zenon
by COPA-DATA

Manuel de zenon

Message Control

v.8.10

Contenu

1 Bienvenue dans l'aide de COPA-DATA ... 5

2 Message Control ... 5

3 Compatibility with version 6.xx ... 6

3.1 Compatibility with version 5.50 ... 8

4 Requirements ... 9

5 Limitations ... 10

6 Supported AT commands ... 11

7 Configure Message Control ... 13

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 ... 51

7.5 Functions ... 55

7.5.1 Screen switching to a screen of type Message Control ... 55

7.5.2 Save current queue ... 60

7.5.3 Group/class/area/equipment suppressed ... 60

7.5.4 Send a Message ... 65

7.5.5 Send Message: activate ... 87

7.5.6 Send Message: deactivate ... 87

7.6 Parameters for messages ... 88

7.6.1 Texte de valeur limite et texte libre ... 89

8 Message Control in the Runtime ... 94

8.1 Connect screen of the type Message Control ... 96

8.2 Acquittement de messages ... 99

8.3 Voice messages in Runtime ... 103

8.4 Network 104
8.5 Voice message process 106

9 Messages and error handling 109
9.1 Check list 111
9.2 LOG entries 114
9.2.1 Miscellaneous messages 114
9.2.2 E-mail via SMTP/POP 121
9.2.3 Voice over IP 126
9.2.4 SMS 126
9.2.5 LOG entries - OpenSSL library 130

1 Bienvenue dans l'aide de COPA-DATA

TUTORIELS VIDÉO DE ZENON.

Des exemples concrets de configurations de projets dans zenon sont disponibles sur notre chaîne YouTube (https://www.copadata.com/tutorial_menu). Les tutoriels sont regroupés par sujet et proposent un aperçu de l'utilisation des différents modules de zenon. Les tutoriels sont disponibles en anglais.

AIDE GÉNÉRALE

Si vous ne trouvez pas certaines informations dans ce chapitre de l'aide ou si vous souhaitez nous suggérer d'intégrer un complément d'information, veuillez nous contacter par e-mail : documentation@copadata.com.

ASSISTANCE PROJET

Si vous avez besoin d'aide dans le cadre d'un projet, n'hésitez pas à adresser un e-mail à notre service d'assistance : support@copadata.com

LICENCES ET MODULES

Si vous vous rendez compte que vous avez besoin de licences ou de modules supplémentaires, veuillezcontacter l'équipe commerciale par e-mail : E-mail 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 **Module 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 **Module Message Control** of the workspace (sending) and properties **Paramètres spécifiques au projet** 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 (à la 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 (à la page 109).
- **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 (à la page 109) 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 (à

la page 109) 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 assigend to groups which have to be further configured.

EXPORT SETTINGS

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

Attention: By executing the property **Exporter les paramètres dans 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 (à la page 6) chapter.

4 Requirements

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

Medium	Requirement
E-mail message via Outlook (à la page 16):	Running Microsoft Outlook instance
E-mail message via SMTP (à la page 17):	The Runtime server executing the sending must be connected to a network. An e-mail account must be available. The servers with access to the account must be available.
SMS via GSM (à la page 19):	A GSM modem must be connected to the computer and must be approachable via a serial port (COM port). usually USB modems create a virtual COM port during the installation through which they can be approached.
SMS message via SMS gateway (à la page 22):	Compatible suitable SMS server.
Speech output: (à la page 23)	Voice modem with DTMF functionality or voice over IP server. Text-to-speech engine.
Telephone: (à la page 23)	Voice modem with DTMF functionality. If an ISDN adapter is used it must support the following functions. ▸ Support of voice messages (LINEMEDIAMODE_AUTOMATEDVOICE) ▸ Support of DTMF signals (for receiving the PIN code when acknowledging a message) ▸ Support of Telephony Application Programming Interface

Medium	Requirement
	(TAPI)
Voice over IP: (à la page 23)	Voice over IP Access. Les conditions suivantes doivent être satisfaites pour que le service VoIP soit utilisable : ▸ Le fournisseur de service VoIP doit prendre en charge les protocoles **SIP** et **RTP**. ▸ Les ports correspondants doivent être ouverts au niveau du pare-feu : SIP (Par défaut : *5060*) RTP (Par défaut : *4000*) RTCP (Par défaut : *4001)*

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 **Voix par IP** group.

Ranges:

- SIP: *0 - 65535*
- RTP: *0 - 65534*

Default values:

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

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:	*AT+CMGF?\r*	*\r\nOK\r\n*	at the end
Check to see if the modem has set the given SMS message center:	*AT+CSCA?\r*	*\r\nOK\r\n*	at the end
Send message to this number. Modem signalizes *Ready for PDU* if the command was accepted:	*AT+CMGS="[PDU length]"\r*	*\r\n>[Space]*	
PDU for the SMS to be sent to the number specified previously with completion flag:	*[SMS-PDU][0x1A]*	*\r\nOK\r\n*	at the end
List SMSs from the memory with status set by flag:	*AT+CMGL=[Flag]\r*	*\r\nOK\r\n*	at the end
Delete all SMSs with status set by flag from the memory (regardless of index):	*AT+CMGD=0,[Flag]\r*	*\r\nOK\r\n*	at the end
Delete SMS with the given index from the memory:	*AT+CMGD=[Index]\r*	*\r\nOK\r\n*	at the end

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

Arguments:

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

ERROR 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 (à la page 109) chapter.

7 Configure Message Control

Message Control is configured via:

- global properties (à la page 15) of the media for sending
- project-specific properties (à la page 44) for the message
- Functions (à la page 55) 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 **Module Message Control**.
   Note: You can find this properties group if you click on the current **Espace de travail** in the zenon Editor.
2. To do this, activate the **Type d'envoi actif** property in the area of the properties group that represents the desired dispatch type.
   Note: The **Type d'envoi actif** property can be activated for several types of dispatch.
3. Configure (à la page 15) the medium.
4. Configure (à la page 44) the project-specific properties.
5. Configure the users (à la page 51) or user groups that are to be reached via the medium.
6. Create a Send message (à la page 65) function and

   link the function:

   a) with a button or

   b) with the property **Exécuter instantanément la fonction** 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 **Envoi d'un 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:
  - **Sujet (ID)** in the global settings for Outlook
  - **Sujet pour les e-mails sortants** in the global settings for e--mail message via SMTP
  - **Partie dynamique du sujet** 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 (à la page 23)

    or
  - Voice message by audio file (à la page 23)
- Message text: defined for e-mails, SMS and voice messages in:
  - the Option (à la page 72) **Constant Text**

CONFIGURE MANAGEMENT OF THE MESSAGE QUEUE

In order to administrate messages in the Runtime:

1. create a screen of type Message Control (à la page 46)
2. create and configure a function Screen switch (à la page 55) 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
   - **Actif lors du démarrage du runtime** property: Once this property is activated Message Control will automatically start in Runtime.
   - Function Send Message: activate (à la page 87): 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 (à la page 60) function. Message Control can be closed with the function Send Message: deactivate (à la page 87) 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** (à la page 65) 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 (à la page 60) in the Runtime. Alarms and equipment can be activated or deactivated. Suppressed entries are logged in the CEL if the property **Enregistrer dans liste d'événements** is configured with *all confirmations* or *only negative confirmations*.

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

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 (à la page 16)
- E-mail message via SMTP (à la page 17)
- SMS message via GSM modem (à la page 19)
- SMS message via SMS gateway (à la page 22)
- Voice message audiofile via modem (à la page 25)
- Voice message text-to-speech via modem (à la page 26)
- Voice over IP as audio file (à la page 28)
- Voice over IP as text-to-speech (à la page 29)

In order to configure the properties:

1. Highlight the **Espace de travail**
2. Select the **Module 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] (à la page 31).

> **Attention**
>
> These properties of the **Module 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 **Versandart aktiv**
- 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 (à la page 84)for acknowledgement. This period of time is defined with the function **Send message** (tab acknowledgement (à la page 84)).

> **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 **Type d'envoi actif**.
   Note: You can find this property in the **Message e-mail (Outlook)** area of the **Module Message Control** property group.
2. Define the name of the Outlook profile to be used via the property **Profil**.
3. Define an unambiguous subject via the property **Sujet (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 (à la page 44) the properties for **Paramètres spécifiques au projet**.
5. Configure Users (à la page 51) and user groups.
6. Create required functions (à la page 55) 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 **Type d'envoi actif**
- 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 (à la page 84)for acknowledgement. This period of time is defined with the function **Send message** (tab acknowledgement (à la page 84)).

CONFIGURATION

1. Activate the sending mode via the property **Type d'envoi actif**.
   Note: You can find this property in the **Paramètres e-mail (SMTP)** area of the **Module Message Control** property group.
2. Configure the properties for:
   - **Serveur de mails entrants**
   - **Serveur de mails sortants**
3. Configure (à la page 44) the properties for project-specific settings.
4. Configure Users (à la page 51) and user groups.
5. Create required functions (à la page 55) 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 **Encryptage**.
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 **Laisser les mails lus sur le serveur** 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 **Type d'envoi actif**
- 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 **Type d'envoi actif**.
   Note: You can find this property in the **Message SMS (modem GSM)** area of the **Module Message Control** property group.
2. Select the COM port of the modem from the drop-down menu via the property**Connexion modem (série)**.
   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 **Code PIN** enter the PIN for the modem, if required.
4. Check the connection by clicking **Vérification connexion**.
5. Enter the telephone number for the short message center in the property field **Numéro du centre SMS**.
6. Configure (à la page 44) the properties for **Paramètres spécifiques au projet**.
7. Configure Users (à la page 51) and user groups.
8. Create required functions (à la page 55) 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
Timeout for AT commands	*10000 ms*	The modem has 10 seconds to respond to the transmission of an AT command.
SMS polling interval	*20000 ms*	Every 20 seconds it is checked if new SMS have arrived and they are being read. If no incomplete multi-part SMS are present anymore all messages which were read are deleted.
COM timeouts	Write: *5 seconds* read: *immediate return*	5 seconds are available for writing the bytes on the COM port (max. 400 in case of SMS-PDU with maximum size). During the reading of the COM port the current content of the receive buffer is read and immediately returned.

Entry	Value	Description
COM state configuration	▸ **Baud:** *9600* ▸ *Binary mode* ▸ **Parity check:** *Inactive* ▸ **Clear-to-Send** and **Data-Set-Ready:** *not controlled* ▸ **Data-Terminal-Ready** and **Request-To-Send Flow Control:** *activated* ▸ **Bit per byte:** *8* ▸ **Parity:** *None* ▸ **Stop Bit:***1*	Standard configuration of a serial port.

BUFFER

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

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 **Type d'envoi actif**
- 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 **Type d'envoi actif**.
   Note: You can find this property in the **Message SMS (passerelle SMS)** area of the **Module Message Control** property group.
2. Configure **Dossier de boîte d'envoi**, **Dossier de boite d'envoi** and **Dossier d'état**.
3. Configure the **Première lettre du fichier SMS**.
   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 **Première lettre du fichier 'lock/semaphore'**: 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 **Gestion d'alarmes intelligente**.
   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 (à la page 44) the properties for **Paramètres spécifiques au projet**.
7. Configure Users (à la page 51) and user groups.
8. Create required functions (à la page 55) 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 **Gestion d'alarmes intelligente** 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 (à la page 25)
- Text-to-speech via modem (à la page 26)

- Voice over IP as audio file (à la page 28)
- Voice over IP as text-to-speech (à la page 29)

You can read information about the process in the Runtime in the subchapters and in the **Voice messages in Runtime** (à la page 103) 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 (Par défaut : *5060*)
  - RTP (Par défaut : *4000*)
  - RTCP (Par défaut : *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 **Modem vocal (fichier audio)**
- 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 **Délai d'attente entre deux tentatives et délai d'attente avant expiration** property must be greater than the value for the **Waiting time** option in the **Envoi d'un message** functions (**Receipt Confirmation** tab).

PROCEDURE

CONFIGURATION

1. Activate the sending mode via the property **Modem vocal (fichier audio)**.
2. Configure the settings for **Time-out [min]** and **Répéter le texte d'accueil**.
3. Select the modem for sending via property **Nom ligne**.
   Selection from a drop-down menu which lists all modems configured in the system control.
4. Configure (à la page 44) the properties for **Paramètres spécifiques au projet**.
5. Configure Users (à la page 51) and user groups.
6. Create required functions (à la page 55) 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 (à la page 99) by the recipient with the respectively valid code:
   - **Code PIN** for confirmation
   - **Code NA** 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 **Modem vocal (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 **Délai d'attente entre deux tentatives et délai d'attente avant expiration** property must be greater than the value for the **Waiting time** option in the **Envoi d'un 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 **Voix :** property.

PROCEDURE

CONFIGURATION

1. Activate the sending mode via the property **Modem vocal (Text-to-Speech)**.
2. Configure the settings for **Time-out [min]** and **Répéter le texte d'accueil**.
3. Select the modem for sending via property **Nom ligne**.
   Selection from a drop-down menu which lists all modems configured in the system control.
4. Configure the properties for text-to-speech:
   - **Voix :**
   - **Vitesse speech**
   - **Volume**
5. Configure (à la page 44) the properties for **Paramètres spécifiques au projet**.
6. Configure Users (à la page 51) and user groups.
7. Create required functions (à la page 55) 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.
   - **Code PIN** for confirmation
   - **Code NA** 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 **Voix par IP (Fichier audio)**
- 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 (Par défaut : *5060*)
  - RTP (Par défaut : *4000*)
  - RTCP (Par défaut : *4001)*

PROCEDURE

CONFIGURATION

1. Activate the sending mode via the property **Voix par IP (Fichier audio)**.
2. Configure the settings of the **Voix par IP** group:
   - **Adresse du serveur**
   - **Identifiant utilisateur**
   - **Mot de passe**
3. Configure (à la page 44) the properties for **Paramètres spécifiques au projet**.
4. Configure Users (à la page 51) and user groups.
5. Create required functions (à la page 55) 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 (à la page 99) by the recipient with the respective valid code:
   - **Code PIN** for confirmation
   - **Code NA** 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 **Voix par 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 (Par défaut : *5060*)
  - RTP (Par défaut : *4000*)

- RTCP (Par défaut : *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 **Voix :** property.

PROCEDURE

CONFIGURATION

1. Activate the sending mode via the property **Voix par IP (Text-to-Speech)**.
2. Configure the properties of the **Voix par IP** group.
   - **Adresse du serveur**
   - **Identifiant utilisateur**
   - **Mot de passe**
3. Configure the properties for text-to-speech:
   - **Voix :**
   - **Vitesse speech**
   - **Volume**
4. Configure (à la page 44) the properties for **Paramètres spécifiques au projet**.
5. Configure Users (à la page 51) and user groups.
6. Create required functions (à la page 55) 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.
   - **Code PIN** for confirmation
   - **Code NA** 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 (à la page 15) and project-specific (à la 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:

Entrée	Description
[MESSAGE CONTROL]	Paramètres du module Message Control. Recommandation : Configuration via les propriétés du groupe **Module Message Control** dans l'espace de travail, avec trois sous-groupes et les propriétés du groupe **Paramètres spécifiques au projet** dans le noeud **Message Control** de l'arborescence des projerts.
GSM_SMS=	Activation du type d'envoi SMS via GMS. - *0* : actif - *pas 0* : inactif

Entrée	Description
	Par défaut : *0* L'entrée dans Message32.ini correspond à l'entrée **[GSM] On** Ceci est pris en compte lors de l'importation/exportation des paramètres ini. Correspond à la propriété **Type d'envoi actif**, dans le groupe **Message SMS (modem GSM)**, dans Editor.
GSM_SMS_COM=	Port COM utilisé pour établir la connexion au modem. Par défaut : *vide* Pour cela, activez la propriété **Connexion modem (série)** dans Editor.
GSM_SMS_PIN=	Code PIN utilisé pour l'authentification auprès du modem. Par défaut : *vide* Pour cela, activez la propriété **Code PIN** dans Editor.
GSM_SMS_SMSC=	Numéro de téléphone du centre de messagerie du fournisseur GSM. Par défaut : *vide* Pour cela, activez la propriété **Numéro du centre SMS** dans Editor.
GSM_BULK_DELETE=	Comportement lors de la suppression d'un SMS : ▸ *0* : Les messages sont supprimés avec la méthode **Index**. Les SMS lus sont supprimés individuellement ▸ *1* : Les messages sont supprimés avec la méthode **Statusflag**. Tous les SMS lus sont supprimés en même temps. **Remarque** : Cette méthode est rapide et plus puissante, mais elle n'est pas prise en charge par tous les modems. Par défaut : *0* Pour cela, activez la propriété **Supprime SMS avec la syntaxe d'état de la commande AT+CMGD** dans Editor.

Entrée	Description
Outlook=	Notification par e-mail via Outlook : ▸ *0* : inactif ▸ *1* : actif Par défaut : *0* Correspond à la propriété **Type d'envoi actif**, dans le groupe **Message e-mail (Outlook)**, dans Editor.
Outlook_Profile=	Nom du profil Outlook utilisé pour l'envoi. Par défaut : *vide* Pour cela, activez la propriété **Profil** dans Editor.
POP_APOP=	Contrôle l'authentification au niveau du serveur de mails entrants avec messagerie via SMTP/POP. ▸ *0* : L'utilisateur (**USER**) et le mot de passe (**PASS**) sont utilisés pour l'authentification. ▸ *1* : La commande **APOP** est utilisée pour l'authentification au lieu de l'utilisateur et du mot de passe. Par défaut : *0* Correspond à la propriété **Utiliser APOP pour l'authentification**, dans le groupe **Paramètres e-mail (SMTP)**, dans Editor.
POP_KEEP_MAILS=	Définit si les e-mails lus restent sur le serveur. ▸ *0* : Les e-mails sont supprimés du serveur après téléchargement. ▸ *1* : Les e-mails restent après téléchargement. Par défaut : *0* **Remarque :** puisque le protocole POP3 n'offre pas la possibilité de filtrer les messages lors de leur téléchargement, tout le courrier entrant est systématiquement téléchargé. Si cette propriété est active, l'utilisation de la mémoire sur le serveur augmente et les performances peuvent diminuer lors de la récupération des e-mails.

