COPADATA
do it your way

Manuel de zenon

Message Control

v. 7.50

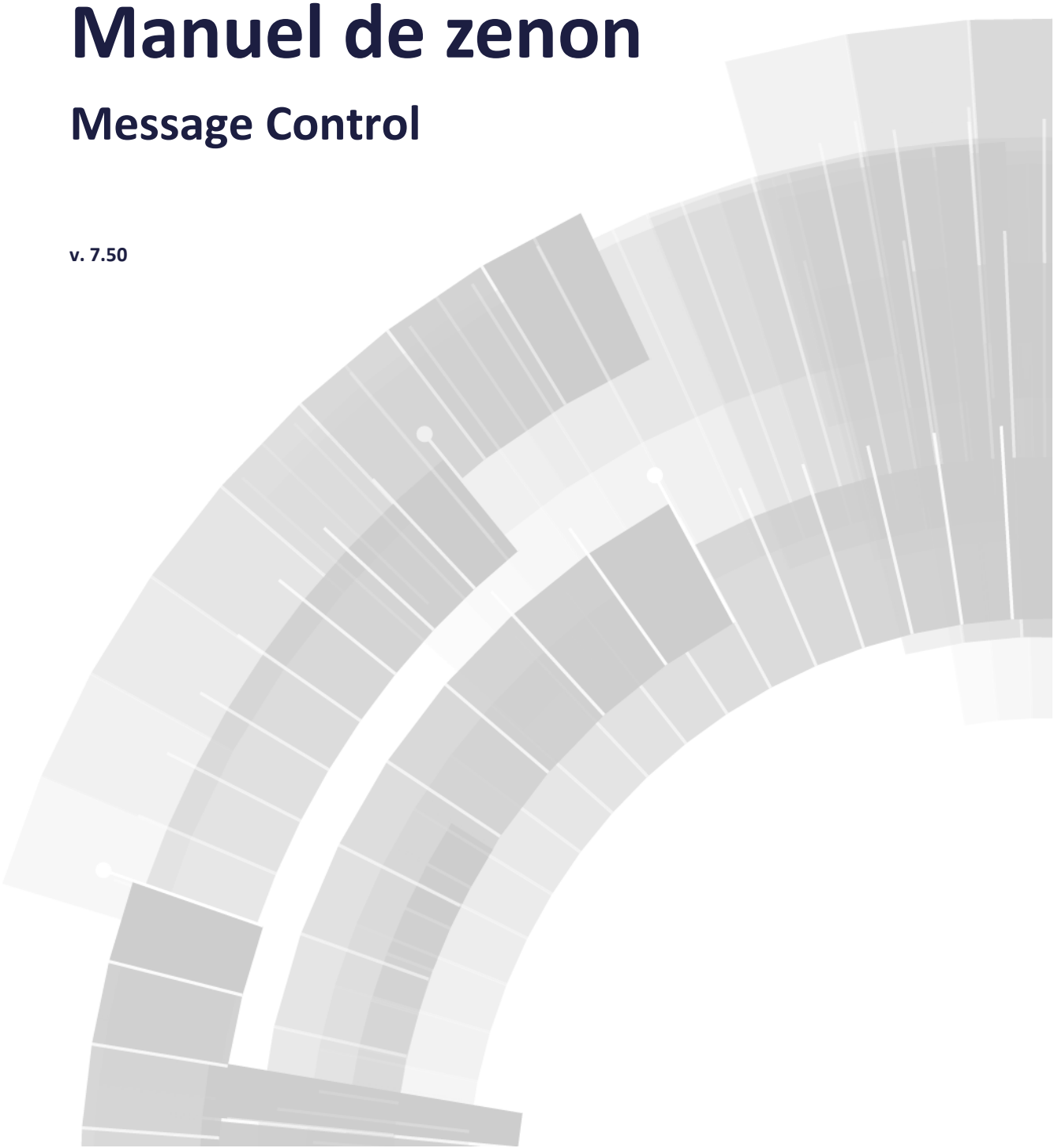


Table des matières

1. Bienvenue dans l'aide de COPA-DATA

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 (mailto:documentation@copadata.com).

ASSISTANCE PROJET

Si vous vous rendez compte que vous avez besoin de licences ou de modules supplémentaires, veuillezcontacter l'équipe commerciale par e-mail : support@copadata.com (mailto: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 (mailto:sales@copadata.com).

2. Message Control

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

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

- Voice message, text-to-speech via modem or via voice over IP to telephone

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

The configuration takes place in two stages:

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

> **Informations concernant la licence**
>
> *Must be licensed for Editor and Runtime (single-user, server, standby). Not available for clients.*

CONTEXT MENU PROJECT-SPECIFIC SETTINGS IN PROJECT MANAGER.

Menu item	Action
Help	Opens online help.

3. Compatibility with version 6.xx

CHANGES IN VERSION 7 COMPARED TO OLDER VERSIONS

À partir de la version zenon 7.00 SP0, le module de Message Control diffère fondamentalement des versions antérieures en termes de technologie et de configuration.

Des modifications techniques importantes ont été mises en œuvre :

- Le serveur COM n'est plus utilisé
- Le composant supplémentaire de l'entreprise `DerDack` n'est plus utilisé
- L'élément `ZenMsgQueue` est remplacé par un synoptique de type `Message Control`
- La configuration est effectuée par le biais de la propriété **Module Message Control** de l'espace de travail (envoi) et des propriétés **Paramètres spécifiques au projet** pour le module Message Control dans le projet (spécifique au projet)
- La vue de détail n'est plus disponible
- Les modèles de postes et la fonctionnalité de calendrier ont été supprimés

- Les fichiers ne sont plus modifiables dans le Runtime
- L'envoi d'e-mails est possible via Outlook ou un serveur SMTP ; le protocole SMTP autorise l'envoi de pièces jointes
- La configuration du type d'envoi n'est plus enregistrée dans le fichier `messag32.ini`, mais dans le fichier zenon6.ini
- Évaluation des textes de valeur limite : Jusqu'à présent, l'évaluation des textes composés dans le module Message Control se déroulait différemment de l'évaluation des textes de limite standard. À partir de la version 7.00, ces deux types de textes sont évalués de la même manière. **@StringTabelle+%var1**

Vous trouverez des informations détaillées concernant la configuration au chapitre Configuration du module Message Control (à la page 14).

Attention

Seuls les projets créés à partir de la version 5.50 SP7 peuvent être convertis au format de la version 7.

