ N: No

### PROCEDURE

The process of the function can be subdivided into main processes and subprocesses.

#### Main processes:

- ▶ Start **Send a Message** function
- ▶ Text description

- ▶ End call

### Subprocesses:

- ▶ Substitute person
- ▶ Misentry
- ▶ Acknowledgement

### MAIN PROCESSES:

#### Start "Send message" function:

1. **Send a Message** function executed
2. Validation: Has the defined value for the **Maximum number of attempts** function been reached?

**Note:** If a message is forwarded to another user, this is considered a new message. The counter is automatically set back to 1.

- ▶ **Y:** **Send a Message** is ended.
  - ▶ **N:** Call is carried out.
3. Validation: Is the call announced at the partner?
    - ▶ **N:** The wait time between two attempts defined in the **Waiting period between trials and timeout** function is waited for according to the configuration. The **Send a Message** function is then executed again
    - ▶ **Y:** There is a wait for a response.

**Attention:** The value for the project-specific **Waiting period between trials and timeout** property must be greater than the value for the **Waiting time** option in the **Send a Message** functions (**Receipt Confirmation** tab).

4. Validation: Has the waiting time defined in the **Timeout [min]** function been exceeded?
  - ▶ **Y:** Further with **Substitute person** process.
  - ▶ **N:** There is a wait for the call to be taken.
5. Validation: Is the call taken?
  - ▶ **N:** Further with **Substitute person** process.
  - ▶ **Y:** Further with **text description** process.

#### Text description:

1. **Description** is sent.
2. Validation: Is # or \* being sent by the partner?
  - ▶ **Y:** **Description** is sent.



- ▶ **N:** Checking for the waiting time.
- 3. Has the waiting time that was defined in the **waiting time** option of the **Send a Message** function been exceeded?
  - ▶ **Y:** Further with **End call** process.
  - ▶ **N:** There is a wait for the entry of the **PIN** or **NA** code.
- 4. Validation: Has there been a correct **PIN** or **NA** code?
  - ▶ **Y** Continue with **Confirmation** process.
  - ▶ **N:** Continue with **incorrect entry** process.
- 5. Validation: Received code \*?
  - ▶ **Y:** Continue with **text description** process.
  - ▶ **N:** Continue with **End call** process.

#### End call

1. Validation: Has **PIN** been received?
  - ▶ **Y:** Execute linked function for **PIN**, if present.
  - ▶ **N:** Continue to **NA** process.
2. Validation: Has **NA** been received?
  - ▶ **Y:** Execute linked function for **NA**, if present.
  - ▶ **N:** Continue to the next point **No confirmation**.
3. No confirmation received: Execute linked function if present.
4. End **Send a Message** function.

### SUBPROCESSES

#### Substitute person:

- ▶ Validation: Has a substitute person been defined?
  - ▶ **Y:** Function for the substitute person is executed once again.  
Continue with check: Has the defined value for the **Maximum number of attempts** function been reached?
  - ▶ **N:** Continue to **End call** process Item 3: No confirmation received.

#### Misentry:

- ▶ Validation: Entry # or number between 0 and 9?
  - ▶ **N:** Validation: Received code \*?
  - ▶ **Y:** Back to checking process: Entry # or number between 0 and 9?

**Confirmation:**

- ▶ Validation: Entry # or number between 0 and 9?
- ▶ **N**: Validation: Received code \*?
- ▶ **Y**: Back to checking process: Entry # or number between 0 and 9?

## 9 Messages and error handling

Messages, notes and errors can be logged and displayed with the Diagnosis Viewer in a separate module [**Message Control**]. Message details can be found in section LOG entries (on page 111).

### 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, it is preferable that you use the driver from the modem 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).