Entrée	Description
	Pour cela, activez la propriété **Laisser les mails lus sur le serveur** dans Editor.
POP_PASSWORD=	Mot de passe de connexion au serveur de courrier entrant (POP3). Le mot de passe est enregistré sous forme chiffrée, et est uniquement déchiffré aux fins de l'authentification. Doit uniquement être configuré dans Editor. Par défaut : *vide* Correspond à la propriété **Mot de passe**, dans le groupe **Serveur de mails entrants**, dans Editor.
POP_POLL_INTERVALL=	Intervalle minimum entre deux demandes POP3, en secondes. La valeur ne doit pas être inférieure à la valeur de consigne du serveur POP3. ▸ Minimum : *10* ▸ Maximum : *4294967295* Par défaut : *60* Pour cela, activez la propriété **Temps minimum entre deux requêtes (s)** dans Editor.
POP_PORT=	Port d'adressage du serveur POP3. ▸ Valeur maximale : *6553* Par défaut pour : ▸ Connexion non sécurisée : *110* ▸ Connexion sécurisée : *995* Correspond à la propriété **Port**, dans le groupe **Serveur de mails entrants**, dans Editor.
POP_SECURITY=	Type de protection de la connexion au serveur POP3. ▸ *0* : pas de sécurité ▸ *1* : SSLv2 et SSLv3 ▸ *2* : TLSv1 Par défaut : *0* Correspond à la propriété **Encryptage**, dans le groupe

Entrée	Description
	Serveur de mails entrants, dans Editor.
POP_SERVER=	Adresse du serveur POP3. Par défaut : *vide* Correspond à la propriété **Serveur de mails**, dans le groupe **Serveur de mails entrants**, dans Editor.
POP_USER=	Nom d'utilisateur du serveur entrant. Par défaut : *vide* Correspond à la propriété **Identifiant utilisateur**, dans le groupe **Serveur de mails entrants**, dans Editor.
SMSGateway=	Notification par SMS via passerelle SMS : ▸ *0* : inactif ▸ *1* : actif Par défaut : *0* Correspond à la propriété **Type d'envoi actif**, dans le groupe **Message SMS (passerelle SMS)**, dans Editor.
SMSGateway_Inbox=	Dossier de réception du serveur de SMS pour l'envoi de SMS. Par défaut : *vide* Pour cela, activez la propriété **Dossier de boite d'envoi** dans Editor.
SMSGateway_OriginId=	Identification de l'expéditeur pour l'envoi de SMS. Par défaut : *vide* Pour cela, activez la propriété **ID expéditeur** dans Editor.
SMSGateway_Outbox=	Dossier d'envoi du serveur de SMS pour l'envoi de SMS. Par défaut : *vide* Pour cela, activez la propriété **Dossier de boîte d'envoi** dans Editor.
SMSGateway_Prefix=	Première lettre du fichier SMS. Doit être unique à chaque projet.

Entrée	Description
	Longueur : 1 caractère Par défaut :*F* Pour cela, activez la propriété **Première lettre du fichier SMS** dans Editor.
SMSGateway_SemaphorPrefix=	Première lettre du fichier de verrouillage. Longueur : 1 caractère Par défaut :*S* Pour cela, activez la propriété **Première lettre du fichier 'lock/semaphore'** dans Editor.
SMSGateway_SmartAlarm=	la gestion intelligente des alarmes est utilisée. Doit être unique à chaque projet. ▸ *0* : inactif ▸ *1* : actif Par défaut : *0* Pour cela, activez la propriété **Gestion d'alarmes intelligente** dans Editor.
SMSGateway_Statusbox=	Dossier des messages d'état de la fonction d'envoi de SMS. Par défaut : *vide* Pour cela, activez la propriété **Dossier d'état** dans Editor.
SMSGateway_TimeOut=	Délai d'attente en minutes des messages envoyés. Définit le délai au terme duquel un message est considéré comme n'ayant pas été correctement envoyé. Par défaut : *60* Pour cela, activez la propriété **Timeout** dans Editor.
SMTP_AUTH=	Authentification sur le serveur de courrier sortant. ▸ *0* : pas de sécurité ▸ *1* : connexion au serveur POP3 avant l'envoi ▸ *2* : SMTP AUTH avec connexion au serveur SMTP avant l'envoi

Entrée	Description
	Par défaut : *0* Correspond à la propriété **Auhentification pour le serveur de mails sortants**, dans le groupe **Serveur de mails sortants**, dans Editor.
SMTP_OTHER_CREDS=	Définit si le serveur de courrier sortant utilise des données de connexion différentes de celles du serveur de courrier entrant. ▸ *0* : inactif ▸ *1* : actif Par défaut : *0* Corresponds à la propriété dans le groupe **Serveur de mails sortants** dans Editor.
SMTP_PASSWORD=	Données hexadécimales du mot de passe chiffré d'authentification sur le serveur de courrier sortant. Par défaut : *vide* Corresponds à la propriété dans le groupe **Serveur de mails sortants** dans Editor.
SMTP_OTHER_CREDS=	Données de connexion pour le serveur SMTP. ▸ *0* : les données de connexion du serveur de courrier entrant sont également utilisées sur le serveur de courrier sortant. ▸ *1* : pour l'authentification sur le serveur de courrier sortant, des données d'authentification différentes de celles du serveur de courrier entrant sont utilisées. Par défaut : *0* Correspond à la propriété **Utiliser un 'log' pour les données différent de celui du serveur de mails entrants**, dans le groupe **Serveur de mails sortants**, dans Editor.
SMTP_OUT_ADDR=	Adresse pour les messages envoyés. Par défaut : *vide* Correspond à la propriété **Adresse pour les e-mails**

Entrée	Description
	sortants, dans le groupe **Serveur de mails sortants**, dans Editor.
SMTP_POP_MAIL=	Notification d'e-mail via SMTP/POP : ▸ *1* : actif ▸ *0* : inactif Par défaut : *0* Correspond à la propriété **Type d'envoi actif**, dans le groupe **Serveur de mails sortants**, dans Editor.
SMTP_PORT=	Définit le port utilisé sur le serveur SMTP. Maximum : *65535* Par défaut : *25* Correspond à la propriété **Port**, dans le groupe **Serveur de mails sortants**, dans Editor.
SMTP_SECURITY=	Type de cryptage de la connexion au serveur SMTP. ▸ *0* : pas de sécurité ▸ *1* : SSLv2 et SSLv3 ▸ *2* : TLSv1 Par défaut : *0* Correspond à la propriété **Encryptage**, dans le groupe **Serveur de mails sortants**, dans Editor.
SMTP_SERVER=	Serveur SMTP défini par l'utilisateur. Par défaut : *vide* Correspond à la propriété **Adresse du serveur**, dans le groupe **Serveur de mails sortants**, dans Editor.
SMTP_SRV_IS_POP=	Cette entrée définit si le serveur POP3 est utilisé en tant que serveur SMTP. ▸ *1* : les serveurs POP3 et SMTP sont identiques ▸ *0* : les serveurs POP3 et SMTP sont différents Par défaut : *0* Correspond à la propriété **Utiliser le serveur de mails**

Entrée	Description
	entrants pour les mails sortants, dans le groupe **Serveur de mails sortants**, dans Editor.
SMTP_SUBJECT=	Objet des messages sortants, permettant de détecter si un e-mail reçu sur le serveur correspond à un type d'envoi précis. Pour être traités par le système, les e-mails reçus doivent comporter ce texte. Les e-mails ne contenant pas ce sujet ne sont ni transmis au module Message Control, ni supprimés du serveur. Par défaut :**MsgCtrl_Alert:** Correspond à la propriété **Sujet pour les e-mails sortants**, dans le groupe **Serveur de mails sortants**, dans Editor.
SMTP_USER=	Nom d'utilisateur enregistré sur le serveur de courrier sortant. Par défaut : *vide* Correspond à la propriété **Identifiant utilisateur**, dans le groupe **Serveur de mails sortants**, dans Editor.
SMTP_USER_IS_ADDR=	Définit si le nom d'utilisateur utilisé pour l'authentification sur le serveur de courrier sortant est utilisé en tant qu'adresse d'expéditeur des e-mails envoyés. Utilisé uniquement l'entrée **SMTP_AUTH=** n'est pas *0*. ▸ *1* : actif ▸ *0* : inactif Par défaut : *0* Correspond à la propriété **Le nom de l'utilisateur est l'adresse pour les e-mails sortants**, dans le groupe **Serveur de mails sortants**, dans Editor.
Speech=	Notification Text-to-Speech via modem : ▸ *0* : inactif ▸ *1* : actif Par défaut : *0* Correspond à la propriété **Modem vocal**

Entrée	Description
	(Text-to-Speech), dans le groupe **Message vocal**, dans Editor.
Speech_Name=	Sélection la voix et de la langue de la synthèse vocale pour Text-to-Speech. Assurez-vous que les voix correctes ont été sélectionnées pour Editor : ▸ Editor 32 bits : *C:\Windows\SysWOW64\Speech\SpeechUX\sapi.cpl* ▸ Editor 64 bits : *C:\Windows\System32\Speech\SpeechUX\sapi.cpl* Par défaut : *vide* Correspond à la propriété **Voix :**, dans le groupe **Text-to-Speech**, dans Editor.
Speech_Rate=	Vitesse de la voix. ▸ Minimum : *-10* ▸ Maximum : *10* Par défaut : *0* Correspond à la propriété **Vitesse speech**, dans le groupe **Text-to-Speech**, dans Editor.
Speech_Volume=	Volume de la voix. Ce nombre est une valeur, en pourcentage, de la valeur maximale pour la voix sélectionnée. ▸ Maximum : *100* ▸ Minimum : *0* nombre issu du système d'exploitation, sans modification Par défaut : *0* Correspond à la propriété **Volume**, dans le groupe **Text-to-Speech**, dans Editor.
Subject=	Identifiant unique que doivent contenir les e-mails entrants pour être traités par le module de Gestion des messages. Par défaut :**MsgCtrl_Alert:**

Entrée	Description
	Correspond à la propriété **Sujet (ID)**, dans le groupe **Message e-mail (Outlook)**, dans Editor.
Telephone=	Notification via fichiers audio via modem : ▸ *0* : inactif ▸ *1* : actif Par défaut : *0* Correspond à la propriété **Modem vocal (fichier audio)**, dans le groupe **Message vocal**, dans Editor.
Telephone_IgnoreDisconnect=	Comportement en cas d'interruption de la connexion : ▸ *1* : une déconnexion (par ex., le destinataire met un terme à l'appel) est ignorée et le message est relu dans son intégralité avant la fermeture de la ligne. ▸ *0* : la transmission du message est abandonnée lors de l'interruption de la connexion. Par défaut : *0* Correspond à la propriété **Ignorer la déconnexion**, dans le groupe **Message vocal**, dans Editor.
Telephone_Line=	Entrée du modem à utiliser. Doit déjà être configuré sur l'ordinateur. Correspond à la propriété **Nom ligne**, dans le groupe **Paramètres téléphone**, dans Editor.
Telephone_LineId=	Identifiant de périphérique généré automatiquement et identifiant le modem sélectionné. En présence de plusieurs modems avec le même nom, permet de distinguer les périphériques. **Attention :** Fournie uniquement à titre informatif. Ne doit pas être modifié ici. Correspond à la propriété **ID ligne**, dans le groupe **Paramètres téléphone**, dans Editor.
Telephone_Timeout=	Délai en minutes au terme duquel une condition existante doit être annulée et fermée. Le délai doit être plus long que le délai nécessaire à la lecture et la confirmation du message.

Entrée	Description
	Par défaut : *1* Correspond à la propriété **Time-out [min]**, dans le groupe **Message vocal**, dans Editor.
Telephone_WelcomeMessageCount=	Nombre de répétitions pour le texte d'accueil. Par défaut : *5* Correspond à la propriété **Répéter le texte d'accueil**, dans le groupe **Message vocal**, dans Editor.
VOIP_AUDIO=	Message vocal comme fichiers audio via Voice over IP : ▸ *0* : inactif ▸ *1* : **Voix par IP (Fichier audio)** Par défaut : *0* Correspond à la propriété **Voix par IP (Fichier audio)**, dans le groupe **Message vocal**, dans Editor. **Remarque :** Les conditions suivantes doivent être satisfaites pour que le service VoIP soit utilisable : ▸ Le fournisseur de service VoIP doit prendre en charge les protocoles **SIP** et **RTP**. ▸ Les ports correspondants doivent être ouverts au niveau du pare-feu : SIP (Par défaut : *5060*) RTP (Par défaut : *4000*) RTCP (Par défaut : *4001)*
VOIP_DOMAIN=	Adresse du serveur du fournisseur VoIP. Par défaut : *vide* Correspond à la propriété **Adresse du serveur**, dans le groupe **Voix par IP**, dans Editor.
VOIP_PASSWORD=	Mot de passe pour l'accès VoIP. Est enregistré sous forme cryptée et ne doit être modifié que dans l'interface utilisateur. Par défaut : *vide*

Entrée	Description
	Correspond à la propriété **Mot de passe**, dans le groupe **Voix par IP**, dans Editor.
VOIP_RTP=	Numéro du port RTP pour VoIP. Par défaut : *4000* Correspond à la propriété **Port RTP**, dans le groupe **Voix par IP**, dans Editor.
VOIP_SIP=	Numéro du port SIP pour VoIP. Par défaut : *5060* Correspond à la propriété **Port SIP**, dans le groupe **Voix par IP**, dans Editor.
VOIP_TTS=	Message vocal comme text-to-speech via voix par IP : ▸ *0* : inactif ▸ 1: **Voix par IP (Text-to-Speech)** Par défaut : *0* Correspond à la propriété **Voix par IP (Text-to-Speech)**, dans le groupe **Message vocal**, dans Editor. **Remarque :** Les conditions suivantes doivent être satisfaites pour que le service VoIP soit utilisable : ▸ Le fournisseur de service VoIP doit prendre en charge les protocoles **SIP** et **RTP**. ▸ Les ports correspondants doivent être ouverts au niveau du pare-feu : SIP (Par défaut : *5060*) RTP (Par défaut : *4000*) RTCP (Par défaut : *4001)*
VOIP_USER=	Nom d'utilisateur pour l'accès VoIP. Par défaut : *vide* Correspond à la propriété **Identifiant utilisateur**, dans le groupe **Voix par IP**, dans Editor.

Information

Certaines propriétés peuvent accepter les valeurs d'autres propriétés. Vous enregistrez toujours la dernière valeur saisie. La valeur des entrées ini ne doit donc pas toujours correspondre aux valeurs des propriétés affichées dans l'éditeur. Les propriétés suivantes sont affectées :

- **SMTP_SERVER=** -> **Adresse du serveur**
- **SMTP_USER=** -> **Identifiant utilisateur**
- **SMTP_PASSWORD=** -> **Mot de passe**
- **SMTP_OUT_ADDR=** -> **Adresse pour les e-mails sortants**

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 (à la page 15)for sending media are configured in the properties **Module Message Control** of the working area.

ACTIVATING MESSAGE CONTROL FOR THE PROJECT

In the Runtime, the Message Control module can be activated as follows:

- **Actif lors du démarrage du runtime** property: Once this property is activated Message Control will automatically start in Runtime.
- Function **Send Message: activate** (à la page 87): 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** (à la page 87) in the Runtime.

PROJECT-SPECIFIC PROPERTIES

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

- Dynamic part of the subject via the property **Partie dynamique du sujet**: Incoming messages are filtered for this content for forwarding to Message Control.
- Logging the message via property **Enregistrer dans liste d'événements**: In the Chronologic Event List messages can be logged according to different criteria:
- *Confirmer tous :*
  Tous les messages

- *Pas de confirmation :*
  Aucun message
- *Confirmations négatives uniquement :*
  Uniquement les messages refusés ou n'ayant pas reçu de réponse (répétitions incluses)
  - *Confirmations positives uniquement :*
    Uniquement les messages ayant déjà été acquittés
- 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 **Nombre maximum d'essais** property.

CONFIGURATION OF THE MESSAGE

The message comprises:

1. Static part: unalterable subject which is defined in the global settings (à la page 15) for e-mails.
2. Dynamic part: project-specific part of the subject. Is defined with the property **Partie dynamique du sujet**.

STRUCTURE OF THE DYNAMIC PART OF THE SUBJECT

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

- A parameter (à la page 88) 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** (à la page 89) 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: **@MyText**	A check is made to see if an entry in @MyText is present in the language table for the current language.	- *Existing*: entry is added to the message. - *non-existent*: According to the settings of the project properties **Montrer les mots-clé non-traduits** either **@MyText** or **MyText** is added to the message.
Variable: **%Var1%**	A check is made if the variable exists and the value can be read.	- *Existing*: Value of the variable is taken and added to the text as string. - *Not existing/not readable*: The text **xxx** is added to the message.
Compound entry: **@StringTable+%var1 Text**	A check is made if: - 1. the variable exists and the value can be read. Value is attached to the prefix text ("MultipleText") as string. - 2. an entry in the language table exists for the text	- Variable and text exist: text is added to the message. Example: Value of the variable is *33*. The language table is then checked for **@StringTable33** and the corresponding text is added. - *non-existent*: According to the settings of the project properties **Montrer les mots-clé non-traduits** either **@MyText** or **MyText** is added to the message.