CONVERSION

En raison de ces modifications importantes, une compatibilité à 100 % ne peut pas être garantie lors de la conversion. Cela s'applique également à la compilation de fichiers RT dans les versions antérieures. Lors de la conversion, prêtez attention aux aspects suivants :

- Utilisateur :
  - Utilisateurs existants avec le même nom (prénom et nom) : L'utilisateur est sélectionné et les informations sont ajoutées.
  - Aucun utilisateur correspondant disponible : Un nouvel utilisateur est créé. Le lien vers l'utilisateur et le groupe d'utilisateurs de remplacement est résolu. L'utilisateur est ajouté au groupe existant ou au groupe créé lors de la conversion.
- Groupes d'utilisateurs :
  - Un groupe d'utilisateurs avec le même nom existe : Le groupe d'utilisateurs est sélectionné et les informations sont ajoutées.
  - Aucun groupe d'utilisateurs correspondant disponible : Un nouveau groupe d'utilisateurs est créé.
- Fonctions :

  La fonction **Afficher la base de données de destinataires** a été supprimée.
  Cette fonction ne peut plus être créée avec Editor. Lors de la conversion, toutefois, elle n'est pas supprimée. Son exécution dans le Runtime n'a aucun effet, et génère une entrée de fichier journal (à la page 104).
- Paging :
  La fonction Paging (radiomessagerie) n'est plus disponible pour les envois de messages. Les

fonctions existantes avec le type d'envoi `Paging` (radiomessagerie) sont transformées en envois du type `GSM` lors de la conversion. Un message (à la page 104) dans la fenêtre de résultat l'indique. Après la conversion, vous devez contrôler les paramètres de la fonction.

- Fichiers modifiables dans le Runtime :
  La fonction de gestion des utilisateurs ayant été modifiée, les fichiers modifiables dans le Runtime ne sont plus utilisés par le module Message Control. Dans la version 7, les données de Runtime de versions antérieures ne peuvent pas être lues. Si les fichiers de Runtime d'un projet antérieur à la version 7 sont requis, vous devez les lire dans Editor avant de les importer dans zenon 7, puis les convertir.
- Postes et calendriers :
  La fonctionnalité de gestion des postes et calendriers a été supprimée. Les fonctions existantes comportant le type de cible **Poste** reçoivent en type de cible **Groupe** lors de la conversion. Toutefois, aucun groupe n'est lié. Un message (à la page 104) dans la fenêtre de résultat l'indique. Après la conversion, vous devez contrôler les paramètres de la fonction.
- Passerelle SMS :
  Puisque l'interface simple n'offre pas la possibilité technique d'attribuer les messages de manière univoque, à partir de la version 7, seule l'interface améliorée est prise en charge. Lors de la conversion, vous devez vous assurer que l'application SMS Server de l'entreprise `Dialogs` est correctement configurée. Dans le cas contraire, l'envoi échouera.

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 7.50 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: Ensure that 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 17):	Running Microsoft Outlook instance
E-mail message via SMTP (à la page 18):	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 20):	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 23):	SMS server of DIALOGS Software GmbH
Speech output: (à la page 24)	Voice modem with DTMF functionality or voice over IP server. Text-to-speech engine. (Partially contained in operating systems. Extendable by a separate text-to-speech package from COPA-DATA.)
Telephone: (à la page 24)	Voice modem with DTMF functionality. If an ISDN adapter is used it must support the following functions. ▶ Support of voice messages (LINEMEDIAMODE_AUTOMATEDVOICE) ▶ Support of DTMF signals (for receiving the PIN code when acknowledging a message) ▶ Support of Telephony Application Programming Interface (TAPI)
Voice over IP: (à la page 24)	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
  The 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 corresponding figures).	AT+CSCA="[SMSC-telephone number]"\r	\r\nOK\r\n	at the end
Check to see if the modem has switched to SMS-PDU mode:	AT+CMGF?\r	\r\nOK\r\n	at the end
Check to see if the modem has set the given SMS message center:	AT+CSCA?\r	\r\nOK\r\n	at the end
Send message to this number. Modem signalizes Ready for PDU if the command was accepted:	AT+CMGS="[PDU length]"\r	\r\n>[space]	
PDU for the SMS to be sent to the number specified previously with completion flag:	[SMS-PDU][0x1A]	\r\nOK\r\n	at the end
List SMSs from the memory with status set by flag:	AT+CMGL=[Flag]\r	\r\nOK\r\n	at the end
Delete all SMSs with status set by flag from the memory (regardless of index):	AT+CMGD=0,[Flag]\r	\r\nOK\r\n	at the end
Delete SMS with the given index from the memory:	AT+CMGD=[Index]\r	\r\nOK\r\n	at the end

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

Arguments:

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

ERROR 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 104) chapter.

7. Configure Message Control

Message Control is configured via:

- global properties (à la page 16) of the media for sending
- project-specific properties (à la page 44) for the message
- Functions (à la page 53) which are executed in 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 from the corresponding group in **Module Message Control**.
2. Activate the **Dispatch type active** property for this group.
3. Configure (à la page 16) the medium.
4. Configure (à la page 44) the project-specific properties.
5. Configure the users (à la page 49) or user groups that are to be reached via the medium.
6. Create a Send message (à la page 64) function and

   link the function:

   a) with a button or

   b) with the property **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 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 24)

    or
  - Voice message by audio file (à la page 24)
- (à la page 76)Message text: defined for e-mails, SMS and voice messages in:
  - the Option (à la page 68) **Constant Text**

CONFIGURE THE ADMINISTRATION OF THE MESSAGE QUEUE

In order to administrate messages in the Runtime:

1. create a screen of type Message Control (à la page 46)
2. create and configure a function Screen switch (à la page 53) for this screen.
3. link the function to a button

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

MESSAGE CONTROL IN RUNTIME

In order to use Message Control in Runtime:

1. activate Message Control via
   - **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 84): If this function is carried out Message Control is started. Messages are only processed from this point of time on.

During normal closing of Runtime an image of the current message queue is saved. This image can also be created anytime in Runtime using the Save current queue (à la page 59) function. Message Control can be closed with the function Send Message: deactivate (à la page 84) in Runtime.

SUPPRESS MESSAGE

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

If during a violation of limit values the function **Send Message** (à la page 64) is linked the messages are only created and sent if neither variable nor alarm are suppressed.

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

This list can be modified with the function Suppress groups/classes/areas/equipment (à la page 59) in Runtime. Alarms and equipment can be activated or deactivated. Suppressed entries are logged in the CEL provided for the property **Enregistrer dans liste d'événements** `all confirmations` or `only negative confirmations` was selected.

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

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 17)
- E-mail message via SMTP (à la page 18)
- SMS message via GSM modem (à la page 20)
- SMS message via SMS gateway (à la page 23)
- Voice message audiofile via modem (à la page 26)
- Voice message text-to-speech via modem (à la page 27)
- Voice over IP as audio file (à la page 29)
- Voice over IP as text-to-speech (à la page 30)

In order to configure the properties:

1. highlight the working area
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 32).

Attention

These properties of the **Module Message Control** group in the workspace are not project-specific. Changes to this area are only effective in 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 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 81)for acknowledgement. This period of time is defined with the function **Send message** (tab acknowledgement (à la page 81)).

Attention

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

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

Mixed operation is not possible in any combination.