## 9.1 Check list

In case of problems check:

Problem	Possible cause	Solution
<b>SMTP mail</b>		
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.	<ul style="list-style-type: none"> <li>▶ Is the <b>Sending mode active</b> property activated?</li> <li>▶ Is the user who is supposed to receive the mail set as „Message Control User“ (property <b>Message Control user</b> activated)?</li> <li>▶ Was an e-mail address configured for the user?</li> <li>▶ Are the user data transferred to the Runtime after a change (property <b>Runtime changeable data</b>)?</li> <li>▶ Is the configuration of the sending mode correct? Compare to the data of the e-mail provider.</li> </ul>
	The server is registered correctly, but is not available (error messages when connecting).	<ul style="list-style-type: none"> <li>▶ Does the name resolution work?</li> <li>▶ Is the server available (Ping)?</li> <li>▶ 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.</li> </ul>
	The server does not support a command (error message with "Command not supported" or similar in the LOG file).	<ul style="list-style-type: none"> <li>▶ Check if the server supports the command (e.g. with Thunderbird or Telnet).</li> </ul> <p>If possible select a different configuration (e.g. SMTP AUTH commands can be bypassed with <i>"register at the incoming mail server before sending e-mails"</i>. APOP can be deactivated and replaced by user and password.</p>
	The server requires an encrypted connection. Since the mail is sent in	<ul style="list-style-type: none"> <li>▶ In this case the analysis with Wireshark shows that the server sends "STARTTLS". Since this command is</li> </ul>

Problem	Possible cause	Solution
	unencrypted form the server disconnects.	not supported the connection should be secured via SSL or TLS.
	The server does not support the selected protection mechanism (error messages when establishing an encrypted connection).	<ul style="list-style-type: none"> <li>▶ Select other protection configuration or plain text transmission.</li> </ul>
<b>SMS-GSM</b>		
zenon cannot establish a connection with the modem.	<ul style="list-style-type: none"> <li>▶ Wrong COM port (file-not-found error when opening the COM port).</li> <li>▶ COM port busy (e.g. by the configuration software of the manufacturer, access-denied error when opening the COM port).</li> <li>▶ Error when configuring the COM port/modem not responding.</li> <li>▶ The modem does not support the SMS-PDU mode (AT+CMGG error message or PDU mode error message in LOG).</li> <li>▶ Probably the modem requires a PIN / is the PIN incorrect? (PIN error messages in LOG)</li> <li>▶ Too many attempts to log on with an</li> </ul>	<ul style="list-style-type: none"> <li>▶ Check COM port and modem configuration.</li> <li>▶ Check PIN, if necessary release manually with the PUK.</li> </ul>

Problem	Possible cause	Solution
	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.	
Connection is established, but sending and receiving do not work.	<ul style="list-style-type: none"> <li>▶ 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!</li> <li>▶ Despite this the modem is not connected with the network (check with manufacturer software): Look for a location with better reception.</li> <li>▶ The SMSC telephone number that has been entered is not correct. Check the number before entering it into zenon or when establishing a</li> </ul>	<ul style="list-style-type: none"> <li>▶ Connect modem for the first time with manufacturer software with the GSM network.</li> <li>▶ Put modem in different location.</li> <li>▶ Check telephone number. Has the country code for the SMSC (00xy or +xy) been set correctly?</li> </ul>

Problem	Possible cause	Solution
	<p>connection for the first time. This check can also be carried out with the <b>Check connection</b> property activated. Load SMSC with HyperTerminal from modem and write.</p> <ul style="list-style-type: none"> <li>▶ Look at error message in LOG</li> </ul>	

## 9.2 LOG entries

Entries in the LOG file of the Diagnosis Viewer.

- ▶ Miscellaneous messages (on page 111)
- ▶ E-mail via SMTP/POP (on page 118)
- ▶ Voice over IP (on page 123)
- ▶ SMS (on page 123)
- ▶ OpenSSL Library (on page 127)

### 9.2.1 Miscellaneous messages

The listing of these LOG entries is ordered alphabetically according to levels and entries.

- ▶ Debug (on page 112)
- ▶ Deep Debug (on page 114)
- ▶ MSG (on page 117)
- ▶ Warnings (on page 118)