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.

DÉVELOPPEMENT

Deux procédures sont disponibles pour créer un écran :

- L'utilisation de la boîte de dialogue de création de synoptique
- par l'intermédiaire des propriétés de création de synoptique

Étapes de création du synoptique à l'aide des propriétés si la boîte de dialogue de création de synoptique a été désactivée dans la barre de menus dans **Outils**, **Paramètres** et **Utiliser l'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 **Nom** property.

   b) Select *Message Control* in the **Type de synoptique** property.

   c) Select the desired frame in the **Gabarit** property.

3. Configurez le contenu du synoptique :

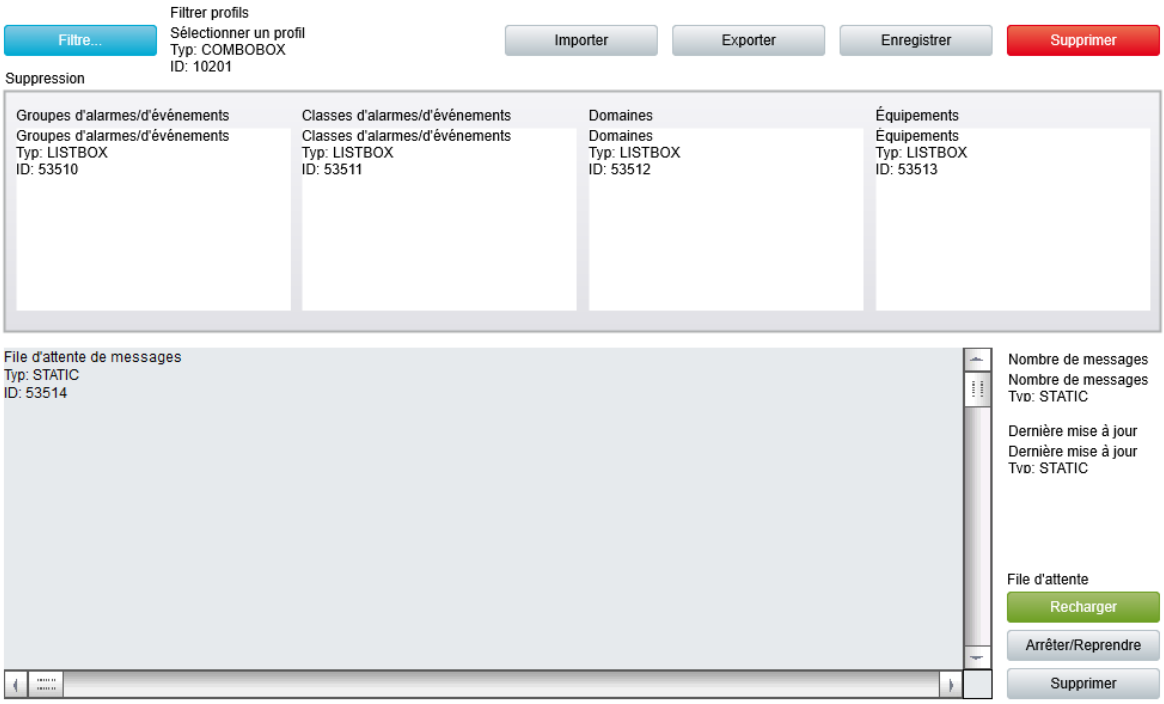
   a) Sélectionnez l'option de menu **Éléments (type de synoptique)** dans la barre de menus.

   b) Sélectionnez *Insérer un modèle* dans la liste déroulante.
   La boîte de dialogue de sélection de mises en forme prédéfinies s'affiche à l'écran. Certains éléments de contrôle sont insérés dans le synoptique à des positions prédéfinies.

   c) Supprimez les éléments superflus du synoptique.

   d) Si nécessaire, sélectionnez des éléments supplémentaires dans la liste déroulante **Éléments**. Placez-les aux emplacements souhaités sur le synoptique.

4. Create a screen switch function.



Si vous souhaitez modifier la liste directement à l'aide du moniteur, activez la fonctionnalité MultiTouch. Vous trouverez des informations détaillées à ce sujet dans le chapitre Configuration des interactions.

Longer texts can also be displayed in the Runtime over several lines using the **Retour à la ligne automatique** property.

In the Editor, go to **Affichage** in the properties of the respective list properties and activate the checkbox of the **Retour à la ligne automatique** property.

The line height must be amended manually.

INSÉRER UN MODÈLE

Éléments de contrôle	Description
Insérer un modèle	Opens the dialog for selecting a template for the screen type. Templates are shipped together with zenon and can also be created by the user. Templates add pre-defined control elements to pre-defined 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

Éléments de contrôle	Description
	and placed in the zenon screen. Elements can be moved on the screen and arranged individually.

FENÊTRE

Listes et champs pouvant être affichés dans le Runtime.

Éléments de contrôle	Description
File d'attente des messages	Liste de messages. Contient les messages envoyés, confirmés ou supprimés ou en attente d'envoi. Les messages sont uniquement affichés, et ne peuvent plus être modifiés. Les en-têtes de colonnes peuvent être définis individuellement (à la page 57) et peuvent être localisés lors d'un changement de langue s'ils sont précédés du signe **@**.
Nombre de messages	Nombre de messages actuellement existants.
Dernière mise à jour	Heure de la dernière mise à jour.
Groupes d'alarmes/d'événements supprimés	**Groupes d'alarmes/d'événements** pour lesquels des messages ont été supprimés, et dont les alarmes ne sont donc pas affichées dans la file d'attente de messages.
Classes d'alarmes/d'événements supprimées	**Classes d'alarmes/d'événements** pour lesquelles des messages ont été supprimés, et dont les alarmes ne sont donc pas affichées dans la file d'attente de messages.
Domaines d'alarmes/d'événements supprimés	**Domaines d'alarmes/d'événements** pour lesquels des messages ont été supprimés, et dont les alarmes ne sont donc pas affichées dans la file d'attente de messages.
Équipements supprimés	**Équipements** pour lesquels des messages ont été supprimés, et dont les alarmes ne sont donc pas affichées dans la file d'attente de messages.

FONCTIONS DE LA LISTE

Éléments de contrôle des listes.

Éléments de contrôle	Description
Filtre	Ouvre une boîte de dialogue (à la page 55) de

Éléments de contrôle	Description
	configuration des filtres pour la vue.
Supprimer	Supprime l'entrée sélectionnée de la liste. Il est uniquement possible de supprimer les messages qui n'ont pas encore été envoyés.
Actualiser	Actualise l'affichage. Les données actuelles sont chargées sur le synoptique de la file d'attente de messages. L'heure de la dernière mise à jour peut être affichée avec le champ **Last update** (Dernière mise à jour).
Arrêter/Reprendre	Démarre et arrête la mise à jour cyclique de la liste.

PROFILS DE FILTRE

Gestion de profils

Éléments de contrôle	Description
Sélection de profil	Ouvre la boîte de dialogue permettant de sélectionner un profil.
Enregistrer	Enregistrer la configuration actuelle sous forme de profil.
Supprimer	Supprime le profil.
Importer	Ouvre la boîte de dialogue d'importation de profils depuis un fichier.
Exporter	Ouvre la boîte de dialogue d'exportation de profils depuis un fichier.

Information

The columns of the list can be configured via the Filter (à la page 55) 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 **Utilisateur Message Control** or respectively **Groupe Message Control** must be active in order to use an user in Message Control.

Users and user groups are referenced by names.

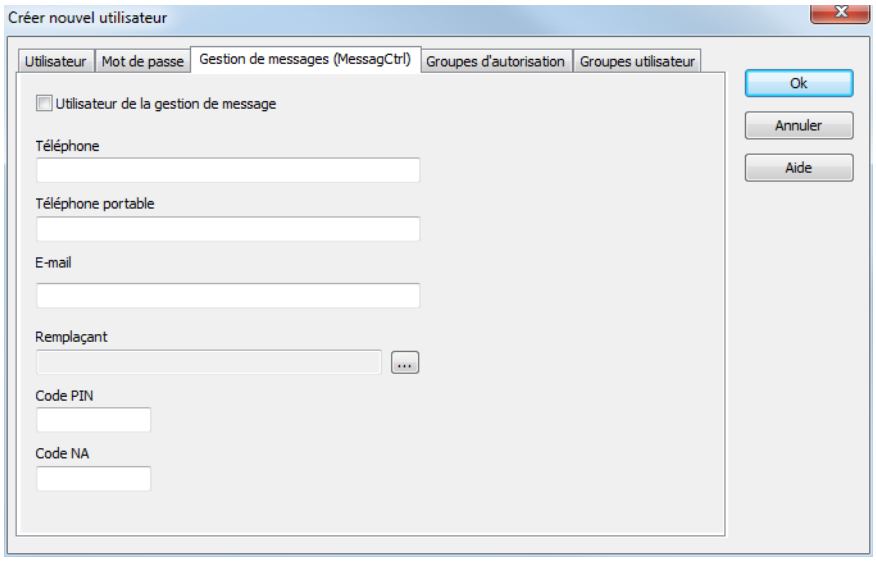
Attention

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
Message Control User	*Active*: The user is used by the module Message Control.
Telephone	Number of the voice-compatible telephone device of the user. Used for text to speech.

Parameter	Description
	Enter numbers. In addition, the following are permitted: ▸ The prefix **+** as an abbreviation for *00* of the international area code is permitted. ▸ The following separators are also permitted in AD user administration: Minus (**-**), slash (**/**) and **space** 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.
Cell phone	Cellphone number of the user. Used for messages via GSM and SMS (text messages). Enter numbers. In addition, the following are permitted: ▸ The prefix **+** as an abbreviation for *00* of the international area code is permitted. ▸ The following separators are also permitted in AD user administration: Minus (**-**), slash (**/**) and **space** 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.
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 **Message Control** users are offered for selection.
PIN code	PIN code with which the user confirms the receipt of the message. Le code est constitué d'un nombre à quatre chiffres compris entre *0000* et *9999*.
NA code	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. If there is no other user entered in the list, the message is entered as "*not successfully acknowledged*". The function assigned to this is executed. In addition, a "*rejected by*" CEL entry is created in each case. Le code est constitué d'un nombre à quatre chiffres compris entre *0000* et *9999*.

Parameter	Description
	Note: You can find further information on the assignment of functions in the Confirmation of receipt - confirmation of receipt settings (à la page 84) chapter.

FERMER LA BOÎTE DE DIALOGUE

Option	Description
OK	**Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.**
Annuler	Annule toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Aide	Opens online help.

Attention

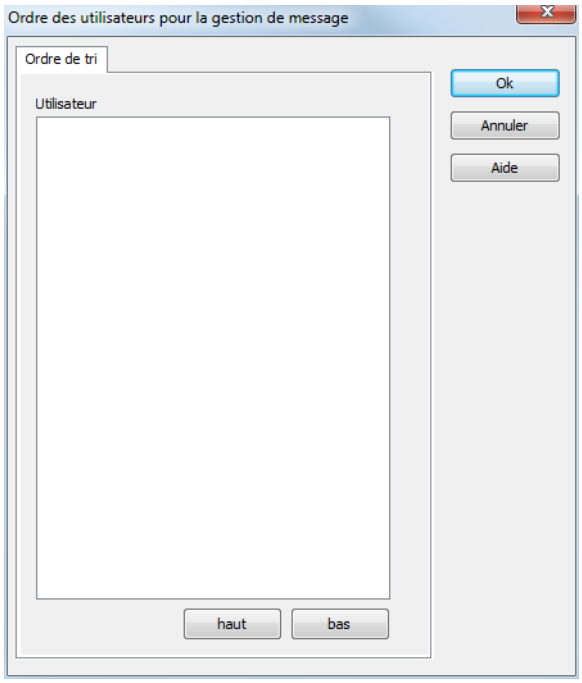
The acknowledgment codes for PIN (confirmation) and NA (rejection) must differ and should not be too similar.

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 **Ordre des utilisateurs** within a user group.



Parameter	Description
User	List of all available users.
Up	Moves selected user up one place.
Down	Moves selected user down one place.
OK	Applique les paramètres et ferme la boîte de dialogue.
Cancel	Annule toutes les modifications et ferme la boîte de dialogue.
Help	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 (à la page 87) and deactivated (à la page 87)
- messages are sent (à la page 65)
- The Message Control screen will be displayed (à la page 55) in order to manage the message queue
- the message queue is saved (à la page 60)
- elements for messages are disabled (à la page 60)

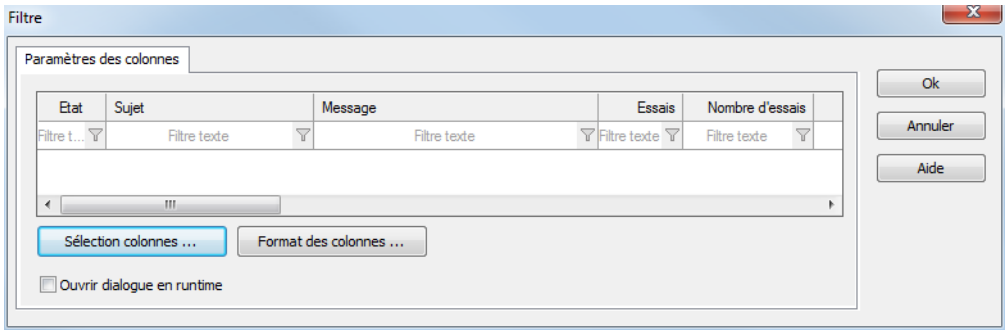
7.5.1 Screen switching to a screen of type Message Control

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 **Appel de synoptique**
3. the selection dialog for the screen is opened
4. select the desired screen of type *Message Control* (à la page 46)
5. the dialog for configuring the column settings (à la page 55) 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

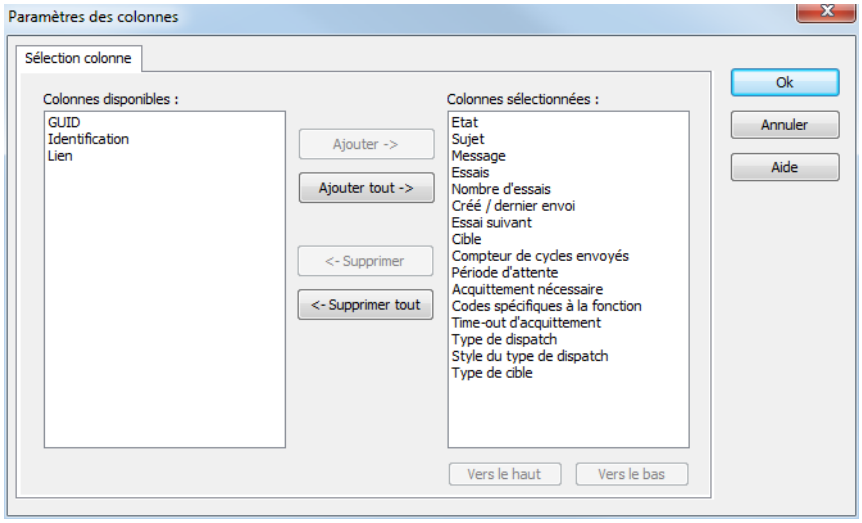
Settings for filtering messages



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	Fonction
Colonnes disponibles	Liste de colonnes pouvant être affichées dans la table.
Colonnes sélectionnées	Colonnes affichées dans la table.
Ajouter ->	Déplace la colonne sélectionnée des colonnes disponibles vers les éléments sélectionnés. Lorsque vous confirmez la boîte de dialogue en cliquant sur OK, ces colonnes sont affichées dans la vue de détail.
Ajouter toutes ->	Déplace toutes les colonnes disponibles vers les colonnes sélectionnées.

Option	Fonction
<- Supprimer	Supprime les colonnes marquées des éléments sélectionnés et les affiche dans la liste des colonnes disponibles. Lorsque vous confirmez la boîte de dialogue en cliquant sur OK, ces colonnes sont supprimées de la vue de détail.
<- Supprimer tout	Toutes les colonnes sont supprimées de la liste des colonnes sélectionnées.
Vers le haut	Déplace l'entrée sélectionnée vers le haut. Cette fonction est uniquement disponible pour les entrées uniques ; les sélections multiples ne sont pas autorisées dans ce cas.
Vers le bas	Déplace l'entrée sélectionnée vers le bas. Cette fonction est uniquement disponible pour les entrées uniques ; les sélections multiples ne sont pas autorisées dans ce cas.