CONFIGURATION

1. Activate the sending mode via the property **Type d'envoi actif**.
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 49) and user groups.
6. Create required functions (à la page 53) 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 81)for acknowledgement. This period of time is defined with the function **Send message** (tab acknowledgement (à la page 81)).

CONFIGURATION

1. Activate the sending mode via the property **Type d'envoi actif**.
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 49) and user groups.
5. Create required functions (à la page 53) 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**.
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 49) and user groups.
8. Create required functions (à la page 53) 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.

Informations

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

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

CONSTANTS, TIMEOUTS AND PARAMETERS

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

BUFFER

Limit	Maximum value	Description
Size of send buffer and size of receive buffer for the serial port in byte	4096	The individual PDUs have a maximum size of 400 byte. During the sending of 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 SMS server of the company DIALOGS GmbH must be present and configured.

PROCEDURE

SMS SERVER CONFIGURATION

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

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

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

CONFIGURATION IN ZENON

1. Activate the sending mode via the property **Type d'envoi actif**.
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 49) and user groups.
8. Create required functions (à la page 53) 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.

7.1.5 Voice messages

Voice messages can be sent as:

- Audio file via modem (à la page 26)
- Text-to-speech via modem (à la page 27)
- Voice over IP as audio file (à la page 29)
- Voice over IP as text-to-speech (à la page 30)

You can read information about the process in Runtime in the subchapters and in the **Voice messages in Runtime** (à la page 98) 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

The following must be the case in order to send messages via voice over IP:

- Voice over IP must be configured:
  - Server address
  - User name
  - Password
- The telephone of the recipient must support DTMF
- The VoIP provider supports the **SIP** and **RTP** protocols
- The firewall has the corresponding ports open
  - SIP (default: 5060)
  - RTP (default 4000)
  - RTCP (default 4001)

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

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

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 49) and user groups.
6. Create required functions (à la page 53) 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 **#** button.
   Then the PIN must be entered. Only then is the message played back.
4. The message must be acknowledged (à la page 94) 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 * key and entering the code again. PIN can thus be changed to NA, for example.

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

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 49) and user groups.
7. Create required functions (à la page 53) 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 **#** button. 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

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

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 (default: `5060`)
  - RTP (default `4000`)
  - RTCP (default `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 49) and user groups.
5. Create required functions (à la page 53) 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 94) 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.

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 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
- The VoIP provider supports the **SIP** and **RTP** protocols
- The firewall has the corresponding ports open
  - SIP (default: `5060`)
  - RTP (default `4000`)
  - RTCP (default `4001`)

LANGUAGE FILES

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

To do this:

1. Close the Editor.
2. Select the correct speech for the Editor (**sapi.cpl**):
   - 32-bit-compatible speech for the 32-bit Editor:
     `C:\Windows\SysWOW64\Speech\SpeechUX\sapi.cpl`
   - 64-bit-compatible speech for the 64-bit Editor:
     `C:\Windows\System32\Speech\SpeechUX\sapi.cpl`
3. Start the Editor and select the configured speech in the **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 49) and user groups.
6. Create required functions (à la page 53) 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.

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

7.1.6 zenon6.ini entries

Message Control is mainly configured via global (à la page 16) 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:

Entry	Description
[MESSAGE CONTROL]	Settings for the module Message Control. Recommendation: Configuration via the properties of the **Module Message Control** group in the workspace including subgroups and the properties of the **Paramètres spécifiques au projet** group in the **Message Control** node in the project tree.
GSM_SMS=	Activation of SMS via GMS as sending type. ▸ 0: active ▸ not 0: inactive Default: 0 The entry in Message32.ini corresponds to the entry **[GSM] On** This is taken into account when importing/exporting the ini settings. It corresponds to the property **Type d'envoi actif** in group **Message SMS (modem GSM)** in the Editor.
GSM_SMS_COM=	COM port that is used for the connection to the modem. Default: empty It corresponds to the **Connexion modem (série)** property in the Editor.
GSM_SMS_PIN=	PIN code which is used for authentication towards the modem. Default: empty It corresponds to the **Code PIN** property in the Editor.
GSM_SMS_SMSC=	Telephone number of the message center of the GSM provider. Default: empty It corresponds to the **Numéro du centre SMS** property in the Editor.
GSM_BULK_DELETE=	Behavior when deleting an SMS: ▸ 0: Messages are deleted with the **Index** method. SMSs that have been read are deleted individually ▸ 1: Messages are deleted with the **Statusflag** method. All SMSs that have been read are deleted at the same time. Note: This method is faster and more powerful, but is not supported by all modems. Default: 0 It corresponds to the **Supprime SMS avec la syntaxe d'état de la commande AT+CMGD** property in the Editor.
Outlook=	Email notification via Outlook:

	▸ 0: inactive ▸ 1: active Default: 0 It corresponds to the property **Type d'envoi actif** in group **Message e-mail (Outlook)** in the Editor.

Outlook_Profile=	Name of the Outlook profile which should be used for sending. Default: empty It corresponds to the **Profil** property in the Editor.
POP_APOP=	Controls authentication at the incoming mail server with messaging via SMTP/POP. ▸ 0: User (**USER**) and password (**PASS**) are used for authentication. ▸ 1: The **APOP** command is used for authentication instead of user and password. Default: 0 It corresponds to the property **Utiliser APOP pour l'authentification** in group **Paramètres e-mail (SMTP)** in the Editor.
POP_KEEP_MAILS=	Defines whether e-mails that have been read remain on the server: ▸ 0: E-mails are deleted after they have been fetched from the server, ▸ 1: E-mails remain after they have been fetched from the server. Default: 0 Note: As POP3 offers no possibility to filter mails when fetching them, always all mails which are in the incoming mail are fetched. If this property is active, the server's memory consumption increases and the performance can decrease when fetching mails. It corresponds to the **Laisser les mails lus sur le serveur** property in the Editor.
POP_PASSWORD=	Password for login at the incoming mail server (POP3). The password is saved encrypted and is only decrypted for authentication purposes. Must only be configured in the Editor. Default: empty It corresponds to the property **Mot de passe** in group **Serveur de mails entrants** in the Editor.
POP_POLL_INTERVALL=	Minimum period between two POP3 requests in seconds. Value should not be below the POP3 server guideline. ▸ Minimum: 10 ▸ Maximum: 4294967295 Default: 60 It corresponds to the **Temps minimum entre deux requêtes (s)**

	property in the Editor.
POP_PORT=	Port in which the POP3 server is addressed. ▸ Maximum value: 6553 Default for: ▸ Insecure connection: 110 ▸ Secure connection: 995 It corresponds to the property **Port** in group **Serveur de mails entrants** in the Editor.
POP_SECURITY=	Type of connection protection to the POP3 Server. ▸ 0, no security ▸ 1: SSLv2 and SSLv3 ▸ 2: TLSv1 Default: 0 It corresponds to the property **Encryptage** in group **Serveur de mails entrants** in the Editor.
POP_SERVER=	Address of the POP3 server. Default: empty It corresponds to the property **Serveur de mails** in group **Serveur de mails entrants** in the Editor.
POP_USER=	User name for the incoming server. Default: empty It corresponds to the property **Identifiant utilisateur** in group **Serveur de mails entrants** in the Editor.
SMSGateway=	SMS notification via SMS gateway: ▸ 0: inactive ▸ 1: active Default: 0 It corresponds to the property **Type d'envoi actif** in group **Message SMS (passerelle SMS)** in the Editor.
SMSGateway_Inbox=	Incoming folder of the SMS server for sending SMS. Default: empty It corresponds to the **Dossier de boite d'envoi** property in the