### 9.2.1.1 Debug

#### DEBUG

Level	Entry	Description
DEBUG	<i>Attaching to current call.</i>	The sending mode attaches to the active call.
DEBUG	<i>Connected</i>	A connection has been established.
DEBUG	<i>Couldn't send MessageDeleteRequest notification to Mainstation.</i>	The attempt to send a delete request telegram to the process-executing computer has failed.
DEBUG	<i>Current Message %Nachricht Inhalt%</i>	The current message (content, settings, etc.) is traced in the LOG.
DEBUG	<i>File alignment request could not be send to mainstation.</i>	The attempt to send an alignment request telegram to the process-executing computer has failed.
DEBUG	<i>Invalid code '%s' entered.</i>	An invalid code was entered.
DEBUG	<i>Mail to Benutzer '%s' was successfully added to the outbox.</i>	The message to user [name] was successfully transferred to the Outlook outbox.
DEBUG	<i>MessageDeleteRequest Response: %d messages were deleted from the queue.</i>	A delete response was received: [Number] messages were removed from the queue.
DEBUG	<i>Messaging canceled: The end of the Benutzer list for the Group (%s) has been reached.</i>	The end of user group [name] was reached, the sending is canceled.
DEBUG	<i>Requesting File alignment from mainstation.</i>	An alignment request is sent to the process-executing computer.
DEBUG	<i>Send SMS to Benutzer %s.</i>	A SMS was sent to the user [address].
DEBUG	<i>Sending mail to Benutzer '%s' using address: %s.</i>	A message for the user [name] is prepared and [sent] to the address.
DEBUG	<i>Sending MessageDeleteRequest for %d messages to Mainstation.</i>	A delete request for [number] messages is sent to the process-executing computer.
DEBUG	<i>SMS to Benutzer %s transmitted to the outbox.</i>	A SMS was transferred to the user [address] in the outbox.
DEBUG	<i>The #-sign was entered:</i>	# was entered: the last message is repeated.



Level	Entry	Description
	<i>Repeating the last message.</i>	
DEBUG	<i>The *-sign was entered: Clearing the current key.</i>	A * was entered, the current entered code is being reset.
DEBUG	<i>The current call be terminated now.</i>	A call is active, it is closed now.
DEBUG	<i>The is no call in progress, nothing to terminate.</i>	No call is active, nothing can be closed.
DEBUG	<i>The line %s has been closed.</i>	The line [] has successfully been closed.
DEBUG	<i>The line %s has been opened.</i>	The line [] has successfully been opened.
DEBUG	<i>The message %s should have been acknowledged by now, handling TimeOut.</i>	The message [ID] should already have been confirmed, TimeOut is being processed.
DEBUG	<i>The message %s should have been sent by now, handling hardware TimeOut.</i>	The message [ID] should already have been sent, TimeOut is being processed.
DEBUG	<i>The Message %s was NOT successfully sent to %s!</i>	The message [ID] could not be sent to [recipient].
DEBUG	<i>The Message %s was not successfully sent to %s, and will be send again!</i>	The message [ID] could not be sent to [recipient] and will be sent again.
DEBUG	<i>The Message %s was successfully sent to %s!</i>	The message [ID] was successfully sent to [recipient].
DEBUG	<i>The next user (%s) for the group (%s) will be notified, previous Benutzer was:%s</i>	The next user [name] from user group [name] is notified, the previous user was user [name].
DEBUG	<i>The sending of message %s was completed.</i>	The sending process for message [ID] has been completed.
DEBUG	<i>Valid Acknowledge-code entered.</i>	A valid PIN code for confirming the message was entered.
DEBUG	<i>Valid NegAcknowledge-code entered.</i>	A valid NA code for rejecting the message was entered.

### 9.2.1.2 Deepdebug

#### DEEP DEBUG

Level	Entry	Description
DEEPDEBU G	<i>A changed notification of message(%s) was received, the old message object will be updated!</i>	The project received a change telegram for the message with the GUID [number]. The local (previous) object is updated.
DEEPDEBU G	<i>A delete message(s) request was received!</i>	A delete request telegram was received.
DEEPDEBU G	<i>A getqueue request was received, sending queue.</i>	An alignment request was received. The current queue and the current status are being sent.
DEEPDEBU G	<i>A file alignment request was received, sending alignment data.</i>	A non-primary server has requested a sync of Runtime data. This data is sent. The message also contains the project names. The entry is created on the primary server.
DEEPDEBU G	<i>A remove notification of message(%s) was received, the message object will be removed from the list. (%s)</i>	The project received a remove telegram for message with GUID [number]. The local (previous) object is deleted.
DEEPDEBU G	<i>All messages for project %s will be removed.</i>	All messages of project [name] are removed from Message Control.
DEEPDEBU G	<i>An added notification of message(%s) was received, the message object will be added to the list. (%s)</i>	The project received an add telegram for message with GUID [number]. The new object is added.
DEEPDEBU G	<i>An ChangeSuppressionRequest was received. The local suppression-state will be updated!</i>	A ChangeSuppression-Request was received. The local suppression status is updated.
DEEPDEBU G	<i>An update of the message control state was received. The local state will be updated!</i>	The project received a MessageControl-StateChange telegram for message with GUID [number]. The new object is added.
DEEPDEBU G	<i>Current message count is (%d)</i>	There are [number] messages in the queue. (Entry is created when creating and deleting messages.)