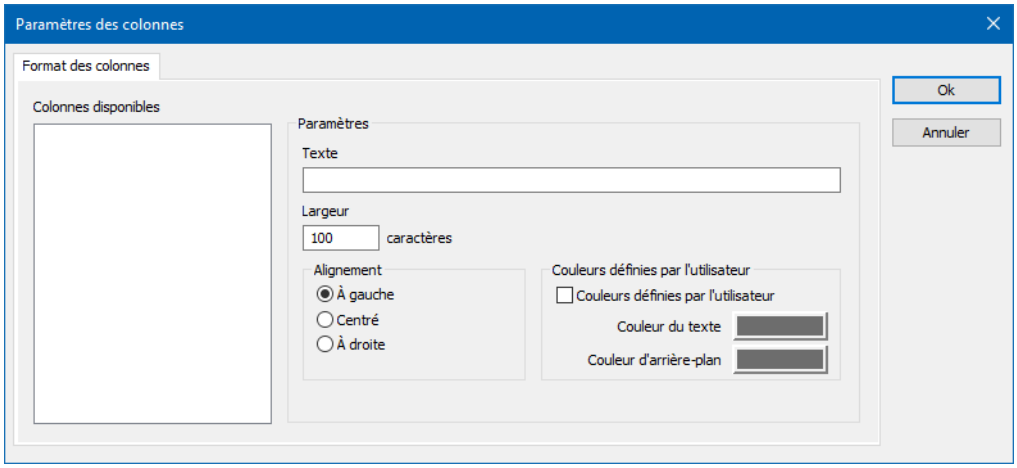
CLOSE DIALOG

Options	Description
OK	**Applique les paramètres et ferme la boîte de dialogue.**
Annuler	Annule toutes les modifications et ferme la boîte de dialogue.
Aide	Opens online help.

7.5.1.1.2 Column format

Formatting of columns.

Configuration des propriétés des colonnes pour les listes configurables. Les paramètres ont un effet sur la liste correspondante dans Editor ou, lors de la configuration d'un appel de synoptique, dans le Runtime.



COLONNES DISPONIBLES

Option	Description
Colonnes disponibles	Liste de colonnes disponibles avec la fonction de **sélection de colonnes**. La configuration de la colonne sélectionnée se déroule via les options de la section **Paramètres**.

PARAMÈTRES

Option	Description
Paramètres	Paramètres de la colonne sélectionnée.
Intitulé	Nom de l'intitulé de colonne. Cet intitulé de colonne est compatible avec la fonction de changement de langue en ligne. Pour cela, le caractère **@** doit être saisi devant le nom.
Largeur	Largeur de la colonne en caractères. Calcul : nombre de caractères multiplié par la largeur moyenne des caractères de la police sélectionnée.
Alignement	Alignement. La sélection de l'attribution s'effectue au moyen des cases d'option : Paramètres possibles : ▸ **Gauche** : Le texte est justifié contre le bord gauche de la colonne.

Option	Description
	▸ **Centré** : Le texte est centré dans la colonne. ▸ **Droite** : Le texte est justifié contre le bord droit de la colonne.
Couleurs définies par l'utilisateur	Propriétés permettant de sélectionner des couleurs définies par l'utilisateur pour le texte et l'arrière-plan. Les paramètres ont une incidence dans Editor et dans le Runtime. **Remarque :** ▸ Ces paramètres sont uniquement disponibles pour les listes configurables. ▸ En outre, le focus correspondant dans la liste peut être indiqué par différentes couleurs de texte et d'arrière-plan dans le Runtime. Celles-ci sont configurées dans les propriétés du projet.
Couleurs définies par l'utilisateur	*Active* : Les couleurs définies par l'utilisateur sont appliquées.
Couleur du texte	Couleur d'affichage du texte. Cliquez sur la couleur pour la palette de sélection de couleurs.
Couleur d'arrière-plan	Couleur d'affichage de l'arrière-plan de la cellule. Cliquez sur la couleur pour la palette de sélection de couleurs.
Désactiver le filtre de colonnes dans le Runtime	▸ *Active* : Le filtre de cette colonne ne peut pas être modifié dans le Runtime. **Remarque :** Uniquement disponible pour : ▸ Contrôle de batch ▸ Extended Trend ▸ Synoptiques de filtre ▸ Module Message Control ▸ Gestionnaire de groupe de recettes ▸ Gestion d'équipe ▸ Liste contextuelle

FERMER LA BOÎTE DE DIALOGUE

Option	Description
OK	**Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.**
Annuler	Annule toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Aide	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 **Enregistrer queue courante** function in the **Module 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 **Valeurs limite** is linked to a function **Envoi d'un 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.

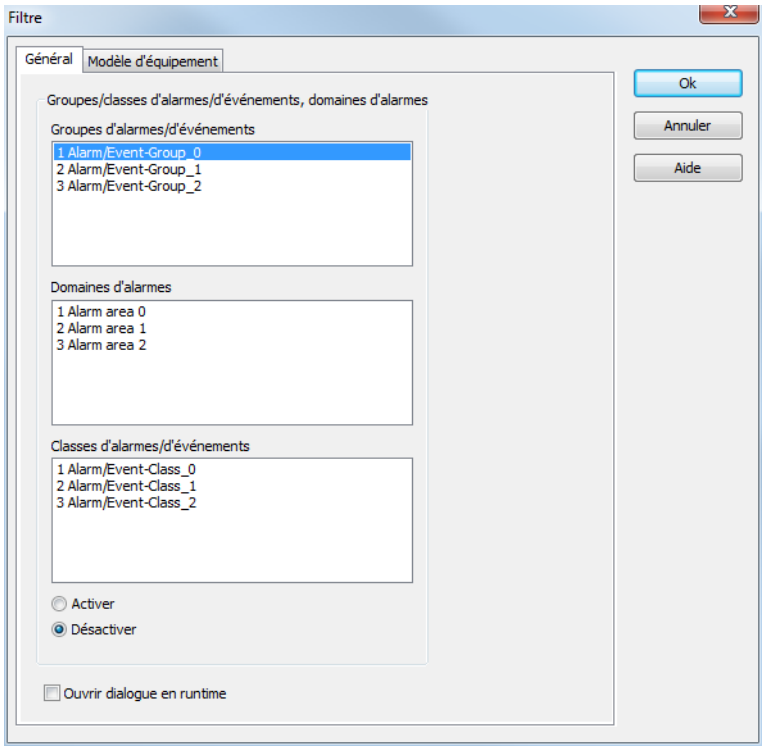
Suppressed entries are logged in the CEL if the property **Enregistrer dans liste d'événements** is configured with *all confirmations* or *only negative confirmations*.

ENGINEERING OF A FUNCTION

To configure the function:

1. Create a new function
2. in group **Module Message Control** select the function **Groupe/classe/domaine/équipement supprimé**
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
Alarm/event groups	List of configured alarm/event groups.
Alarm Areas	List of configured alarm areas.
Alarm/event classes	List of configured alarm/event classes.
Activate	*Active*: When activating the function the selected elements are added to the list of suppressed elements.
Deactivate	*Active*: When activating the function the selected elements are

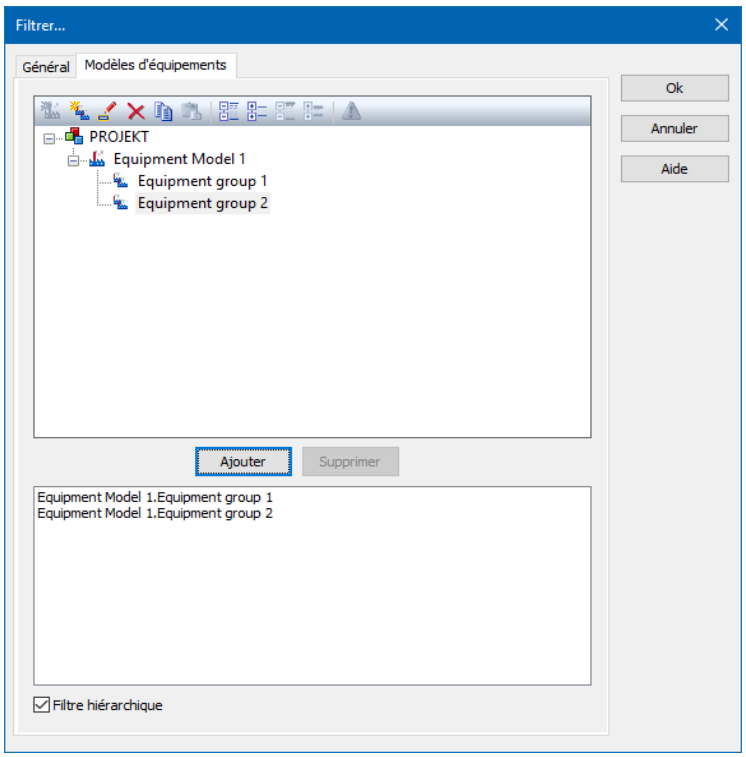
Parameter	Description
	removed from the list of suppressed elements.
Show dialog in the Runtime	Checkbox to select whether this dialog is shown in Runtime: ▸ *active*: This dialog is called up during operation in the Runtime on the current computer. In the network, this dialog is called up on the computer that executes the function. As a result, changes to existing parameter settings of an zenon Editor configuration are possible during execution in zenon Runtime. ▸ *Inactive*: This dialog is not shown in the Runtime during operation. The function or the command is immediately executed with the project configuration created in the Editor.

FERMER LA BOÎTE DE DIALOGUE

Option	Description
OK	**Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.**
Annuler	Annule toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Aide	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
Barre d'outils	Symboles pour: ▸ Modifier les modèles d'équipement locaux ▸ Agrandir ou réduire l'affichage ▸ Afficher des informations
Liste de modèles d'équipements	Autorise la sélection de modèles et de groupes. La liste distingue les modèles d'équipement issus du projet global et des projets locaux. Les modèles d'équipement locaux peuvent être créés, modifiés ou supprimés. **Remarque :** Les modèles d'équipement du projet global ne peuvent pas être affichées si des modèles portant le même nom que dans le projet local. Les modèles affectés sont affichés en cliquant sur le symbole d'avertissement (triangle avec point d'exclamation). Pour plus de détails, reportez-vous au manuel **Modèles d'équipement**, chapitre **Modification de modèles d'équipement locaux**.
Ajouter	Ajoute les groupes sélectionnés à la liste de filtres.

Option	Description
Supprimer	Supprime tous les groupes sélectionnés de la liste de filtres.
Liste de filtres	Affiche tous les groupes d'équipements devant être filtrés.
Filtre hiérarchique	Case à cocher pour l'activation du filtre hiérarchique de modèle d'équipement. ▸ *Active* : les variables liées à une sous-hiérarchie du groupe d'équipement sélectionné sont prises en compte dans le filtre et sont incluses dans l'affichage du Runtime. ▸ *Inactive* : lors du filtre, seules les variables liées au groupe d'équipement sélectionné sont prises en compte. Par défaut : *active*

FERMER

Option	Description
OK	Applique les paramètres et ferme la boîte de dialogue.
Annuler	Annule la sélection et ferme la boîte de dialogue. **Attention :** Toute modification apportée à la structure des équipements locaux est conservée.
Aide	Opens online help.

CONFIGURATION

De nouveaux modèles et groupes peuvent être créés pour le projet actif et les modèles et groupes existants peuvent être supprimés.

AJOUTER UN MODÈLE AU PROJET

Pour ajouter un nouveau modèle :

1. Cliquez sur le projet.
2. Dans la barre d'outils, sélectionnez **Nouveau modèle d'équipement**

SUPPRIMER UN MODÈLE

Pour supprimer un modèle existant :

1. Cliquez sur le modèle.
2. Sélectionnez **Supprimer** dans la barre d'outils.

AJOUTER DES GROUPES AU MODÈLE

Pour ajouter un groupe au modèle :

1. Sélectionnez le modèle d'équipement de votre choix.
   Attention : Si des conflits de nom existent entre les modèles d'équipement globaux et locaux, les modèles d'équipement locaux sont affichés et les modèles locaux sont ignorés. Vous pouvez obtenir des informations concernant d'éventuels conflits en cliquant sur le symbole correspondant (un triangle avec un point d'exclamation) dans la barre d'outils.
2. Sélectionnez un groupe ou un niveau d'équipements.
3. Ajoutez le nouveau groupe à la liste affichée dans la partie inférieure de la boîte de dialogue en cliquant sur le bouton **Ajouter**.

   Remarque :

   - Les sous-groupes ne sont pas ajoutés automatiquement.
   - Vous pouvez sélectionner autant de groupes que vous le souhaitez.

SUPPRIMER UN GROUPE DU MODÈLE

Pour supprimer un groupe d'un modèle :

1. Sélectionnez les éléments souhaités dans la liste affichée dans la partie inférieure de la boîte de dialogue (les sélections multiples sont autorisées).
2. Cliquez sur le bouton **Supprimer**.

Remarque : les modifications apportées à un élément dans l'arborescence sont conservées, même si vous cliquez sur le bouton **Annuler**. **Annuler** signifie uniquement qu'aucun élément n'a été sélectionné.

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

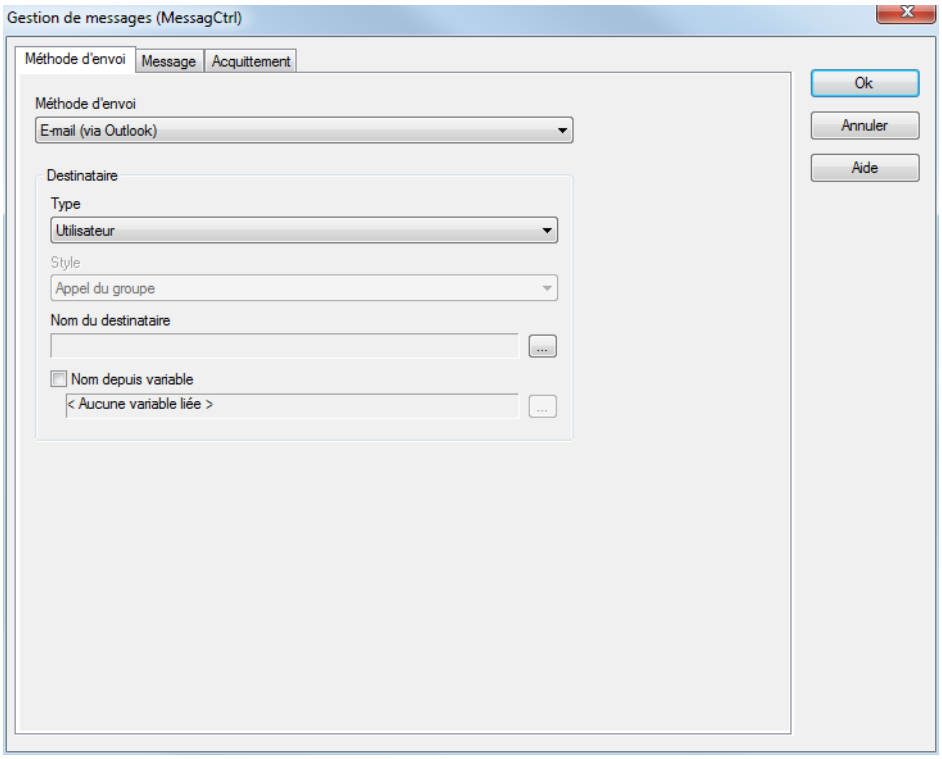
This function allows for a sending of messages in the Runtime. To do so, link this function with:

- an alarm (function for group, class or area)
- a limit value (property **Valeurs limite**/**Exécuter instantanément la fonction**),
- 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 **Envoi d'un message** function in the **Module Message Control** group.
3. The dialog for configuration is opened:



4. Configure
   - Sending mode (à la page 67)
   - Message (à la page 72)

- Acknowledgment of receipt (à la page 84)

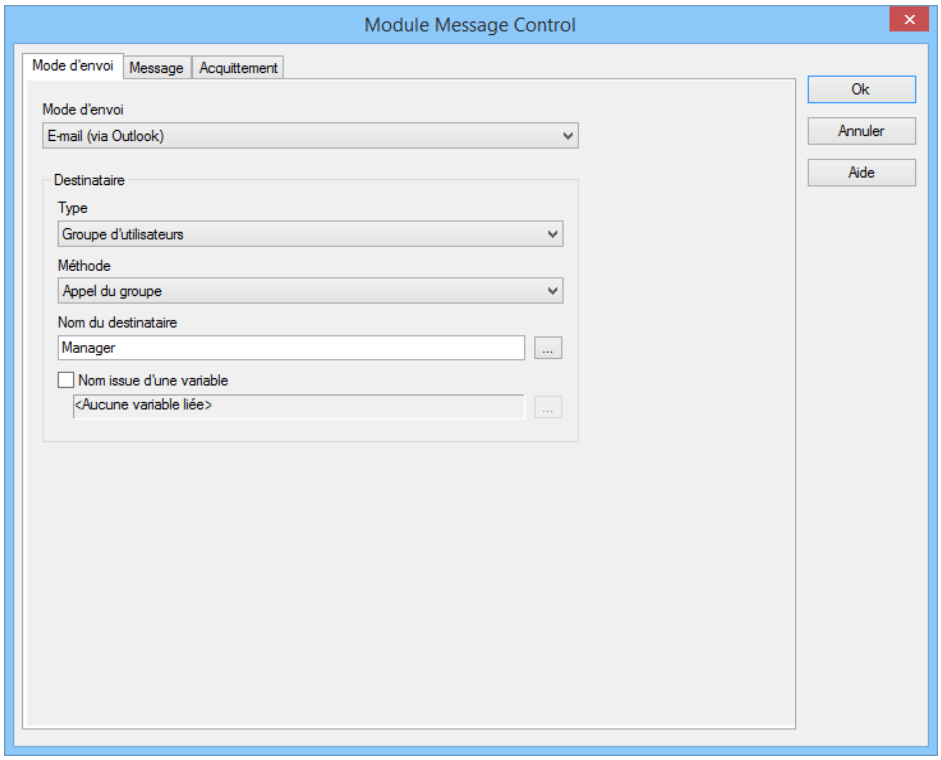
5. Close the dialog by clicking on **OK**.
   Note: The entries are validated in the tabs by clicking on **OK**. If incorrect configurations are discovered, the user is informed by means of a message box. The user can then change the configuration or continue with the saving of the incorrect configuration.
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 **Envoi d'un 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 **Envoi d'un 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
Sending mode	Selection of the sending mode from the drop-down list: The following are available depending on configuration: ▸ E-mail message via Outlook ▸ E-mail message via SMTP ▸ Voice message audiofile via modem ▸ Voice message Text-to-Speech via modem ▸ SMS message via SMS server ▸ SMS message via GSM modem ▸ Voice over IP as audio file ▸ Voice over IP configured as Text-to-Speech **Note:** In the drop-down list, only media whose **Versandart aktiv** property has been activated in the Message Control Properties (à la page 15) of the workspace are shown.
Recipient	Recipient settings:
Type	Configuration of recipient pattern from the drop-down list. In doing so, the dialog to select a user or a user group is opened. The following are available: ▸ *User group*: Sending to a group. The **Method** option stipulates how group members are notified. ▸ *User*: Dispatch to an individual user. ▸ *User group with shift filter*: Selection of a user group for the shift management. Users linked to a shift can be notified. **Recommendation:** Select *infinite successive call* as a **method**. Configured users or user groups can be amended using the **Recipient name** option.
Method	Only available if, for the **Type** *User group* or *User group with shift filter* option has been selected. Selection of sending method from drop-down list. Possible methods:

Parameter	Description
	▸ *Group call*: All users of the group will receive the message parallel. (Notice: In versions prior to zenon 7.00, this method corresponded to the **group sending** dispatch type.) ▸ *Following call*: The message is delivered to the users according to the defined sequence (à la page 51) 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. ▸ *Endless following call*: In principle corresponds to the *Following call*, however, includes a repetition: After the last recipient the group is being run through again beginning on the top. The message is being sent until it is positively acknowledged by the first recipient. **Note:** This setting is independent of the **Nombre maximum d'essais** property.
Recipient name	Only available if the option **Name from variable** is deactivated. Opens dialog for selection of a recipient or a recipient group - depending on the selection in the option **Type**.
Name from variable	*Active*: The variable is read in Runtime and a user or user group from the user administration is read off by means of its value. Click on button **...** in order to open the dialog for selecting a variable.
OK	Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue. Dialog can only be closed if all required options have been configured.
Cancel	Annule toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Help	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 **Utilisateur Message Control** property has been activated
   - The **Groupe Message Control** 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 **Ordre des utilisateurs** property for this. The users are notified in the stipulated order.
4. For **Message Control**, configure the **recipient** in the **Envoi d'un message** function:
   a) Open the function and switch to the **Dispatch Type** (à la page 65) 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 **Envoi d'un 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
Group call:	▸ 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. ▸ If a user does not confirm the message, a message is sent to their substitute if they are linked in a shift.
Following call:	▸ An attempt is made to find the first user of the group who is linked to the shift. ▸ This user is notified. ▸ If they cannot be contacted, the next user is searched for and notified. ▸ This is run through once until the end of the group has been reached. ▸ 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.
Endless following call:	▸ An attempt is made to find the first user of the group who is linked to the shift. ▸ This user is notified. ▸ If they cannot be contacted, the next user is searched for and notified. ▸ This is run through cyclically until a user is reached. ▸ 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.

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 **Envoi d'un 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 (à la page 67).

Messages are individually configured:

- E-mail via Outlook and SMS via GSM or SMS gateway (à la page 73)
- E-mail message via SMTP (à la page 76)

- Voice message by audio file (à la page 80)
- Voice message via text to speech (à la page 82)

You can find details on configuration of the parameters for messages in the **Parameters for messages** (à la page 88) chapter.

Information

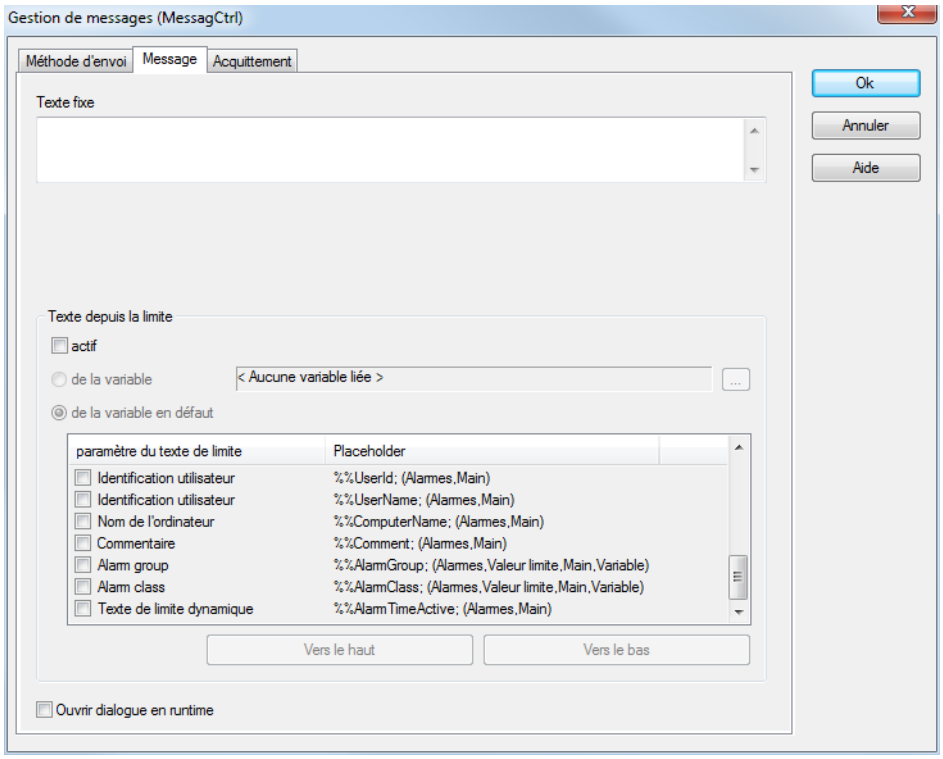
If parameters or values of additional variables are included in a message, these variables are not automatically logged on at the driver. If a variable is not logged on at the point of time the function is being carried out it must be logged on during the activity. Under certain circumstances (e.g. in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

7.5.4.2.1E-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
Free message text	Integral part of the message. This text is searched for placeholders: - **$**: indicates "formatted" text. The text is subsequently parsed for parameters. - Text components are considered parameters if they are separated from the rest of the text with a semicolon (**;**) and end with a semicolon (**;**). - A Parameter (à la page 88) can be: - key of the language table - name of a variable - compound entry in the language table Parameters can be copied from the List (à la page 89) of **limit value text parameters**. You can read more about the subject in the **Parameters for meesages** (à la page 88) chapter.
Text from limit value	Properties for additional parameters from limit values.
active	*Active*: Adds additional parameters from the limit value to the constant text. Configuration via options:

Parameter	Description
	▸ **from variable** ▸ **from penultimate variable** ▸ **list limit value text parameters** **Attention:** If this option is active only limit value violations can trigger this function.
from variable	Parameters are taken over from the defined variable. Click on button **...** in order to open the dialog for selecting a variable.
from penultimate variable	Parameters are taken over from the variable whose limit value was violated.
list limit value text parameters	Selecting parameters from list by activating the corresponding checkbox. Sequence is definded by drag & drop with the mouse or the buttons **move entry up** and **move entry down**. **Split placeholders:** Displays placeholders which can be used in free message text. If the option **from variable** or **from penultimate variable** or the variable selection is changed the example entry will be adjusted. Placeholders can be copied. Structure: **%Variable%Parameter; (origin)** ▸ **%**: percentage sign as prefix for variables and parameters ▸ **Variable**: Variable. If no variable is indicated the following parameter refers to the main variable. ▸ **Parameter**: Parameter ▸ ;: semicolon ends string. ▸ Origin: Only information, not used for free text. Indicates which variables can be used for the parameter : main variable via limit value main variable via AML additional variable via limit value or AML For details, see the **Parameters for messages** (à la page 88) and Parameter limit value text (à la page 89) chapters. If invalid parameters are used a corresponding error message is displayed in the message: **<Invalid parameter: 'Parameter'>.**

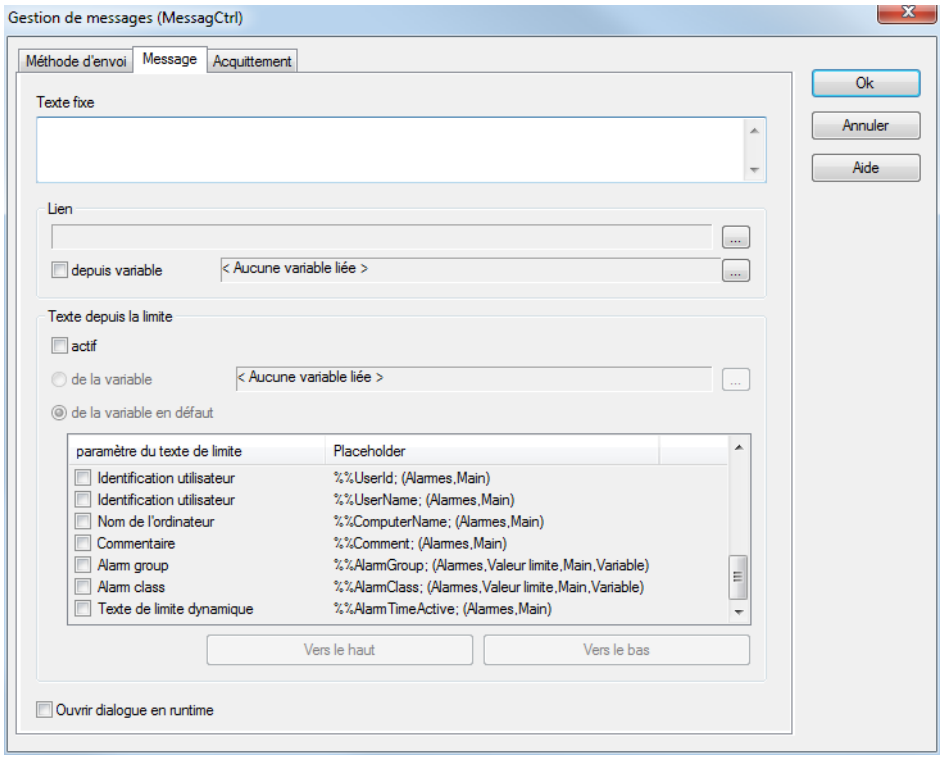
Parameter	Description
Move selected entry up	Places selected element one step higher.
Move selected entry down	Places selected element one step lower.
Show this dialog in the Runtime	*Active*: Opens this dialog in the Runtime before executing the function. **Attention - Behavior in the Runtime:** ▸ In 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. ▸ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out. ▸ If the dialog is edited and closed with OK the changes will be saved and the function carried out. ▸ If the dialog is closed with **cancel** no message is being sent. This process is documented in the CEL provided for the property **Enregistrer dans liste d'événements** *all confirmations* or *only negative confirmations* was selected.
OK	Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue. Dialog can only be closed if all required options have been configured.
Cancel	Annule toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
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
Free message text	Integral part of the message. This text is searched for placeholders: - **$**: indicates "formatted" text. The text is subsequently parsed for parameters. - Text components are considered parameters if they are separated from the rest of the text with a semicolon (**;**) and end with a semicolon (**;**). - A Parameter (à la page 88) can be: - key of the language table - name of a variable - compound entry in the language table Parameters can be copied from the List (à la page 89) of **limit value text parameters**. You can read more about the subject in the **Parameters for meesages** (à la page 88) chapter.
Appendix	Configuration of an appendix to the e-mail. Any desired file can be sent as an attachment. Clicking on the **...** button opens the dialog to select a file with the presetting to the **Files/graphics** node.

Parameter	Description
	If the attachment is in another folder: ▸ Activate the **Direct file selection** checkbox. ▸ Click on the **...** button for the file selection. ▸ Select the desired folder and the desired file. When sending the message the selected file is added as attachment to the e-mail. **Note:** To send files from other folders, select the **From variable** option.
from variable	Sending of attachments using a variable. Any desired file can thus be sent from any desired path: ▸ *Active*: Select a string variable. Click on button **...** in order to open the dialog for selecting a variable. The value of the string variables is read in the Runtime and the content is analyzed. The path to the appendix is set according to the analysis: ▸ Absolute path: The path is used unchanged. ▸ No absolute path: The value of the string variables is attached to the path to the zenon graphics folder. When setting the path, a check is not carried out to see which expansion the file has.
Text from limit value	Properties for additional parameters from limit values.
active	*Active*: Adds additional parameters from the limit value to the constant text. Configuration via options: ▸ **from variable** ▸ **from penultimate variable** ▸ **list limit value text parameters** **Attention:** If this option is active only limit value violations can trigger this function.
from variable	Parameters are taken over from the defined variable. Click on button **...** in order to open the dialog for selecting a variable.
from penultimate variable	Parameters are taken over from the variable whose limit value was

Parameter	Description
	violated.
list limit value text parameters	Selecting parameters from list by activating the corresponding checkbox. Sequence is definded by drag & drop with the mouse or the buttons **move entry up** and **move entry down**. **Split placeholders:** Displays placeholders which can be used in free message text. If the option **from variable** or **from penultimate variable** or the variable selection is changed the example entry will be adjusted. Placeholders can be copied. Structure: **%Variable%Parameter; (origin)** ▸ **%**: percentage sign as prefix for variables and parameters ▸ **Variable**: Variable. If no variable is indicated the following parameter refers to the main variable. ▸ **Parameter**: Parameter ▸ ;: semicolon ends string. ▸ Origin: Only information, not used for free text. Indicates which variables can be used for the parameter : main variable via limit value main variable via AML additional variable via limit value or AML For details, see the **Parameters for messages** (à la page 88) and Parameter limit value text (à la page 89) chapters. If invalid parameters are used a corresponding error message is displayed in the message: **<Invalid parameter: 'Parameter'>.**
Move selected entry up	Places selected element one step higher.
Move selected entry down	Places selected element one step lower.
Show this dialog in the Runtime	Active: Opens this dialog in the Runtime before executing the function. **Attention - Behavior in the Runtime:** ▸ 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.

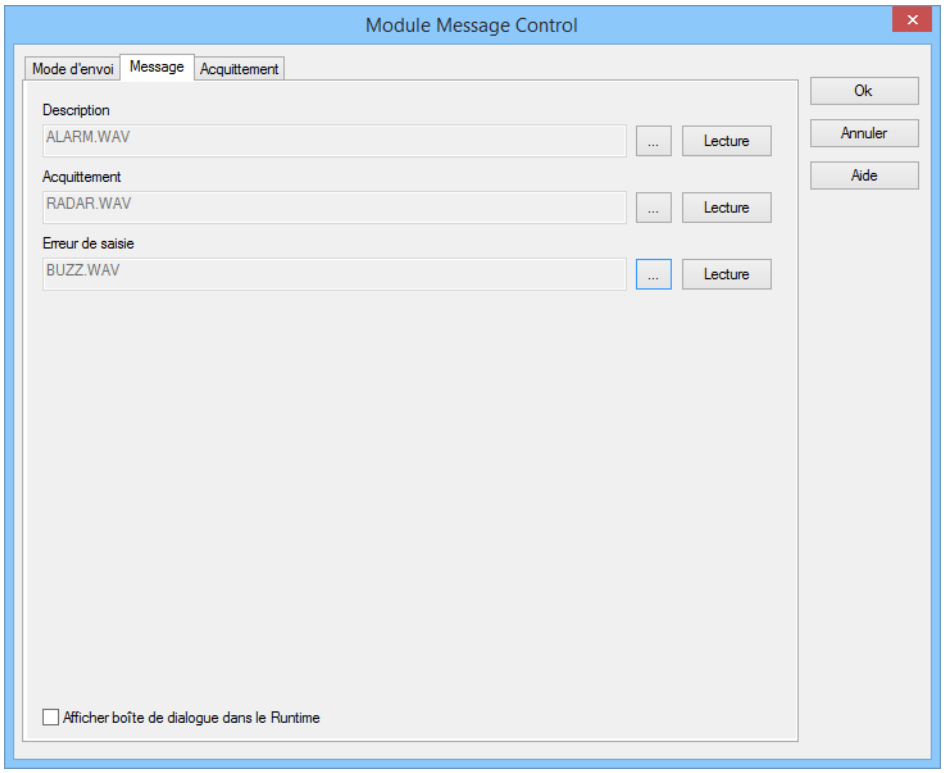
Parameter	Description
	▸ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out. ▸ If the dialog is edited and closed with OK the changes will be saved and the function carried out. ▸ If the dialog is closed with **cancel** no message is being sent. This process is documented in the CEL provided for the property entries into CEL *all confirmations* or *only negative confirmations* was selected.
OK	Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue. Dialog can only be closed if all required options have been configured.
Cancel	Annule toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Help	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
Description	Select the file that contains the description of the message. Click on button **...** in order to open the dialog for selecting a file. This has to be deposited in the node File/Multimedia. The file can be played for testing with the button **Play**.
Acknowledgement	Select the file that contains the text for the message if the **description** is confirmed positively. Click on button **...** in order to open the dialog for selecting a file. This has to be deposited in the node File/Multimedia. The file can be played for testing with the button **Play**.
Misentry	Select the file containing the text in case of failure entries by the recipient. Click on button **...** in order to open the dialog for selecting a file. This has to be deposited in the node File/Multimedia. The file can be played for testing with the button **Play**.
Show this dialog in the	*Active*: Opens this dialog in the Runtime before executing the