	Editor.
SMSGateway_OriginId=	Sender identification for sending SMS. Default: empty It corresponds to the **ID expéditeur** property in the Editor.
SMSGateway_Outbox=	Outgoing folder of the SMS server for sending SMS. Default: empty It corresponds to the **Dossier de boîte d'envoi** property in the Editor.
SMSGateway_Prefix=	First letter of the SMS files. Must be unique for each project. Length: 1 characters Default: `F` It corresponds to the **Première lettre du fichier SMS** property in the Editor.
SMSGateway_SemaphorPrefix=	First letter of the lock file. Length: 1 characters Default: `S` It corresponds to the **Première lettre du fichier 'lock/semaphore'** property in the Editor.
SMSGateway_SmartAlarm=	Smart alarming is used. Must be unique for each project. ▸ `0`: inactive ▸ `1`: active Default: `0` It corresponds to the **Gestion d'alarmes intelligente** property in the Editor.
SMSGateway_Statusbox=	Folder for the status message at SMS sending. Default: empty It corresponds to the **Dossier d'état** property in the Editor.
SMSGateway_TimeOut=	Timeout for outgoing messages in minutes. Defines after what period of time a message is interpreted as "not send successfully". Default: `60` It corresponds to the **Timeout** property in the Editor.

SMTP_AUTH=	Authentication at the outgoing server. ▸ 0, no security ▸ 1: log in to the POP3 Server before sending ▸ 2: SMTP AUTH with signing in to the SMTP server before dispatch Default: 0 It corresponds to the property **Auhentification pour le serveur de mails sortants** in group **Serveur de mails sortants** in the Editor.
SMTP_OTHER_CREDS=	Defines whether the outgoing server uses different log in data than the incoming server. ▸ 0: inactive ▸ 1: active Default: 0 Corresponds to the property in the **Serveur de mails sortants** group in the Editor.
SMTP_PASSWORD=	Hex dump of the encrypted password for authentication at the outgoing server. Default: empty Corresponds to the property in the **Serveur de mails sortants** group in the Editor.
SMTP_OTHER_CREDS=	Sign-in data for SMTP server. ▸ 0: Login data for the incoming mail server is also used for the outgoing mail server. ▸ 1: For the authentication at the outgoing mail server different login data than at the incoming mail server are used. Default: 0 It corresponds to the property **Utiliser un 'log' pour les données différent de celui du serveur de mails entrants** in group **Serveur de mails sortants** in the Editor.
SMTP_OUT_ADDR=	Address for outgoing e-mails. Default: empty It corresponds to the property **Adresse pour les e-mails sortants** in group **Serveur de mails sortants** in the Editor.

SMTP_POP_MAIL=	Email notification via SMTP/POP: ▸ 1: active ▸ 0: inactive Default: 0 It corresponds to the property **Type d'envoi actif** in group **Serveur de mails sortants** in the Editor.
SMTP_PORT=	Defines the used port at the SMTP Server. Maximum: 65535 Default: 25 It corresponds to the property **Port** in group **Serveur de mails sortants** in the Editor.
SMTP_SECURITY=	Type of encryption for connection to the SMTP Server. ▸ 0, no security ▸ 1: SSLv2 and SSLv3 ▸ 2: TLSv1 Default: 0 It corresponds to the property **Encryptage** in group **Serveur de mails sortants** in the Editor.
SMTP_SERVER=	The SMTP Server entered by the user. Default: empty It corresponds to the property **Adresse du serveur** in group **Serveur de mails sortants** in the Editor.
SMTP_SRV_IS_POP=	This entry defines whether the POP3 Server is used as SMTP Server. ▸ 1: POP 3 is SMTP ▸ 0: POP3 and SMTP are different Servers Default: 0 It corresponds to the property **Utiliser le serveur de mails entrants pour les mails sortants** in group **Serveur de mails sortants** in the Editor.
SMTP_SUBJECT=	Subject for outgoing e-mails and for detecting whether an incoming e-mail at the server is relevant for the sending type. Incoming e-mails must have this text as subject in order to be processed by the system. E-mails which do not contain this subject are neither passed on to Message Control nor deleted from the server. Default: **MsgCtrl_Alert:**

	It corresponds to the property **Sujet pour les e-mails sortants** in group **Serveur de mails sortants** in the Editor.
SMTP_USER=	User name saved for the outgoing server. Default: empty It corresponds to the property **Identifiant utilisateur** in group **Serveur de mails sortants** in the Editor.
SMTP_USER_IS_ADDR=	Defines whether the user name for authentication towards the outgoing server is used as sender address for outgoing mails. Is only used if the **SMTP_AUTH=** entry is not `0`. ▸ `1`: active ▸ `0`: inactive Default: `0` It corresponds to the property **Le nom de l'utilisateur est l'adresse pour les e-mails sortants** in group **Serveur de mails sortants** in the Editor.
Speech=	Text-to-Speech notification via modem: ▸ `0`: inactive ▸ `1`: active Default: `0` It corresponds to the property **Modem vocal (Text-to-Speech)** in group **Message vocal** in the Editor.
Speech_Name=	Selection of speech and language for text-to-speech. Ensure that the correct voices for the Editor have been selected: ▸ 32-bit Editor: `C:\Windows\SysWOW64\Speech\SpeechUX\sapi.cpl` ▸ 64-bit Editor: `C:\Windows\System32\Speech\SpeechUX\sapi.cpl` Default: empty It corresponds to the property **Voix :** in group **Text-to-Speech** in the Editor.

Speech_Rate=	Speech speed. ▸ Minimum: -10 ▸ Maximum: 10 Default: 0 It corresponds to the property **Vitesse speech** in group **Text-to-Speech** in the Editor.
Speech_Volume=	Speech volume. Number equals the percent value of the maximum value of the selected speech. ▸ Maximum: 100 ▸ Minimum: 0 Number taken from the operating system without change. Default: 0 It corresponds to the property **Volume** in group **Text-to-Speech** in the Editor.
Subject=	Unique ID which incoming e-mails must contain in order to be processed in Message Control. Default: **MsgCtrl_Alert:** It corresponds to the property **Sujet (ID)** in group **Message e-mail (Outlook)** in the Editor.
Telephone=	Notification by means of audio file via modem: ▸ 0: inactive ▸ 1: active Default: 0 It corresponds to the property **Modem vocal (fichier audio)** in group **Message vocal** in the Editor.
Telephone_IgnoreDisconnect=	Behavior in the event of a loss of connection:: ▸ 1: A disconnection (e.g. recipient ends call) is ignored and the message is played back completely before the line is closed. ▸ 0: Message is aborted when the connection breaks. Default: 0 It corresponds to the property **Ignorer la déconnexion** in group **Message vocal** in the Editor.
Telephone_Line=	Entry of the modem to be used. Must already be configured on the computer. It corresponds to the property **Nom ligne** in group **Paramètres téléphone** in the Editor.