Level	Entry	Description
DEEPDEBU G	<i>Firing MessageAcknowledged</i>	MessageAcknowledged is being fired.
DEEPDEBU G	<i>Firing MessageAcknowledgeTimeOut</i>	MessageAcknowledgeTimeOut is being fired.
DEEPDEBU G	<i>Firing MessageCreated</i>	MessageCreated is being fired.
DEEPDEBU G	<i>Firing MessageDeleted</i>	MessageDeleted is being fired.
DEEPDEBU G	<i>Firing MessageIn: (Identifier:'%',Message:'%')</i>	Message is being fired, Id value is %, MessageText is %.
DEEPDEBU G	<i>Firing MessageNegAcknowledged</i>	MessageNegAcknowledged is being fired.
DEEPDEBU G	<i>Firing MessageSendError</i>	MessageSendError is being fired.
DEEPDEBU G	<i>Firing MessageSent</i>	MessageSent is being fired.
DEEPDEBU G	<i>Last index set to %d.</i>	The last applied index was set to [value].
DEEPDEBU G	<i>MessageAcknowledged fired</i>	MessageAcknowledged was fired.
DEEPDEBU G	<i>MessageAcknowledgeTimeOut fired</i>	MessageAcknowledgeTimeOut was fired.
DEEPDEBU G	<i>MessageChangedNotify received for Message %s ignored (no longer Mainstation)!</i>	The project received a changed notification for message with GUID [number] from Message Control, however, it is no longer the process-executing instance.
DEEPDEBU G	<i>MessageCreated fired</i>	MessageCreated was fired.
DEEPDEBU G	<i>MessageDeleted fired</i>	MessageDeleted was fired.
DEEPDEBU G	<i>MessageIn fired: (Identifier:'%',Message:'%')</i>	Message was fired, Id value is %, MessageText is %.

Level	Entry	Description
DEEPDEBU G	<i>MessageNegAcknowledged fired</i>	MessageNegAcknowledged was fired.
DEEPDEBU G	<i>MessageSendError fired</i>	MessageSendError was fired.
DEEPDEBU G	<i>MessageSent fired</i>	MessageSent was fired.
DEEPDEBU G	<i>Messaging TimeOut watchdog activated for Project '%s'</i>	For the project [name] the TimeOut supervision was started.
DEEPDEBU G	<i>Messaging-Queue is being saved.</i>	The current message queue will be saved.
DEEPDEBU G	<i>Project %s signed off.</i>	The project is logged off from the engine.
DEEPDEBU G	<i>Project %s signed on.</i>	The project is logged on to Message Control.
DEEPDEBU G	<i>SB state: %s(%u)</i>	The status of the standby server has changed and is now %s (%d).
DEEPDEBU G	<i>Sending added notification for message %s to SB.</i>	The message object with GUID [number] was created. The "new" object is also sent to the standby server.
DEEPDEBU G	<i>Sending changed notification for message %s to SB.</i>	The message object with GUID [number] has changed, the "new" object is sent to the standby server.
DEEPDEBU G	<i>Sending MessageCtrl-StateChanged to SB.</i>	A MessageControl-StateChange telegram is sent to the standby server.
DEEPDEBU G	<i>Sending remove notification for message %s to SB.</i>	The message object with GUID [number] was removed; a remove telegram is sent to the standby server.
DEEPDEBU G	<i>Switch to server:%s</i>	The current instance is upgraded to server.
DEEPDEBU G	<i>Switch to standby:%s</i>	The current instance is downgraded to standby server.
DEEPDEBU G	<i>The message %s is added to the Queue.</i>	The message with GUID [number] was added to the queue.