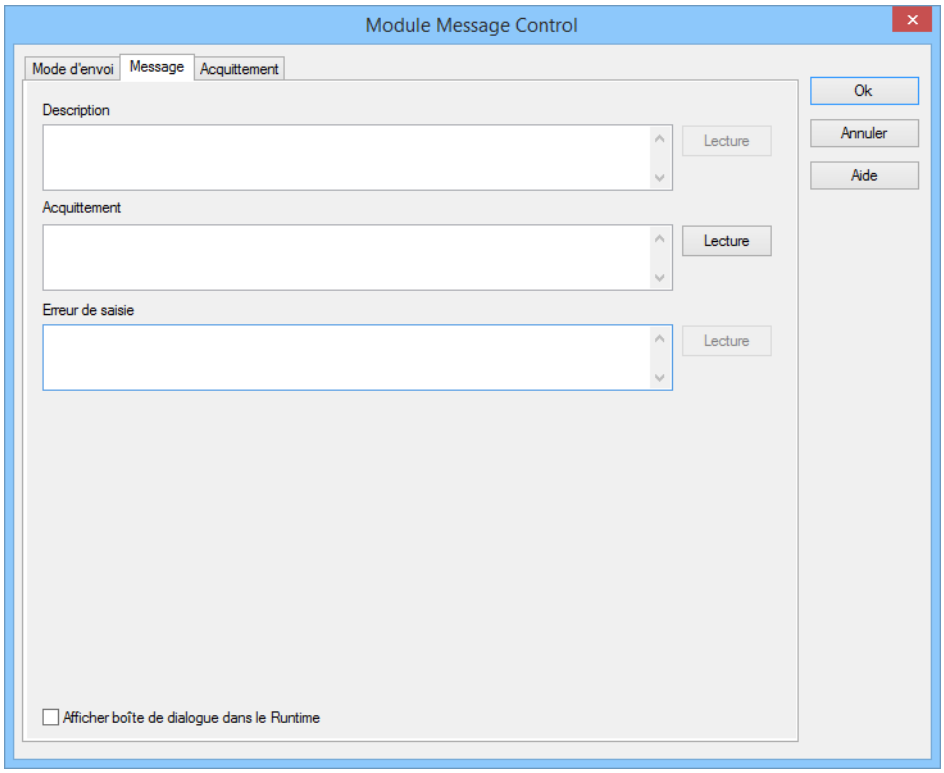
Parameter	Description
Runtime	function. **Attention - Behavior in the Runtime:** ▸ In the Runtime this function is always performed at the computer executing the process. If the function is started on a client the dialog is displayed on the computer executing the process. ▸ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out. ▸ If the dialog is edited and closed with OK the changes will be saved and the function carried out. ▸ If the dialog is closed with **cancel** no message is being sent. This process is documented in the CEL provided for the property entries into CEL *all confirmations* or *only negative confirmations* was selected.
OK	Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue. **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.
Cancel	Annule toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
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
Description	Enter the text to be displayed as message via the Text-to-Speech engine. 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 (à la page 88). The text can be played for testing with the button **Play**.
Acknowledgment	Enter the text to be displayed as a message if, on positive confirmation of the **description**, playback is to be via the Text-to-Speech engine. 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 (à la page 88). The text can be played for testing with the button **Play**.
Misentry	Enter the text that is to be played back via the Text-to-Speech engine in the event of incorrect entries. The text can be entered as free message text. It can also contain dynamic components. These are configured in the same way as

Parameter	Description
	dynamic texts (à la page 88). The text can be played for testing with the button **Play**.
Show this dialog in the Runtime	▸ *Active*: Opens this dialog in the Runtime before executing the function. **Attention: Behavior in the Runtime** Note the following when calling up this dialog in the Runtime: ▸ 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. ▸ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out. ▸ If the dialog is edited and closed with OK the changes will be saved and the function carried out. ▸ If the dialog is closed with **cancel** no message is being sent. This process is documented in the CEL if *all confirmations* or *only negative confirmations* was selected for the **Enregistrer dans liste d'événements** property (**Paramètres spécifiques au projet**group).
OK	Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue. **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.
Cancel	Annule toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Help	Opens online help.

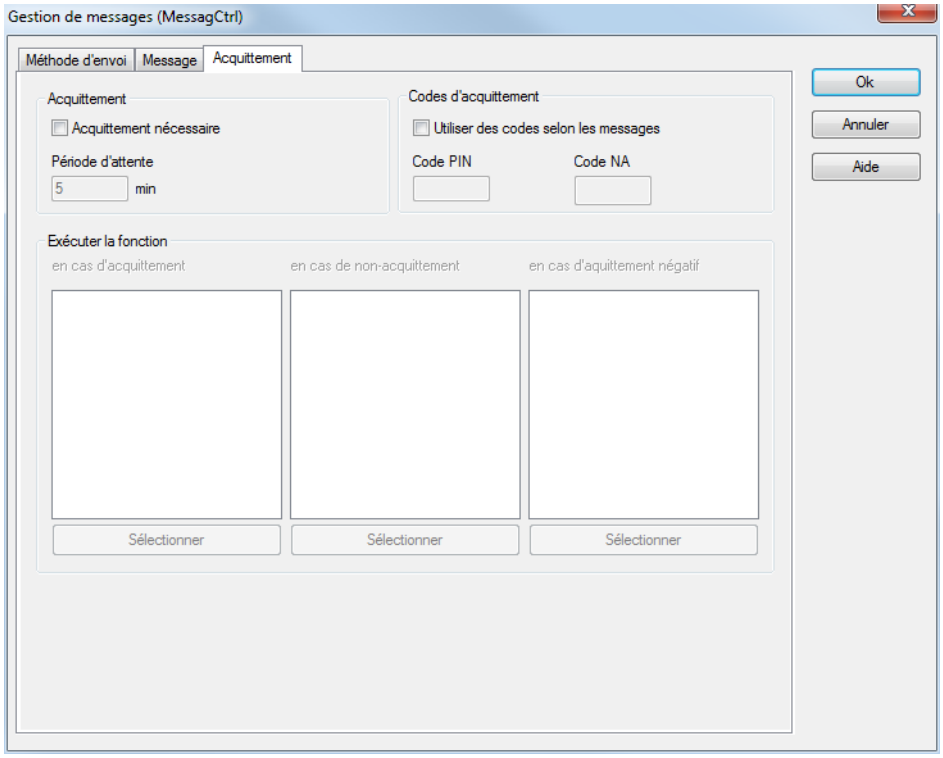
7.5.4.3 Acknowledgment of receipt

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

Attention

Synchronously-transmitted messages (voice message via text-to-speech (à la page 82) and voice message via audio file (à la page 80)) 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
Acknowledgment of receipt	Settings for acknowledgment of receipt.
Acknowledgment necessary	*Active*: The message is only removed from the message queue if, within the **waiting period**, an acknowledgment of receipt has arrived or the **waiting period** has run out. For voice messages this option is always active and cannot be deactivated.
Waiting period	Period of time in minutes during which the receipt has to be acknowledged before the message is forwarded to a substitute person.

Parameter	Description
	Par défaut : **5 minutes**
Confirmation codes	Settings for confirmation codes.
Use message-specific codes	*Inactive*: The codes defined in the properties of the user administration for **Utilisateur Message Control**/**Code PIN** and **Utilisateur Message Control**/**Code NA** are used. *Active*: The defined codes in the options **PIN code** and **NA code** are used for confirmation.
PIN code	Individual code for confirming the message. Le code est constitué d'un nombre à quatre chiffres compris entre *0000* et *9999*.
NA code	Individual code for rejecting the message. Le code est constitué d'un nombre à quatre chiffres compris entre *0000* et *9999*.
Function execution	Selection of functions to be executed after a message has been confirmed, rejected or not acknowledged at all.
In case of confirmation	Functions which are executed if the message is confirmed. A click on the **Select** button opens the function selection dialog.
In case of lack of confirmation	Functions which are executed if there is no response to the message. A click on the **Select** button opens the function selection dialog.
In case of negative confirmation	Functions which are executed if the receipt of message is rejected. A click on the **Select** button opens the function selection dialog.
OK	Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue. Dialog can only be closed if all required options have been configured.
Cancel	Annule toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Help	Opens online help.

CONFIGURATION OF CONFIRMATION OF RECEIPT VIA-EMAIL

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

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

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

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 **Envoi de messages : activer**
3. the function is created

Note: In order to securely log on a project during the Runtime start activate the property **Actif lors du démarrage du runtime** in property group **Paramètres spécifiques au projet** 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 **Envoi de messages : désactiver**
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** (à la page 73)
- **E-mail message via SMTP** (à la page 76)
- **Voice message via text-to-speech** (à la page 82) via modem or VoIP
- **Partie dynamique du sujet** property:

CREATION OF A MESSAGE

Syntax for the structure of a message:

- A parameter (à la page 88) 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** (à la page 89) 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: **@MyText**	A check is made to see if an entry in @MyText is present in the language table for the current language.	▸ *Existing*: entry is added to the message. ▸ *non-existent*: According to the settings of the project properties **Montrer les mots-clé non-traduits** either **@MyText** or **MyText** is added to the message.
Variable: **%Var1%**	A check is made if the variable exists and the value can be read.	▸ *Existing*: Value of the variable is taken and added to the text as string. ▸ *Not existing/not readable*: The text **xxx** is added to the message.
Compound entry: **@StringTable+%var1 Text**	A check is made if: ▸ 1. the variable exists and the value can be read. Value is attached to the prefix text ("MultipleText") as string. ▸ 2. an entry in the language table exists for the text	▸ Variable and text exist: text is added to the message. Example: Value of the variable is *33*. The language table is then checked for **@StringTable33** and the corresponding text is added. ▸ *non-existent*: According to the settings of the project properties **Montrer les mots-clé non-traduits** either **@MyText** or **MyText** is added to the message.

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

7.6.1 Texte de valeur limite et texte libre

Les messages peuvent être configurés sous forme de texte libre. Ils peuvent également contenir des textes issus de valeurs limites et d'éléments de texte dynamiques. Tous les types d'envoi dans **Message Control** ne sont pas compatibles avec les textes issus de valeurs limites. Cependant, tous les textes issus d'éléments dynamiques peuvent être complétés.

CONFIGURATION DU TEXTE DE VALEUR LIMITE

Pour lier des valeurs limite :

1. Dans l'onglet **Message**, activez l'option **Texte de valeur limite actif** (fonction **Envoi d'un message** de Message Control).

Remarque : Cette option n'est pas disponible pour tous types d'envoi. Dans ce cas, configurez l'option **Texte de message libre** avec des éléments de texte dynamique. (Voir la section **Configuration de texte de message libre**.)

2. Activez l'option **En cas de dépassement de valeur d'une variable**.
3. Cochez les cases correspondant aux valeurs souhaitées.
   Pour les paramètres, reportez-vous à la section **Liste des paramètres de texte de valeur limite**.
4. Ordonnez les paramètres sélectionnés en effectuant un Drag&Drop ou en utilisant les boutons correspondants.
5. Assurez-vous que le dépassement de la valeur limite déclenche l'exécution de la fonction.

CONFIGURATION DU TEXTE DU MESSAGE LIBRE :

STRUCTURE

$FREETEXT;%VARIABLE%TEXT;

Signe de début -> texte libre entre points-virgules -> contient les paramètres avec le signe %%

- Signe de début : **$**
  Les paramètres de *texte libre* sont donc pris en compte.
- Texte libre :
  Inséré entre des points-virgules (**;**)
- Préfixe des **paramètres de texte de valeur limite** :
  2 signes de pourcentage (**%%** ou **%** nom de la variable **%**)

  Si aucune variable n'est indiquée, le paramètre suivant désigne la variable principale.
- Fin de la séquence de caractères : Points-virgules (**;**)

EXEMPLE

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

- **$** : Début du message avec texte du premier message
- **%%Name** : Nom de la variable présentant un dépassement de valeur et texte
- **%%LimitText** : Texte de valeur limite de la variable présentant un dépassement de valeur et texte
- **%Master%VariableStatus** : État de la variable **Master** et texte

> **Hint**
>
> Insérez un espace devant le séparateur. Les blocs individuels sont ensuite séparés correctement par un espace.

LIST LIMIT VALUE TEXT PARAMETERS (LISTE DES PARAMÈTRES DU TEXTE DE VALEUR LIMITE)

Paramètre	Variable principale - Activation par le biais d'une valeur limite	Variable principale - Activation via la liste AML	Variable supplémentaire de type liste AML/valeur limite
%%Name (Nom de variable)	Nom de la variable, correspond à la propriété **Nom**.	Nom de la variable, correspond à la propriété **Nom**.	Nom de la variable, correspond à la propriété **Nom**.
%%Identification (Identification)	Description de la variable, correspond à la propriété **Identification**.	Description de la variable, correspond à la propriété **Identification**.	Description de la variable, correspond à la propriété **Identification**.
%%LimitText (Texte de valeur limite)	Texte de valeur limite de l'objet alarme, avec interprétation du texte de valeur limite dynamique.	Texte de valeur limite de l'objet alarme, avec interprétation du texte de valeur limite dynamique.	État du texte tel qu'il est défini dans la propriété **Texte valeur limite**.
%%MessageCreateTime Stamp (Horodatage du message)	Heure de création du message. (Heure d'activation de la fonction, pas du dépassement de limite ou de l'envoi).	Heure de création du message. (Heure d'activation de la fonction, pas du dépassement de limite ou de l'envoi).	Heure de création du message. (Heure d'activation de la fonction, pas du dépassement de limite ou de l'envoi).
%%ResourceLabel (Identification de l'intitulé de ressource)	Texte de l'identification d'intitulé de ressource de la variable ; correspond à la propriété **Description externe**.	Texte de l'identification d'intitulé de ressource de la variable ; correspond à la propriété **Description externe**.	Texte de l'identification d'intitulé de ressource de la variable ; correspond à la propriété **Description externe**.
%%LimitTimeReceived (Heure de création de l'alarme)	Horodatage de l'objet alarme.	Horodatage de l'objet alarme.	-

Paramètre	Variable principale - Activation par le biais d'une valeur limite	Variable principale - Activation via la liste AML	Variable supplémentaire de type liste AML/valeur limite
%% (Valeur actuelle)	Pas de paramètre. Valeur de la variable lors de l'activation de la fonction (sans unité).	Pas de paramètre. Valeur de la variable lors de l'activation de la fonction (sans unité).	Pas de paramètre. Valeur de la variable lors de l'activation de la fonction (sans unité).
%%VariableStatus (État)	État de la variable en tant que chaîne (lors de l'activation de la fonction).	État de la variable en tant que chaîne (lors de l'activation de la fonction).	État de la variable en tant que chaîne (lors de l'activation de la fonction).
%%VariableTimeStamp (Horodatage de la variable)	Horodatage de la variable (lors de l'activation de la fonction).	Horodatage de la variable (lors de l'activation de la fonction).	Horodatage de la variable (lors de l'activation de la fonction).
%%MessageTimeAcknowledge (Temps restant pour la confirmation)	Temps restant pour la confirmation, en minutes. Disponible uniquement si la fonction de confirmation de réception demandée (à la page 84) a été activée.	Temps restant pour la confirmation, en minutes. Disponible uniquement si la fonction de confirmation de réception demandée (à la page 84) a été activée.	Temps restant pour la confirmation, en minutes. Disponible uniquement si la fonction de confirmation de réception demandée (à la page 84) a été activée.
%%Address (Adresse)	Adresse.	Adresse.	Adresse.
%%Unit Unité de mesure :	Unité technique, conformément à la propriété **Unité de mesure**.	Unité technique, conformément à la propriété **Unité de mesure**.	Unité technique, conformément à la propriété **Unité de mesure**.
%%AlarmArea (Domaine d'alarme)	Domaine d'alarme de la variable principale.	Domaine d'alarme de la variable principale.	Domaine d'alarme de la variable supplémentaire.
%%LimitTimeAcknowledged (Heure d'acquittement	--	Horodatage de l'objet alarme.	--

Paramètre	Variable principale - Activation par le biais d'une valeur limite	Variable principale - Activation via la liste AML	Variable supplémentaire de type liste AML/valeur limite
de l'alarme)			
%%UserId (Identification utilisateur)	--	Identification de l'utilisateur ayant démarré l'action. Conforme à la propriété **Identifiant utilisateur**.	--
%%UserName (Nom d'utilisateur)	--	Nom complet de l'utilisateur ayant démarré l'action. Conforme à la propriété **Nom complet**.	--
%%ComputerName (Nom de l'ordinateur)	--	Nom de l'ordinateur sur lequel a été démarrée l'action.	--
%%Comment (Commentaire)	--	Commentaire de l'objet alarme.	--
%%AlarmGroup (Groupe d'alarmes/d'événements)	Groupe d'alarmes/d'événements de l'objet alarme (sous forme de texte).	Groupe d'alarmes/d'événements de l'objet alarme (sous forme de texte).	Groupe d'alarmes/d'événements de l'objet alarme (sous forme de texte).
%%AlarmClass (Classe d'alarmes/d'événements)	Classe d'alarmes/d'événements de l'objet alarme (sous forme de texte).	Classe d'alarmes/d'événements de l'objet alarme (sous forme de texte).	Classe d'alarmes/d'événements de l'objet alarme (sous forme de texte).
%%AlarmTimeActive (Durée d'activité de l'alarme)	--	Durée pendant laquelle l'alarme est restée active.	--

GESTION DES ERREURS

Si des paramètres non valides sont utilisés, un message d'erreur correspondant est affiché dans le message :
<Paramètre non valide : "Paramètre">.