Telephone_LineId=	Automatically created device ID which identifies the selected modem. Serves - for several modems with the same name - the purpose of distinguishing the devices. Attention: For information only. Must not be changed here. It corresponds to the property **ID ligne** in group **Paramètres téléphone** in the Editor.
Telephone_Timeout=	Time in minutes after which a standing condition should be canceled and closed. Time interval must be longer as the time needed for playing back and confirming the message. Default: 1 It corresponds to the property **Time-out [min]** in group **Message vocal** in the Editor.
Telephone_WelcomeMessageCount=	Number of repetitions for the welcome text. Default: 5 It corresponds to the property **Répéter le texte d'accueil** in group **Message vocal** in the Editor.
***VOIP_AUDIO*=**	Voice message as audio file by means of Voice over IP: ▸ 0: inactive ▸ 1: **Voix par IP (Fichier audio)** Default: 0 It corresponds to the property **Voix par IP (Fichier audio)** in group **Message vocal** in the Editor. Note: 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=	Server address of the VoIP provider. Default: (empty) It corresponds to the property **Adresse du serveur** in group **Voix par IP** in the Editor.
VOIP_PASSWORD=	Password for VoIP access. Is saved in encrypted form and must only be changed in the user interface. Default: `(empty)` It corresponds to the property **Mot de passe** in group **Voix par IP** in the Editor.
VOIP_RTP=	Number of the RTP port for VoIP. Default: 4000 It corresponds to the property **Port RTP** in group **Voix par IP** in the Editor.
VOIP_SIP=	Number of the SIP port for VoIP. Default: 5060 It corresponds to the property **Port SIP** in group **Voix par IP** in the Editor.
VOIP_TTS=	Voice message as text-to-speech by means of voice over IP: ▸ `0`: inactive ▸ 1: **Voix par IP (Text-to-Speech)** Default: `0` It corresponds to the property **Voix par IP (Text-to-Speech)** in group **Message vocal** in the Editor. Note: 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=	User name for VoIP access. Default: `(empty)` It corresponds to the property **Identifiant utilisateur** in group **Voix par IP** in the Editor.

Informations

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

- **SMTP_SERVER= -> 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 16)for sending media are configured in the properties **Module Message Control** of the working area.

ACTIVATING MESSAGE CONTROL FOR THE PROJECT

In Runtime, Message Control 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 84): 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 84) in 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

> **Informations**
>
> 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

It comprises:

1. Static part: unalterable subject which is defined in the global settings (à la page 16) 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:

- Text parts are separated from one another with a semi colon (**;**).
- **$**: marks text that contains the parameter.
- A parameter (à la page 84) can:
  - Be a key from the language table, such as **@MyText;**
  - Relate to a variable, when executing the function via AML or a limit value violation such as **%Var1;**
  - Be a compiled entry in the language table: for example **%@Var2+MultipleText**

- @ marks language switching
- **%** marks variables
- **%%** marks **limit value text parameters** (à la page 86) for variables

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

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

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

DÉVELOPPEMENT

Étapes de création du synoptique :

1. Créez un nouveau synoptique :

   Dans la barre d'outils ou le menu contextuel du nœud **Synoptiques**, sélectionnez la commande **Nouveau synoptique**.
   Un synoptique vide `Standard` est créé.

2. Modifiez les propriétés du synoptique :

   a) Nommez le synoptique dans la propriété **Nom**.

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

   c) Sélectionnez le cadre souhaité dans la propriété **Cadre**.

3. Configurez le contenu du synoptique :

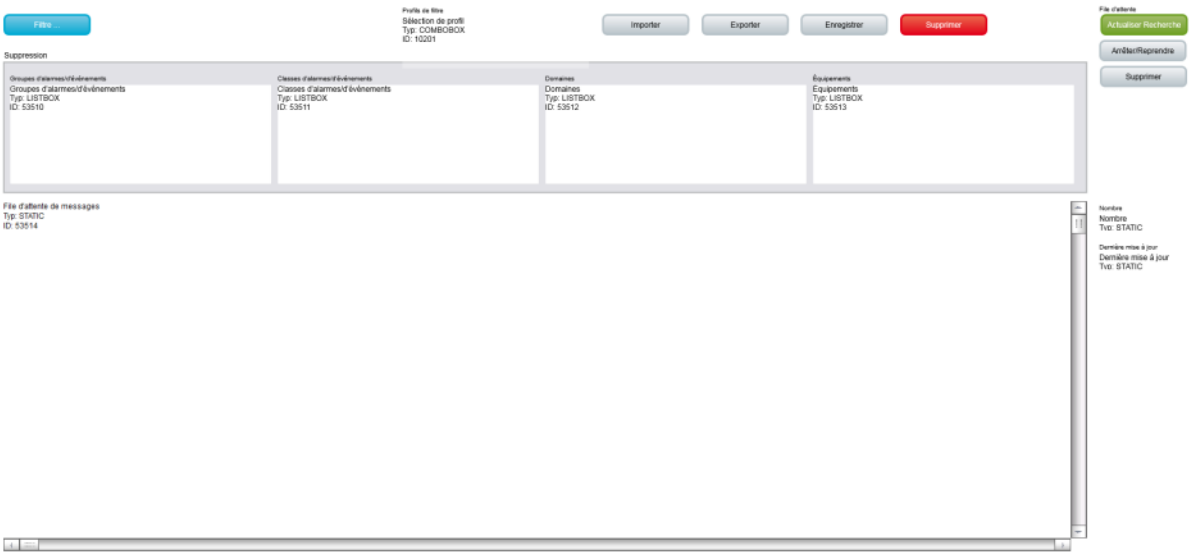
   a) Sélectionnez l'option de menu **Éléments de contrôle** dans la barre de menus

   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. Créez une fonction d'appel de synoptique.