Level	Entry	Description
DEEPDEBUG	<i>The Message (%s) was removed from the project queue.</i>	The message with GUID [number] was removed from the queue.

### 9.2.1.3 MSG

#### LOG LEVEL MSG & MSG

Level	Entry	Description
logLe_MSG	<i>Logging on to default mapi profile %s</i>	If no profile was selected, the sending mode identifies the first profile and logs on with it.
MSG	<i>Logging off of MAPI profile %s.</i>	Log-off from profile [name].
MSG	<i>Logging on to mapi profile %s</i>	Log-on on profile [name].
MSG	<i>MessageControl has been activated.</i>	Message Control was activated.
MSG	<i>MessageControl has been deactivated.</i>	Message Control was deactivated.
MSG	<i>Selected dispatcher '%s' is busy-&gt;try again...</i>	The selected sending mode [type] is busy: Another attempt will be made later.
MSG	<i>The incoming message for %s contains the correct Ack-Code.</i>	The incoming message [ID] contains a valid PIN code for confirming the message.
MSG	<i>The incoming message for %s contains the correct NegAck-Code.</i>	The incoming message [ID] contains a valid NA code for rejecting the message.
MSG	<i>The incoming message for %s does not contain a valid code.</i>	The incoming message [ID] does not contain a valid code.

### 9.2.1.4 Warnings

#### WARNINGS

Level	Entry	Description
WARNING S	<i>A Send-Message function was executed for a Group which does not contain any Message Control users!</i>	A sending-message function was executed for a group which is empty or does not contain activated users for Message Control.
WARNING S	<i>Index exceeded 99999, starting from 0.</i>	Maximum index for file name reached, starts again with 0.
WARNING S	TAPI returned [number] available devices.	<p>Number of modems that can be addressed via TAPI.</p> <p>This message is created in the LOG if the devices available from zenon will be determined and displays which devices can be addressed by the OS by means of TAPI.</p>
WARNING S	Not all required modes are supported: LINEBEARERMODE_VOICE:[Mode mID],LINEMEDIAMODE_AUTOMATEDVOICE:[Modem ID],LINEFEATURE_MAKECALL:[Modem ID]-> the device [Modem name] can't be used!	<p>Once the number of TAPI devices has been determined, these are filtered for the necessary modes.</p> <p><b>Note:</b> If a device does not support one of the required modes, this mode is not shown. This log entry displays clearly which mode does not exist.</p>

### 9.2.2 E-mail via SMTP/POP

The listing of these LOG entries is ordered alphabetically according to levels and entries.

- ▶ Debug
- ▶ Error (on page 121)
- ▶ Warning (on page 122)

### 9.2.2.1 Debug

#### DEBUG

Level	Entry	Description
DEBUG	<i>Attaching to current call.</i>	The sending mode attaches to the active call.
DEBUG	<i>Connected</i>	A connection has been established.
DEBUG	<i>Couldn't send MessageDeleteRequest notification to Mainstation.</i>	The attempt to send a delete request telegram to the process-executing computer has failed.
DEBUG	<i>Current Message %Nachricht Inhalt%</i>	The current message (content, settings, etc.) is traced in the LOG.
DEBUG	<i>File alignment request could not be send to mainstation.</i>	The attempt to send an alignment request telegram to the process-executing computer has failed.
DEBUG	<i>Invalid code '%s' entered.</i>	An invalid code was entered.
DEBUG	<i>Mail to Benutzer '%s' was successfully added to the outbox.</i>	The message to user [name] was successfully transferred to the Outlook outbox.
DEBUG	<i>MessageDeleteRequest Response: %d messages were deleted from the queue.</i>	A delete response was received: [Number] messages were removed from the queue.
DEBUG	<i>Messaging canceled: The end of the Benutzer list for the Group (%s) has been reached.</i>	The end of user group [name] was reached, the sending is canceled.
DEBUG	<i>Requesting File alignment from mainstation.</i>	An alignment request is sent to the process-executing computer.
DEBUG	<i>Send SMS to Benutzer %s.</i>	A SMS was sent to the user [address].
DEBUG	<i>Sending mail to Benutzer '%s' using address: %s.</i>	A message for the user [name] is prepared and [sent] to the address.
DEBUG	<i>Sending MessageDeleteRequest for %d messages to Mainstation.</i>	A delete request for [number] messages is sent to the process-executing computer.
DEBUG	<i>SMS to Benutzer %s transmitted to the outbox.</i>	A SMS was transferred to the user [address] in the outbox.
DEBUG	<i>The #-sign was entered:</i>	# was entered: the last message is repeated.