Ce texte peut être modifié et traduit avec la fonction de changement de langue. Pour cela, deux mots-clé doivent être créés :

- **<Paramètre non valide :**
- **>**

Si les paramètres sont utilisés dans un contexte incorrect, ou si un paramètre ne peut pas être lu (par exemple, parce qu'une variable n'existe pas), la chaîne **IDS_STRING2501 (---)** est ajoutée au message, à la place de la valeur. Ce texte peut être modifié et traduit avec la fonction de changement de langue.

8 Message Control in the Runtime

In the Runtime Message Control is either started automatically (property **Actif lors du démarrage du runtime** active) or with the function Send Message: activate (à la page 87).

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

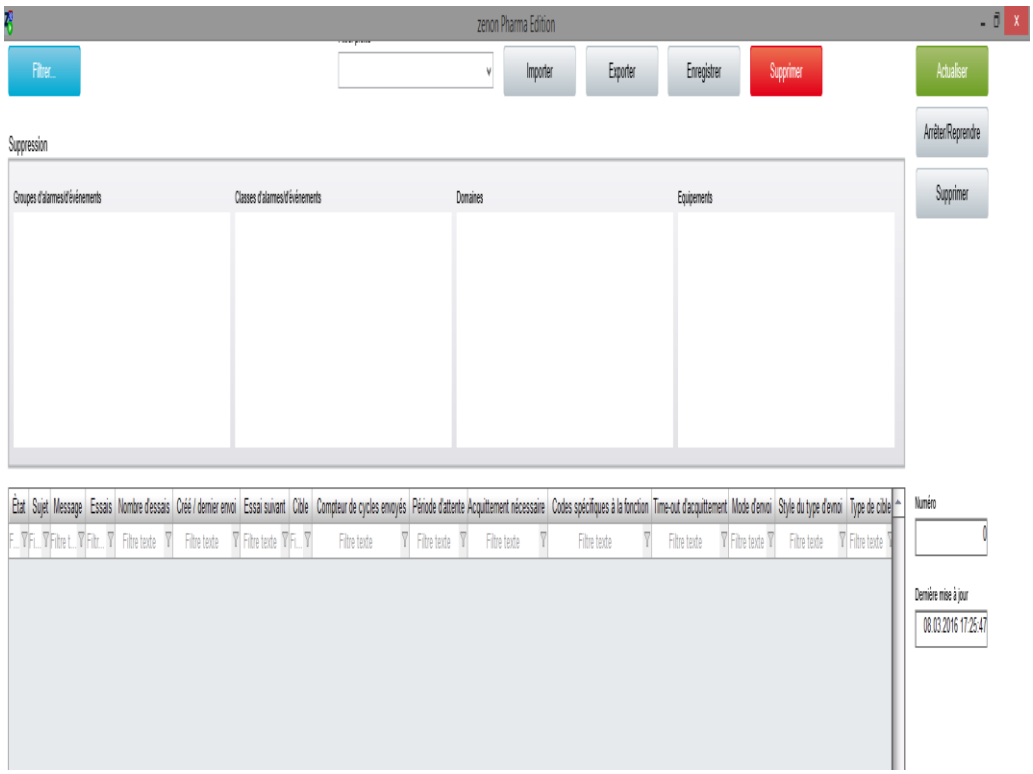
- a message configured for this is being sent
- its acknowledgement (à la page 99) is being analyzed
- the functions used for acknowledgement are executed

> **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 (à la page 96).

Message Control is being closed when closing the Runtime or by the function Send Message: deactivate (à la page 87).



TIME STAMP IN ALARM MESSAGE LIST AND MESSAGE CONTROL

When the function Send message (à la page 65) 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 (à la page 104)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 (à la page 99) 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.

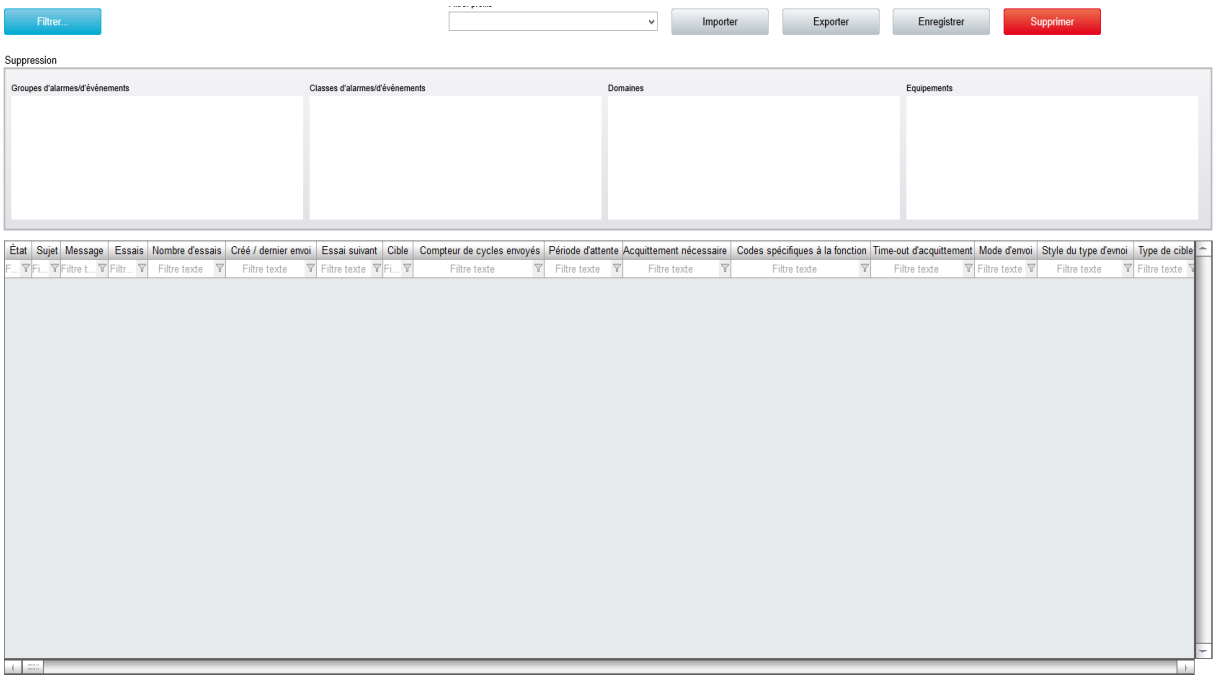
The **message list** contains messages with this status:

- 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 (à la page 55)) and can be provided with individual column headings. These labelings are localizable.



INSÉRER UN MODÈLE

Éléments de contrôle	Description
Insérer un modèle	Opens the dialog for selecting a template for the screen type. Templates are shipped together with zenon and can also be created by the user. Templates add pre-defined control elements to pre-defined 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.

FENÊTRE

Listes et champs pouvant être affichés dans le Runtime.

Éléments de contrôle	Description
File d'attente des messages	Liste de messages. Contient les messages envoyés, confirmés ou supprimés ou

Éléments de contrôle	Description
	en attente d'envoi. Les messages sont uniquement affichés, et ne peuvent plus être modifiés. Les en-têtes de colonnes peuvent être définis individuellement (à la page 57) et peuvent être localisés lors d'un changement de langue s'ils sont précédés du signe **@**.
Nombre de messages	Nombre de messages actuellement existants.
Dernière mise à jour	Heure de la dernière mise à jour.
Groupes d'alarmes/d'événements supprimés	**Groupes d'alarmes/d'événements** pour lesquels des messages ont été supprimés, et dont les alarmes ne sont donc pas affichées dans la file d'attente de messages.
Classes d'alarmes/d'événements supprimées	**Classes d'alarmes/d'événements** pour lesquelles des messages ont été supprimés, et dont les alarmes ne sont donc pas affichées dans la file d'attente de messages.
Domaines d'alarmes/d'événements supprimés	**Domaines d'alarmes/d'événements** pour lesquels des messages ont été supprimés, et dont les alarmes ne sont donc pas affichées dans la file d'attente de messages.
Équipements supprimés	**Équipements** pour lesquels des messages ont été supprimés, et dont les alarmes ne sont donc pas affichées dans la file d'attente de messages.

FONCTIONS DE LA LISTE

Éléments de contrôle des listes.

Éléments de contrôle	Description
Filtre	Ouvre une boîte de dialogue (à la page 55) de configuration des filtres pour la vue.
Supprimer	Supprime l'entrée sélectionnée de la liste. Il est uniquement possible de supprimer les messages qui n'ont pas encore été envoyés.
Actualiser	Actualise l'affichage. Les données actuelles sont chargées sur le synoptique de la file d'attente de messages. L'heure de la dernière mise à jour peut être affichée avec le champ **Last update** (Dernière mise à jour).

Éléments de contrôle	Description
Arrêter/Reprendre	Démarre et arrête la mise à jour cyclique de la liste.

PROFILS DE FILTRE

Gestion de profils

Éléments de contrôle	Description
Sélection de profil	Ouvre la boîte de dialogue permettant de sélectionner un profil.
Enregistrer	Enregistrer la configuration actuelle sous forme de profil.
Supprimer	Supprime le profil.
Importer	Ouvre la boîte de dialogue d'importation de profils depuis un fichier.
Exporter	Ouvre la boîte de dialogue d'exportation de profils depuis un fichier.

8.2 Acquittement de messages

Les messages peuvent être confirmés ou refusés par le destinataire. Selon le support, les messages doivent être impérativement acquittés, ou leur acquittement est laissé au choix de l'utilisateur.

Support	Acquittement obligatoire - : non + : oui
E-mail via Outlook (à la page 16)	-
E-mail via SMTP (à la page 17)	-
SMS via modem GSM (à la page 19)	-
SMS via passerelle SMS (à la page 22)	-
Message vocal (fichier audio) (à la page 25)	+
Message vocal (synthèse vocale) (à la page 26)	+

Si un message n'est pas acquitté, est incorrectement acquitté ou est refusé, il est transmis au destinataire de remplacement suivant. Si un sous-titre est défini, le message n'est pas renvoyé. L'état est consigné dans la liste CEL. Durant l'acquittement de messages, des fonctions liées peuvent être exécutées.

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 **Nombre maximum d'essais** property.

CONFIGURATION DE LA CONFIRMATION

Pour configurer la confirmation ou le refus du message :

1. Dans la fonction Envoi de messages (à la page 65), ouvrez l'onglet Acknowledgement of receipt (à la page 84) (Confirmation de réception).
2. Cochez la case **Acknowledgement required** (Acquittement demandé) (toujours active pour les messages vocaux).
3. Définissez le délai d'attente (Timeout) dans lequel le message doit être acquitté.
   Après expiration du délai d'attente, le message est transmis à la personne suivante sur la liste.
4. Configurez les codes de confirmation (code PIN) ou de refus (code NA) comme suit :
   a) Propriétés de l'utilisateur :
      Utilisateur Message Control -> **Code PIN** et **Code NA**
      ou
   b) Dans la fonction, via :
      La case à cocher **Utiliser des codes spécifiques aux messages** -> **Code PIN** et **Code NA**

   Le code est constitué d'un nombre à quatre chiffres compris entre *0000* et *9999*.

   Si des codes sont configurés par le biais des propriétés et la fonction, les codes de la fonction sont non valides.
   Conseil : si des codes sont configurés par le biais de la fonction, il est recommandé de créer une note au destinataire dans le texte du message (à la page 72) pour lui indiquer le code à utiliser pour activer ou refuser le message.
5. Définissez les fonctions devant être exécutées après réception du code pour indiquer que :

a) Le message a été confirmé

b) Le message n'a été ni confirmé, ni refusé

c) Le message a été refusé

6. Fermez la boîte de dialogue en cliquant sur le bouton **OK**.

ACQUITTEMENT DANS LE RUNTIME

Durant l'envoi, chaque message devant être acquitté reçoit une identification explicite individuelle (GUID). Toutes les activités du message sont attribuées par le biais de cette identification. Si un message doit être acquitté, l'identification est automatiquement ajoutée au texte du message.

Pour acquitter un message, selon le support, le destinataire doit envoyer le numéro du message et un code PIN dans un format défini.

Il est ainsi possible également d'acquitter des messages au format texte transmis par le biais d'autres modes d'envoi que les modes configurés. Les conditions suivantes doivent en outre être vraies dans le navigateur

- le message doit avoir été envoyé sous forme d'e-mail ou de SMS
- La réponse doit être transmise sous forme d'e-mail ou de SMS
- La réponse doit contenir le GUID
- La réponse doit contenir le code d'acquittement
- Le support utilisé pour la réponse doit être configuré

E-ACQUITTEMENT D'E-MAILS

L'acquittement d'un e-mail se déroule comme suit :

- La réponse est envoyée à l'aide de la fonction Répondre du client de messagerie du destinataire.
- Le sujet d'origine de l'e-mail ne doit pas être modifié. Ceci signifie : Les contenus définis dans Outlook dans **Sujet (ID)** ou, pour le protocole SMTP, dans **Sujet pour les e-mails sortants** ne doivent pas être modifiés. Ces contenus identifient et attribuent le message lors de sa réception dans zenon.
- Toutefois, le sujet doit comporter au moins trois caractères en préfixe, en guise de réponse. par exemple **RE:** ou **AW**
- Le texte de la réponse doit contenir les éléments suivants :
  - GUID : Au début du message. Aucun caractère (espaces inclus) ne doit être inséré devant.
  - Point-virgule (**;**) : Séparateur.
  - Texte : Code PIN ou code NA.

Par exemple : **43d3c61d-ccc9-4c76-bc2c-61c2d12b0db3;0246**

- Dans le module de gestion des messages, les e-mails destinés au module Gestion de messages sont identifiés par leur objet.
  Attention : l'objet de la réponse doit être identifié comme une réponse. Pour cela, le programme vérifie si le message commence par au moins trois caractères tels que **RE :** ou **AW:**.
- Le module Gestion de messages analyse le GUID et le code, et exécute ensuite les fonctions définies dans la fonction Envoi de messages (à la page 65).

Remarque : si le support SMS a été configuré, un message envoyé par e-mail peut également être acquitté par SMS.

ACQUITTEMENT DE SMS

L'acquittement d'un SMS se déroule comme suit :

- La réponse est envoyée à l'aide de la fonction Répondre du téléphone de l'utilisateur.
- Le texte de la réponse doit contenir les éléments suivants :
  - GUID : Au début du message. Aucun caractère (espaces inclus) ne doit être inséré devant.
  - Point-virgule (**;**) : Séparateur.
  - Texte : Code PIN ou code NA.

  Par exemple : **43d3c61d-ccc9-4c76-bc2c-61c2d12b0db3;0246**
- Le module Gestion de messages analyse le GUID et le code, et exécute ensuite les fonctions définies dans la fonction Envoi de messages (à la page 65).

> **Information**
>
> Les SMS envoyés à des Smartphones peuvent également être reçu et acquittés avec **Notifier App by zenon** . Pour cela, l'application envoie la réponse sous forme de SMS au format **Alarm-ID;PIN** ou **Alarm-ID;NA**.
>
> Si le support e-mail a été configuré, un message envoyé par SMS peut également être acquitté par e-mail.

MESSAGE VOCAL

Les messages vocaux doivent toujours être acquittés. Un message vocal est acquitté comme suit :

- Le téléphone du destinataire doit être compatible avec les tonalités multifréquences.
- Le modem de réception de l'équipement doit être compatible avec les tonalités multifréquences.
- Le message est lu au destinataire avec le texte de la **description**.
- Le destinataire acquitte le message en saisissant un **code PIN** ou un **code NA**.

- Le destinataire peut effectuer les actions suivantes :
  - Demander la répétition d'un message :
    En appuyant sur la touche **#** du téléphone. Le message précédemment envoyé (description, confirmation ou message d'erreur) est alors répété.
  - Réinitialiser le code envoyé :
    En appuyant sur la touche ***** du téléphone. Le code (PIN ou NA) précédemment envoyé par le destinataire est alors annulé. La lecture du message commence à nouveau par la **description**.
- Le module Gestion de messages analyse le code, et exécute ensuite les fonctions définies dans la fonction Envoi de messages (à la page 65).

Remarque : les messages vocaux doivent toujours être acquittés par téléphone.

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 **Paramètres spécifiques au projet**/**Nombre maximum d'essais** property. The time between the attempts corresponds to the value in the **Paramètres spécifiques au projet**/**Délai d'attente entre deux tentatives et délai d'attente avant expiration** property.
   Remarque : si un message est transféré à un autre utilisateur, il est considéré comme un nouveau message. Le compteur est automatiquement réinitialisé à *1*.

   Attention: The value for the project-specific **Délai d'attente entre deux tentatives et délai d'attente avant expiration** property must be greater than the value for the **Waiting time** option in the **Envoi d'un 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: **Code PIN**
   - Rejection: **Code NA**
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 (à la page 60) is being synchronized and remanently saved.