Control elements	Description
Insert template	Ouvre la boîte de dialogue de sélection de modèle pour le type de synoptique correspondant. Des modèles sont fournis avec zenon, et peuvent également être créés par l'utilisateur. Modèle ajoute des éléments de contrôle prédéfinis à des emplacements prédéfinis sur le synoptique. Les éléments qui ne sont pas nécessaires peuvent également être supprimés individuellement après leur création. Les éléments supplémentaires peuvent être sélectionnés dans la liste déroulante, puis insérés dans le synoptique. Les éléments peuvent être déplacés et disposés individuellement sur le synoptique.
Buttons	Buttons for Runtime actions.
Refresh	Refreshes view. The current data are loaded onto the screen from the message queue. The point of time of the last update can be displayed with the field **Last update**.
Delete	Deletes selected entry from the list. It is only possible to delete messages which have not been sent yet.
Filter	Opens Dialog (à la page 54) to configure the filters for the view.
Stop/continue	Starts and stops the cyclic update of the list.
Window	Lists and fields which can be displayed in Runtime.
Number of messages	Number of currently existing messages.
Last Update	Point of time of last update.
Suppressed alarm/event groups	**Alarm/event groups** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Suppressed alarm/event classes	**Alarm/event classes** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Suppressed alarm/event areas	**Alarm/event areas** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Suppressed equipment	**Equipment** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Message queue	List of messages. Contains messages which have been sent, confirmed or deleted or remain to be sent. The messages are only displayed and cannot be edited anymore. Column headings can be named individually (à la page 56) and are localizable by putting @ before them.

Filter profiles	Profile administration
Profile selection	Opens the dialog for selecting a profile.
Save	Saves current setting as a profile.
Delete	Deletes profile.
Import	Opens dialog for importing profiles from a file.
Export	Opens dialog for exporting profiles from a file.

Informations

*The columns of the list can be confirgured via the Filter (à la page 54) for the function screen switching in the Editor, in Runtime via screen switching -> show dialog in the RT or the button **Filter**. Columns of the list can be provided with individual labels and are localizable.*

7.4 User Administration

Users and user groups for Message Control are created and managed in the zenonuser 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.

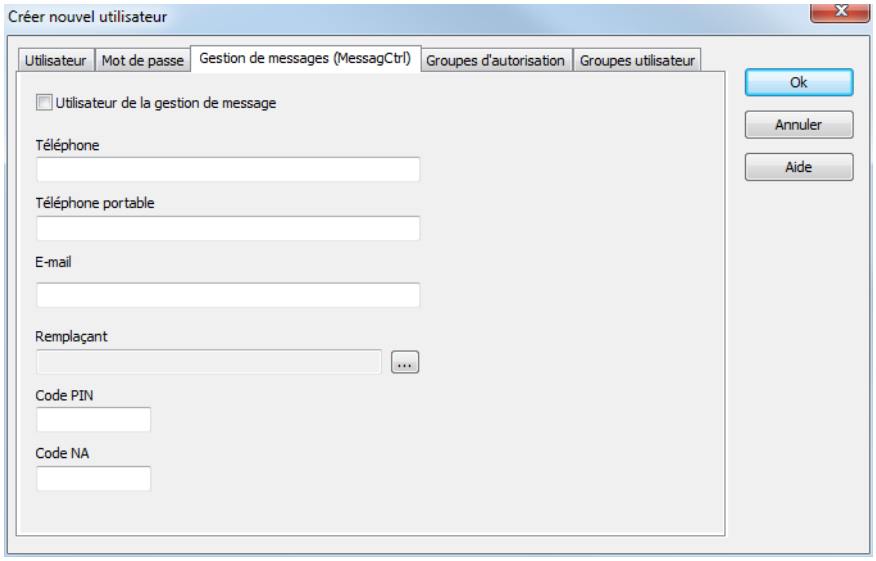
Attention

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 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


Créer nouvel utilisateur
Utilisateur
Mot de passe
Gestion de messages (MessagCtrl)
Groupes d'autorisation
Groupes utilisateur
Utilisateur de la gestion de message
Téléphone
Téléphone portable
E-mail
Remplaçant
Code PIN
Code NA
Ok
Annuler
Aide

Paramètres	Description
Utilisateur du module Message Control	`Active` : L'utilisateur est utilisé par le module Message Control.
Téléphone	Numéro de l'équipement téléphonique compatible voix de l'utilisateur. Utilisé pour la fonction de synthèse vocale. Saisissez le numéro. En outre, les pratiques suivantes sont autorisées : ▸ Le préfixe **+** peut être utilisé à la place des chiffres `00` précédant le code international. ▸ Les séparateurs suivants sont également autorisés dans l'interface de gestion des utilisateurs d'AD : Moins (**-**), tiret (**/**) et **espace**. Remarque : Lors des communications entre AD et le module Message Control, les séparateurs sont ignorés dès que les données provenant d'AD sont associées à un objet zenon.
Téléphone portable	Numéro de téléphone cellulaire de l'utilisateur. Utilisé pour l'envoi de messages par téléphone portable et SMS ("textos"). Saisissez le numéro. En outre, les pratiques suivantes sont autorisées : ▸ Le préfixe **+** peut être utilisé à la place des chiffres `00` précédant le code international. ▸ Les séparateurs suivants sont également autorisés dans l'interface de gestion des utilisateurs d'AD : Moins (**-**), tiret (**/**) et **espace**. Remarque : Lors des communications entre AD et le module Message Control, les séparateurs sont ignorés dès que les données provenant d'AD sont associées à un objet zenon.
E-mail	Adresse e-mail de l'utilisateur
Personne de remplacement	Si un utilisateur n'est pas joignable ou n'accepte pas le message, un remplaçant peut être indiqué. Cliquez sur le bouton **...** Ouvre la boîte de dialogue de sélection d'utilisateurs. Seuls les utilisateurs activés en tant qu'utilisateurs de **Message Control** peuvent être sélectionnés.
Code PIN	Code PIN permettant à l'utilisateur de confirmer le message.
Code NA	Code d'indisponibilité permettant à l'utilisateur de refuser de confirmer la réception du message (indisponible). Le message est ensuite envoyé à l'utilisateur suivant dans la liste. Si aucune autre utilisateur n'est défini dans la liste, le message est enregistré avec l'état `non acquitté`. La fonction associée à cet état est exécutée. En outre, dans tous les cas, une entrée "`refusé par`" est générée dans la liste CEL. Remarque : D'autres informations concernant l'attribution de fonctions sont disponibles au chapitre Confirmation de réception - Paramètres de la confirmation de réception (à la page 81).

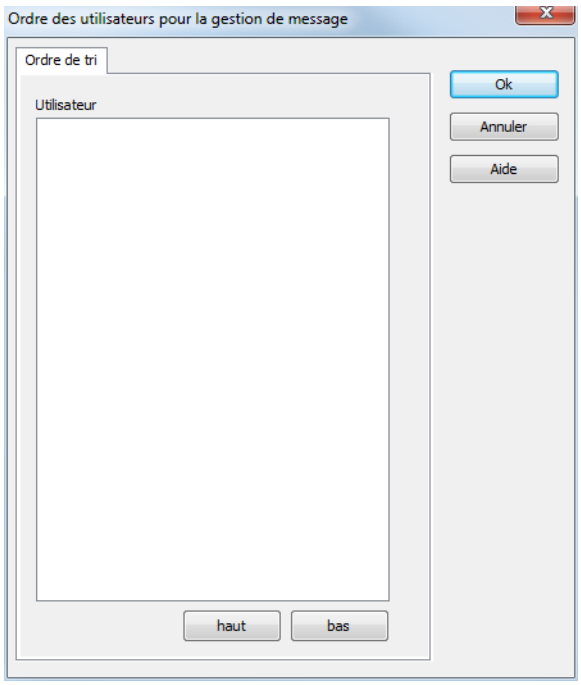
FERMER BOÎTE DE DIALOGUE

Paramètres	Description
OK	Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Cancel	Annule toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Help	Ouvre l'aide en ligne.