Level	Entry	Description
	<i>Repeating the last message.</i>	
DEBUG	<i>The *-sign was entered: Clearing the current key.</i>	A * was entered, the current entered code is being reset.
DEBUG	<i>The current call be terminated now.</i>	A call is active, it is closed now.
DEBUG	<i>The is no call in progress, nothing to terminate.</i>	No call is active, nothing can be closed.
DEBUG	<i>The line %s has been closed.</i>	The line [] has successfully been closed.
DEBUG	<i>The line %s has been opened.</i>	The line [] has successfully been opened.
DEBUG	<i>The message %s should have been acknowledged by now, handling TimeOut.</i>	The message [ID] should already have been confirmed, TimeOut is being processed.
DEBUG	<i>The message %s should have been sent by now, handling hardware TimeOut.</i>	The message [ID] should already have been sent, TimeOut is being processed.
DEBUG	<i>The Message %s was NOT successfully sent to %s!</i>	The message [ID] could not be sent to [recipient].
DEBUG	<i>The Message %s was not successfully sent to %s, and will be send again!</i>	The message [ID] could not be sent to [recipient] and will be sent again.
DEBUG	<i>The Message %s was successfully sent to %s!</i>	The message [ID] was successfully sent to [recipient].
DEBUG	<i>The next user (%s) for the group (%s) will be notified, previous Benutzer was:%s</i>	The next user [name] from user group [name] is notified, the previous user was user [name].
DEBUG	<i>The sending of message %s was completed.</i>	The sending process for message [ID] has been completed.
DEBUG	<i>Valid Acknowledge-code entered.</i>	A valid PIN code for confirming the message was entered.
DEBUG	<i>Valid NegAcknowledge-code entered.</i>	A valid NA code for rejecting the message was entered.



## 9.2.2.2 Error

### ERROR

Level	Entry	Description
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	Creating Connection failed. Errorcode: [Code]	The object for a connection (both secure and plain text) could not be created.
Error	Creating SSL/TLS Method failed. Errorcode: [Code]	The SSL/TLS method required for establishing a secure connection could not be created.
Error	No Destination E-Mail-Address specified	In case of an outgoing e-mail no target address was entered.
Error	Opening Connection failed. Errorcode: [Code]	A connection (both secure and plain text) could not be established.
Error	Reading SSL Structure failed. Errorcode: [Code]	Reading the SSL/TLS structure of a secure connection failed.
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 required commands for an EHLO response listed in the list of supported extension commands (AUTH LOGIN and AUTH PLAIN, the latter is the minimum implementation in accordance with RFC 2554).
Error	The Server did not respond an OK Code to HELO. Response: [response from the SMTP server]	The outgoing mail server did not respond to the EHLO command or the 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 a Waiting-For-Data-Response to	The outgoing mail server has not sent the expected response to the AUTH PLAIN

Level	Entry	Description
	the AUTH PLAIN command: [response from the SMTP server]	command. (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 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	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.)

### 9.2.2.3 Email via SMTP/POP - warning

#### WARNING

Level	Entry	Description
<i>WARNING</i>	<i>No Subject for the E-Mail specified</i>	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.
<i>WARNING</i>	<i>No Text for the E-Mail specified</i>	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.
<i>WARNING</i>	<i>The Server did not respond an OK Code to EHLO. Response:</i> [response from the 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.

### 9.2.3 Voice over IP

LOG entries for VoIP:

#### ERRORS

Level	Entry	Description
<i>ERRORS</i>	<b>VoIP:Error in VoIP connection '%s'</b>	<p>Error message from the library for the connection.</p> <ul style="list-style-type: none"> <li>▶ % is replaced by the current code text for the SIP status or a specific error text. For example: <b>VoIP:Error in VoIP connection 'User not found'</b>.</li> </ul>
<i>ERRORS</i>	<b>VoIP:Error during hardware initialization</b>	An error has occurred during the initialization of the hardware.
<i>ERRORS</i>	<b>VoIP:Error when trying to make a call</b>	An error occurred during the attempt to carry out a Voice over IP call.