The function Send message (à la page 65) 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

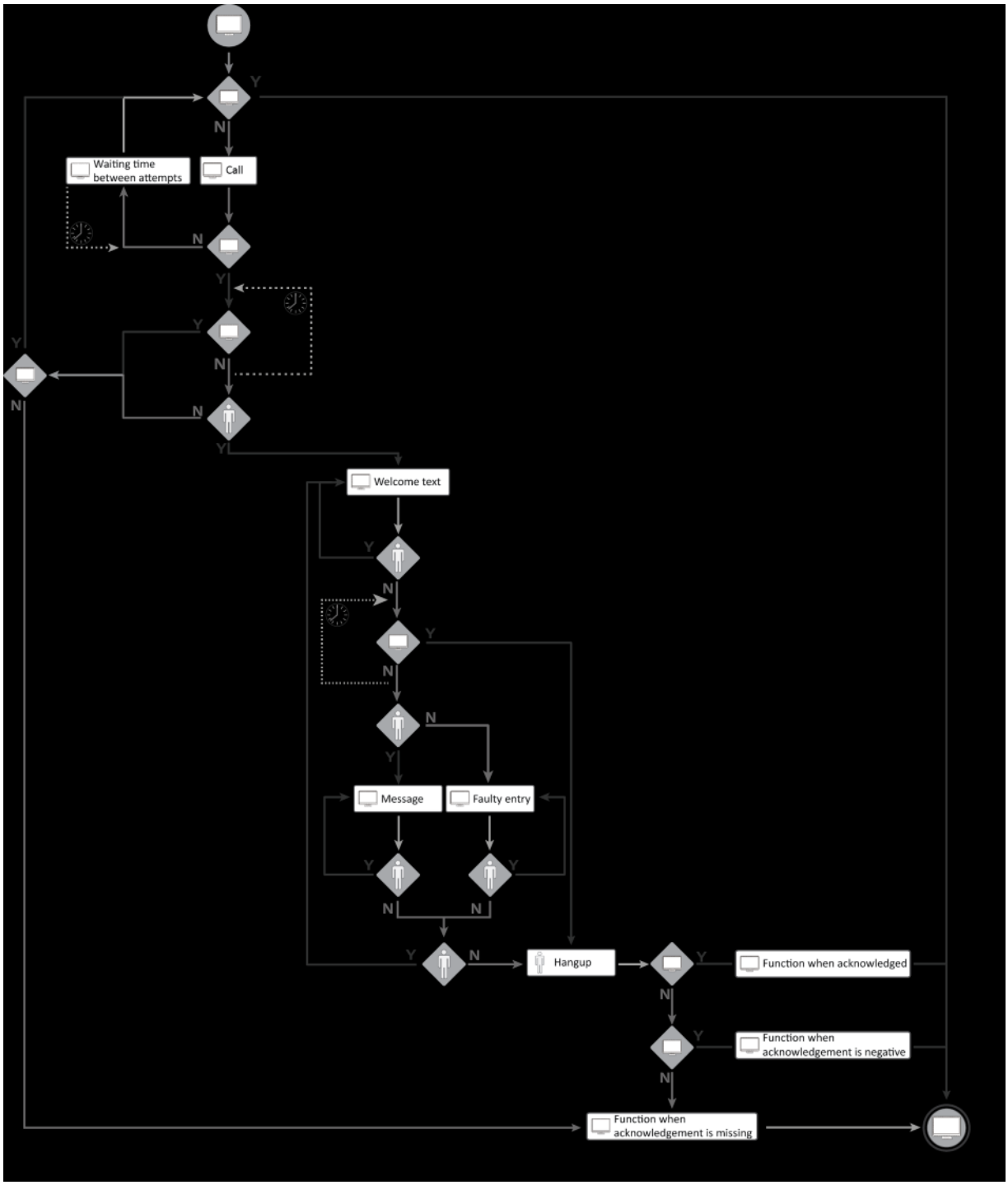
BEHAVIOR IN CASE OF FAILURE

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 **Envoi d'un message** function
- Text description
- End call

Subprocesses:

- Substitute person
- Misentry
- Acknowledgement

MAIN PROCESSES:

Start "Send message" function:

1. **Envoi d'un message** function executed
2. Validation: Has the defined value for the **Nombre maximum d'essais** function been reached?

   Remarque : si un message est transféré à un autre utilisateur, il est considéré comme un nouveau message. Le compteur est automatiquement réinitialisé à *1*.

   - **Y**: **Envoi d'un 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 **Délai d'attente entre deux tentatives et délai d'attente avant expiration** function is waited for according to the configuration. The **Envoi d'un message** function is then executed again
   - **Y**: There is a wait for a response.

   Attention: The value for the project-specific **Délai d'attente entre deux tentatives et délai d'attente avant expiration** property must be greater than the value for the **Waiting time** option in the **Envoi d'un message** functions (**Receipt Confirmation** tab).
4. Validation: Has the waiting time defined in the **Time-out [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 **Envoi d'un 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 **Envoi d'un 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 **Nombre maximum d'essais** 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 (à la page 114).

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 (à la page 11).

9.1 Check list

In case of problems check:

Problem	Possible cause	Solution
SMTP mail		
Outgoing mails are not being sent, incoming mails not received.	The sending mode is not configured correctly. LOG messages like "inactive sending mode", "server not available", "user cannot be found", "SMTP / POP 3 authentication failed" are indicating this.	▸ Is the **Versandart aktiv** property activated? ▸ Is the user who is supposed to receive the mail set as „Message Control User" (property **Utilisateur Message Control** activated)? ▸ Was an e-mail address configured for the user? ▸ Are the user data transferred to the Runtime after a change (property **Données modifiables dans le Runtime**)? ▸ Is the configuration of the sending mode correct? Compare to the data of the e-mail provider.
	The server is registered correctly, but is not available (error messages when connecting).	▸ Does the name resolution work? ▸ Is the server available (Ping)? ▸ Does the firewall allow for the connection with the configured port? Both the computer firewall and the domain firewall can block this. Check with Telnet.
	The server does not support a command (error message with "Command not supported" or similar in the LOG file).	▸ Check if the server supports the command (e.g. with Thunderbird or Telnet). If possible select a different configuration (e.g. SMTP AUTH commands can be bypassed with "*register at the incoming mail*

Problem	Possible cause	Solution
		server before sending e-mails". APOP can be deactivated and replaced by user and password.
	The server requires an encrypted connection. Since the mail is sent in unencrypted form the server disconnects.	▸ In this case the analysis with Wireshark shows that the server sends "STARTTLS". Since this command is not supported the connection should be secured via SSL or TLS.
	The server does not support the selected protection mechanism (error messages when establishing an encrypted connection).	▸ Select other protection configuration or plain text transmission.
SMS-GSM		
zenon cannot establish a connection with the modem.	▸ Wrong COM port (file-not-found error when opening the COM port). ▸ COM port busy (e.g. by the configuration software of the manufacturer, access-denied error when opening the COM port). ▸ Error when configuring the COM port/modem not responding. ▸ The modem does not support the SMS-PDU mode (AT+CMGG error message or PDU mode error message in LOG). ▸ Probably the	▸ Check COM port and modem configuration. ▸ Check PIN, if necessary release manually with the PUK.

Problem	Possible cause	Solution
	modem requires a PIN / is the PIN incorrect? (PIN error messages in LOG) ▸ Too many attempts to log on with an incorrect PIN? (PUK error message in LOG) In this case, the connection between zenon and the modem is only possible again once the PUK has been entered manually and thus the lock been lifted.	
Connection is established, but sending and receiving do not work.	▸ Has a connection first been established between the modem and the GSM network? In most cases, the connection with the network is only established after accessing the manufacturer software! ▸ Despite this the modem is not connected with the network (check with manufacturer software): Look for a location with better reception. ▸ The SMSC telephone number that has been	▸ Connect modem for the first time with manufacturer software with the GSM network. ▸ Put modem in different location. ▸ Check telephone number. Has the country code for the SMSC (00xy or +xy) been set correctly?

Problem	Possible cause	Solution
	entered is not correct. Check the number before entering it into zenon or when establishing a connection for the first time. This check can also be carried out with the **Vérification connexion** property activated. Load SMSC with HyperTerminal from modem and write. ▸ Look at error message in LOG	

9.2 LOG entries

Entries in the LOG file of the Diagnosis Viewer.

- Miscellaneous messages (à la page 114)
- E-mail via SMTP/POP (à la page 121)
- Voice over IP (à la page 126)
- SMS (à la page 126)
- OpenSSL Library (à la page 130)

9.2.1 Miscellaneous messages

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

- Debug (à la page 115)
- Deep Debug (à la page 117)
- MSG (à la page 120)

- Warnings (à la page 121)

9.2.1.1 Debug

DEBUG

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

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

Level	Entry	Description
DEBUG	*Valid NegAcknowlegde-code entered.*	A valid NA code for rejecting the message was entered.

9.2.1.2 Deepdebug

DEEP DEBUG

Level	Entry	Description
DEEPDEBUG	*A changed notification of message(%s) was received, the old message object will be updated!*	The project received a change telegram for the message with the GUID [number]. The local (previous) object is updated.
DEEPDEBUG	*A delete message(s) request was received!*	A delete request telegram was received.
DEEPDEBUG	*A getqueue request was received, sending queue.*	An alignment request was received. The current queue and the current status are being sent.
DEEPDEBUG	*A file alignment request was received, sending alignment data.*	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.
DEEPDEBUG	*A remove notification of message(%s) was received, the message object will be removed from the list. (%s)*	The project received a remove telegram for message with GUID [number]. The local (previous) object is deleted.
DEEPDEBUG	*All messages for project %s will be removed.*	All messages of project [name] are removed from Message Control.
DEEPDEBUG	*An added notification of message(%s) was received, the message object will be added to the list. (%s)*	The project received an add telegram for message with GUID [number]. The new object is added.
DEEPDEBUG	*An ChangeSuppressionRequest was received. The local suppression-state will be updated!*	A ChangeSuppression-Request was received. The local suppression status is updated.
DEEPDEBUG	*An update of the message control state was received. The local state*	The project received a MessageControl-StateChange telegram for

Level	Entry	Description
	will be updated!	message with GUID [number]. The new object is added.
DEEPDEBUG	*Current message count is (%d)*	There are [number] messages in the queue. (Entry is created when creating and deleting messages.)
DEEPDEBUG	*Firing MessageAcknowledged*	MessageAcknowledged is being fired.
DEEPDEBUG	*Firing MessageAcknowledgeTimeOut*	MessageAcknowledgeTimeOut is being fired.
DEEPDEBUG	*Firing MessageCreated*	MessageCreated is being fired.
DEEPDEBUG	*Firing MessageDeleted*	MessageDeleted is being fired.
DEEPDEBUG	*Firing MessageIn: (Identfier:'%s',Message:'%s')*	Message is being fired, Id value is %s, MessageTExt is %s.
DEEPDEBUG	*Firing MessageNegAcknowledged*	MessageNegAcknowledged is being fired.
DEEPDEBUG	*Firing MessageSendError*	MessageSendError is being fired.
DEEPDEBUG	*Firing MessageSent*	MessageSent is being fired.
DEEPDEBUG	*Last index set to %d.*	The last applied index was set to [value].
DEEPDEBUG	*MessageAcknowledged fired*	MessageAcknowledged was fired.
DEEPDEBUG	*MessageAcknowledgeTimeOut fired*	MessageAcknowledgeTimeOut was fired.
DEEPDEBUG	*MessageChangedNotify received for Message %s ignored (no longer Mainstation)!*	The project received a changed notification for message with GUID [number] from Message Control, however, it is no longer the process-executing instance.
DEEPDEBUG	*MessageCreated fired*	MessageCreated was fired.

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

Level	Entry	Description
DEEPDEBUG	*Switch to standby:%s*	The current instance is downgraded to standby server.
DEEPDEBUG	*The message %s is added to the Queue.*	The message with GUID [number] was added to the queue.
DEEPDEBUG	*The Message (%s) was removed from the project queue.*	The message with GUID [number] was removed from the queue.

9.2.1.3 MSG

LOG LEVEL MSG & MSG

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

9.2.1.4 Warnings

WARNINGS

Level	Entry	Description
WARNINGS	*A Send-Message function was executed for a Group which does not contain any Message Control users!*	A sending-message function was executed for a group which is empty or does not contain activated users for Message Control.
WARNINGS	*Index exceeded 99999, starting from 0.*	Maximum index for file name reached, starts again with 0.
WARNINGS	TAPI returned [number] available devices.	Number of modems that can be addressed via TAPI. 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.
WARNINGS	Not all required modes are supported: LINEBEARERMODE_VOICE:[ModemID],LINEMEDIAMODE_AUTOMATEDVOICE:[Modem ID],LINEFEATURE_MAKECALL:[Modem ID]-> the device [Modem name] can't be used!	Once the number of TAPI devices has been determined, these are filtered for the necessary modes. **Note:** 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.

9.2.2 E-mail via SMTP/POP

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

- Debug
- Error (à la page 124)
- Warning (à la page 125)

9.2.2.1 Debug

DEBUG

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

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

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
WARNING	*No Subject for the E-Mail specified*	No subject was specified for an outgoing mail. Though this is not an error a mail without subject may cause a spam filter to block this mail.
WARNING	*No Text for the E-Mail specified*	No text was included for an outgoing mail. Though this is not an error a mail without text may cause a spam filter to block this mail.
WARNING	*The Server did not respond an OK Code to EHLO. Response:* [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
ERRORS	**VoIP:Error in VoIP connection '%s'**	Error message from the library for the connection. ▸ **%** is replaced by the current code text for the SIP status or a specific error text. For example: **VoIP:Error in VoIP connection 'User not found'**.
ERRORS	**VoIP:Error during hardware initialization**	An error has occurred during the initialization of the hardware.
ERRORS	**VoIP:Error when trying to make a call**	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!	The inserted telephone number of the short message center is invalid. Has the country code (00xy or +xy) been set correctly? Only the contained characters (probably + as prefix, otherwise only numbers) will be checked, however, it is not being checked if the telephone number actually exists or if this really is a short message center!
ERROR	[telephone number] is not a valid telephone number!	The inserted recipient telephone number is invalid. Only the contained characters (probably + as prefix, otherwise only numbers) will be checked, however, it is not being checked if the telephone number actually exists or if this really is a short message center!

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-Porp] [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]	The modem responded with an error message to an AT command. The error message is attached to the LOG message. An AT error code may be contained in the error message. If an authentication via PIN code is required this is transferred with the command AT+CPIN. If an incorrect code is entered the modem will respond with an error to this request.
ERROR	The modem did not respond to the [AT 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.	The modem requests the Personal Unlock Key (PUK) for authentication. This is not used by Message Control and, as a consequence, the attempt to establish a connection with the modem is cancelled. After an incorrect PIN code was entered several times the SIM card in the modem requests the PUK for authentication, subsequently requiring to reset the PIN code. **Attention:** Incorrect entering of the PUK may render the SIM card useless!
ERROR	The Modem responded with an unknown PIN Status: [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]: *[response from the SMTP server]*	The outgoing mail server has not sent the expected response to the AUTH LOGIN command of the corresponding user names. The response is attached to the LOG message.
ERROR	The Server did not send an OK-Response to the AUTH LOGIN password data: *[response from the SMTP server]*	The SMTP server has not confirmed the AUTH LOGIN password data with OK. The login has failed.
ERROR	The Server did not send a Waiting-For-Data-Response to the AUTH PLAIN command:*[response from the SMTP server]*	The outgoing mail server has not sent the expected response to the AUTH PLAIN command. The response is attached to the LOG message. .
ERROR	The Server did not send an OK-Response to the AUTH PLAIN data: *[response from the SMTP*	The SMTP server has not confirmed the AUTH PLAIN data with OK. The login has failed. The

Level	Entry	Description
	server]	response is attached to the LOG message.
ERROR	SMTP AUTH should be used, but the Server neither supports AUTH LOGIN nor the minimum implementation AUTH PLAIN	SMTP authentication should be used, however, during the EHLO response the server has not listed the required commands in the list of supported extension commands. The following commands are required: AUTH LOGIN and AUTH PLAIN, the latter is the minimum implementation according to RFC 2554.
ERROR	[SMTP / POP3] not connected	The attempt was made to send data to the outgoing mail server or to retrieve data from the incoming mail server, although the server is not connected.
ERROR	The Server did not send an OK-Response to the MAIL command: *[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.
ERROR	The Server did not send an OK-Response to the RCPT command: *[response from the SMTP server]*	The outgoing mail server has not confirmed the RCPT command (adds a recipient) with OK. The response is attached to the LOG message.
ERROR	The Server did not send a Waiting-For-Data-Response to the DATA command: *[response from the SMTP server]*	The outgoing mail server has not sent the expected response to the DATA command (mail data is being transferred). The response is attached to the LOG message.
ERROR	The Server did not send an OK-Respone to the transmitted Data: *[response from the SMTP server]*	The outgoing mail server has not confirmed the transmitted mail data with OK. The response is attached to the LOG message.
ERROR	APOP Authentication is required but the Server did not send APOP Data	For authentication at the incoming mail server the APOP command is required, however, the incoming mail server has not sent the required data.
ERROR	Creating APOP Digest failed. Errorcode: [Code]. Errormessage: [String]	The MD5 hash for APOP authentication could not be created.

Level	Entry	Description
ERROR	The Server did not send an OK-Response to the [POP3-Kommando] command: [response from the POP3 server]	The incoming mail server has not confirmed a command with OK. The response is attached to the LOG message. The following commands might be affected: APOP, USER, PASS, STAT, RETR, DELE. **Note:** If the DELE command still contains the reference „Could not delete mail" the e-mail was received though could not be deleted. Since the e-mail was successfully received it is being forwarded to Message Control.
ERROR	The OK-Response to the STAT command did not have the expected Format: [response from the POP3 server]	The response of the incoming mail server to the STAT command did not correspond to the format definition according to RFC 1939. Format: **OK<space>[number of mails in mailbox]<space>[total size of all mails in mailbox in bytes]**
ERROR	[Sending / Receiving] Data failed. Errorcode: [Code]. Errormessage: [String]	Sending or receiving data via an OpenSSL network connection has failed.