> **Attention**
>
> *Les codes d'acquittement PIN (confirmation) et NA (refus) doivent être différents et ne doivent pas être trop semblables.*
>
> *Si les deux codes sont identiques, le code est interprété comme un code PIN, et indique alors la confirmation du message.*
>
> *En cas de réception d'un code inconnu, un SMS et un e--mail sont envoyés au contact secondaire. Le message d'erreur est diffusé pour les messages vocaux.*

SEQUENCE WITHIN THE USER GROUP

Users can be sequenced through the property **Ordre des utilisateurs** within a user group.



Paramètres	Description
Utilisateurs	Liste de tous les utilisateurs disponibles.
Vers le haut	Déplace l'utilisateur sélectionné d'une place.
Vers le bas	Abaisse l'utilisateur sélectionné d'une place.
Ok	Applique les paramètres et ferme la boîte de dialogue.
Annuler	Annule toutes les modifications et ferme la boîte de dialogue.
Aide	Ouvre l'aide en ligne.

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

Informations

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

7.5 Functions

Functions control the use of Message Control in Runtime.

With this:

- Message control is activated (à la page 84) and deactivated (à la page 84)
- messages are sent (à la page 64)
- the screen of type Message Control is connected (à la page 53) for managing the message queue
- the message queue is saved (à la page 59)
- elements for messages are disabled (à la page 59)

7.5.1 Screen switching to a screen of type Message Control

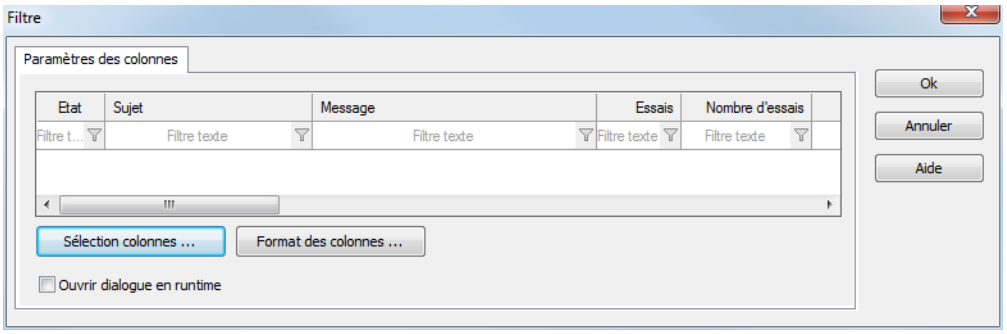
With a screen switching of type Message Control you access a in for Runtime in order to supervise the message queue. To configure screen switching:

1. Create a new function
2. select **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 54) is opened
6. configure the columns
7. close the configuration by clicking **OK**
8. connect the function with a button in order to be able to access it for Runtime

Column settings

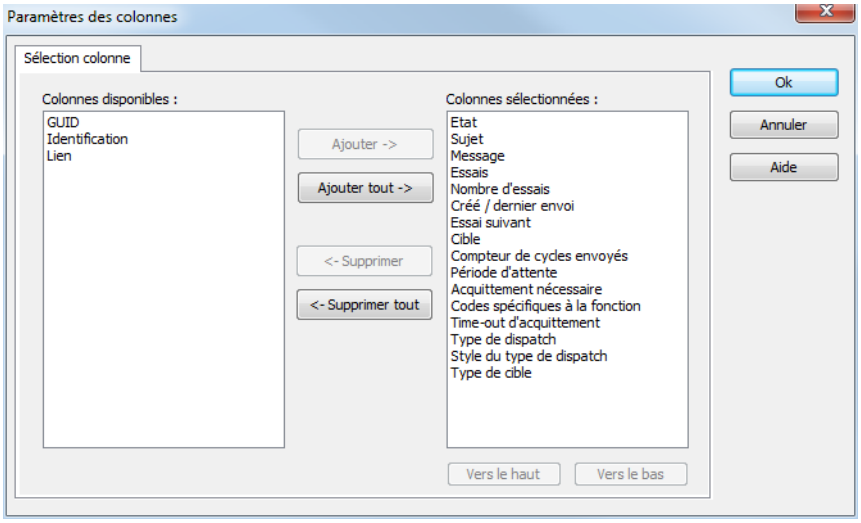
Settings for filtering messages



Paramètres	Description
Champ de liste	Affichage des colonnes configurées.
Sélection de colonnes	Ouvre la boîte de dialogue de sélection des colonnes.
Format des colonnes	Ouvre la boîte de dialogue de mise en forme des colonnes.
OK	Applique toutes les modifications et ferme la boîte de dialogue.
Annuler	Annule toutes les modifications et ferme la boîte de dialogue.
Aide	Ouvre l'aide en ligne.

Column selection

Definition of the columns displayed in the Runtime.



Bouton	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.
<- 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.
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.
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.

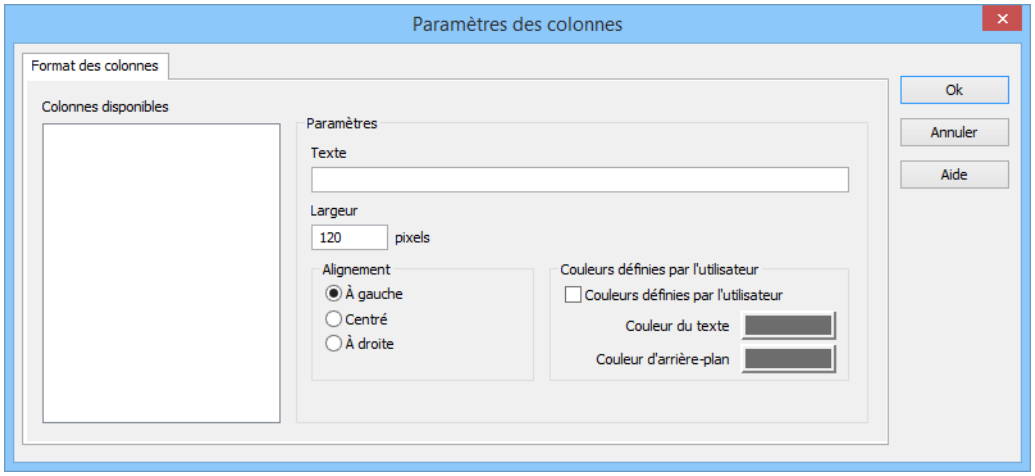
FERMER BOÎTE DE DIALOGUE

Paramètres	Description
OK	Applique les paramètres et ferme la boîte de dialogue.
Annuler	Annule toutes les modifications et ferme la boîte de dialogue.
Aide	Ouvre l'aide en ligne.