### 9.2.4 SMS

#### ERROR

Level	Entry	Description
ERROR	[telephone number] is not a valid SMSC telephone number!	<p>The inserted telephone number of the short message center is invalid. Has the country code (00xy or +xy) been set correctly?</p> <p>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!</p>
ERROR	[telephone number] is not a valid telephone number!	<p>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!</p>

Level	Entry	Description
ERROR	Error on opening the port [COM-Port]: [error number 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-Port] [state / timeouts / buffers]: [error number 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 ([number] bytes) is too small for the [number] 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 of 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 command] command: [error message]	<p>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.</p> <p>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.</p>
ERROR	The modem did not respond to the [AT command] 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	An attempt was made to send, receive or delete an SMS while no connection with the modem could be established beforehand.

Level	Entry	Description
ERROR	The modem returned an error to the transmission of the PDU: [error message]	The modem responded with an error message to the transmission of a 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 a SMS-PDU a timeout occurred.
ERROR	Error on writing data: [error number 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: [error number 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 [telephone number] contains the not supported character [character]	The telephone number of the short message center of the sender received during the SMS receipt contains an invalid character.
ERROR	The source telephone number [telephone number] contains the not supported character [character]	The telephone number of the sender received during the SMS receipt contains an invalid character.
ERROR	The syntax of the timestamp [time stamp] 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 as 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 as HEX] is not supported.	The formatting of an incoming SMS is not supported.
ERROR	Could not convert [byte as HEX]	A character contained in the 7-bit-SMS data

Level	Entry	Description
	from GSM 7-Bit-Alphabet. Baselanguage: [byte as HEX], Extensionlanguage [byte as HEX]	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 multilanguage fragmented SMS!	An SMS was received in several parts, the individual 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.	<p>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.</p> <p><b>Attention:</b> Incorrect entering of the PUK may render the SIM card useless!</p>
ERROR	The Modem responded with an unknown PIN Status: [response from the modem]	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

Level	Entry	Description
		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 [telephone number]	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 [telephone number]	An SMS was successfully sent to the indicated telephone number.
DEBUG	Successfully received SMS from [telephone number]	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.

### 9.2.5 LOG entries - 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

**ERROR MESSAGES IN THE DIAGNOSIS VIEWER.**

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]: <i>[response from the SMTP server]</i>	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: <i>[response from the SMTP server]</i>	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: <i>[response from the SMTP server]</i>	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: <i>[response from the SMTP</i>	The SMTP server has not confirmed the AUTH PLAIN data with OK. The login has failed. The



Level	Entry	Description
	<i>server]</i>	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	<p>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.</p> <p>The following commands are required: AUTH LOGIN and AUTH PLAIN, the latter is the minimum implementation according to RFC 2554.</p>
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: <i>[response from the SMTP server]</i>	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: <i>[response from the SMTP server]</i>	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: <i>[response from the SMTP server]</i>	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-Response to the transmitted Data: <i>[response from the SMTP server]</i>	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.

Level	Entry	Description
ERROR	The Server did not send an OK-Response to the [POP3-Kommando] command: [response from the POP3 server]	<p>The incoming mail server has not confirmed a command with OK. The response is attached to the LOG message.</p> <p>The following commands might be affected: APOP, USER, PASS, STAT, RETR, DELE.</p> <p><b>Note:</b> 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.</p>
ERROR	The OK-Response to the STAT command did not have the expected Format: [response from the POP3 server]	<p>The response of the incoming mail server to the STAT command did not correspond to the format definition according to RFC 1939.</p> <p>Format: <b>OK&lt;space&gt;[number of mails in mailbox] &lt;space&gt;[total size of all mails in mailbox in bytes]</b></p>
ERROR	[Sending / Receiving] Data failed. Errorcode: [Code]. Errormessage: [String]	Sending or receiving data via an OpenSSL network connection has failed.