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

Paramètres	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

Paramètres	Description
Paramètres	Paramètres de la colonne sélectionnée.
Texte	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 : ▸ **Justifié à gauche** : Le texte est justifié contre le bord gauche de la colonne. ▸ **Centré** : Le texte est centré dans la colonne. ▸ **Justifié à droite** : Le texte est justifié contre le bord droit de la colonne.
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 : ▸ Module Batch Control ▸ Extended Trend ▸ Synoptiques de filtre ▸ Module Message Control ▸ Recipe Goup Manager
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.

FERMER BOÎTE DE DIALOGUE

Paramètres	Description
OK	Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Cancel	Annule toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue.
Help	Ouvre l'aide en ligne.

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 closing the Runtime. To engineer the function:

1. Create a new function
2. in group **Module Message Control** select the function **Enregistrer queue courante**
3. connect the function with a button in order to be able to access it for Runtime

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

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

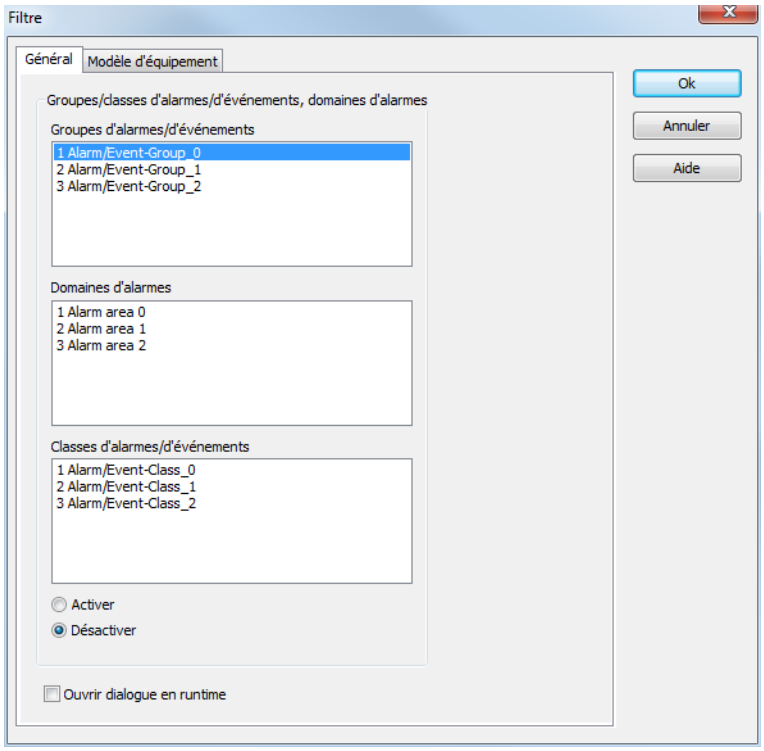
Suppressed entries are logged in the CEL provided for the property **Enregistrer dans liste d'événements** `all confirmations` or `only negative confirmations` was selected.

CONFIGURING FUNCTIONS

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 for Runtime

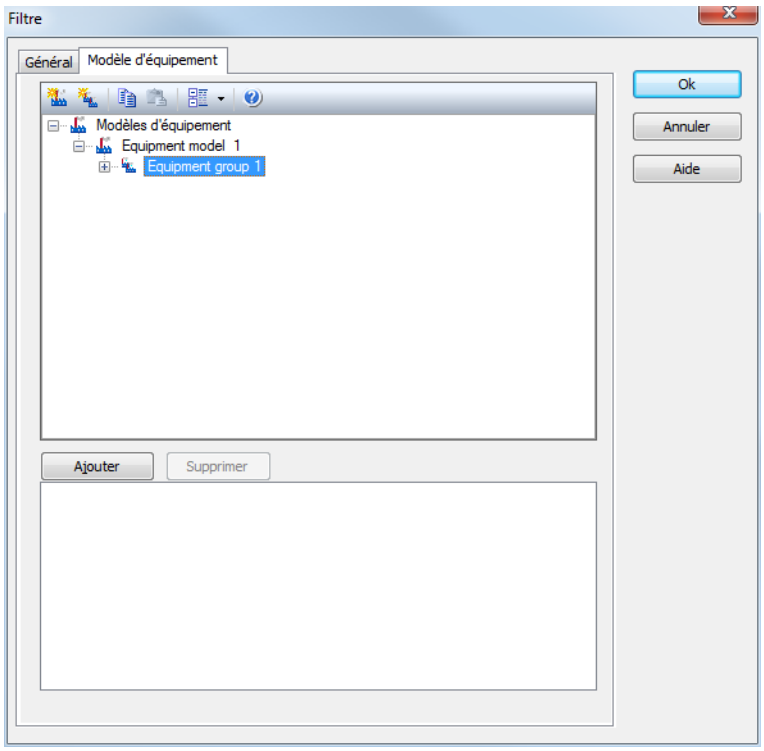
General



Parameters	Description
Alarm/event groups	List of configured alarm/event groups.
Alarm areas	List of configured alarm areas.
Alarm/event classes	List of configured alarm/event classes.
Activate	`Active`: When activating the function the selected elements are added to the list of suppressed elements.
Deactivate	`Active`: When activating the function the selected elements are removed from the list of suppressed elements.
Display this dialog in Runtime	`Active` : Lors de l'appel de la fonction dans le Runtime, cette boîte de dialogue est ouverte et l'utilisateur peut ajuster la configuration avant son exécution. La boîte de dialogue est affichée sur l'ordinateur actuel dans le Runtime. Dans le cadre de l'utilisation du réseau, lors de l'activation du client, la boîte de dialogue est également affichée sur le client
OK	Accepts settings in all tabs and closes dialog.
Cancel	Discards settings in all tabs and closes dialog.
Help	Opens online help.

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

Equipment Modeling



Propriété	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.
Ajouter	Ajoute les groupes sélectionnés à la liste de filtres.
Supprimer	Supprime tous les groupes sélectionnés de la liste de filtres.
Liste de filtres	Affiche tous les groupes d'équipements devant être filtrés.
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	Ouvre l'aide en ligne.

AJOUTER DES GROUPES

- 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.
- Sélectionnez un groupe ou un niveau d'équipements.
- Ajoutez le nouveau groupe à la liste affichée dans la partie inférieure de la boîte de dialogue en cliquant sur le bouton **Ajouter**.
- Les sous-groupes ne sont pas ajoutés automatiquement.
- Vous pouvez sélectionner autant de groupes que vous le souhaitez.

SUPPRIMER LES GROUPES

- Sélectionnez les éléments souhaités dans la liste affichée dans la partie inférieure de la boîte de dialogue (les sélections multiples sont autorisées).
- 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 Runtime. To do so, link this function with:

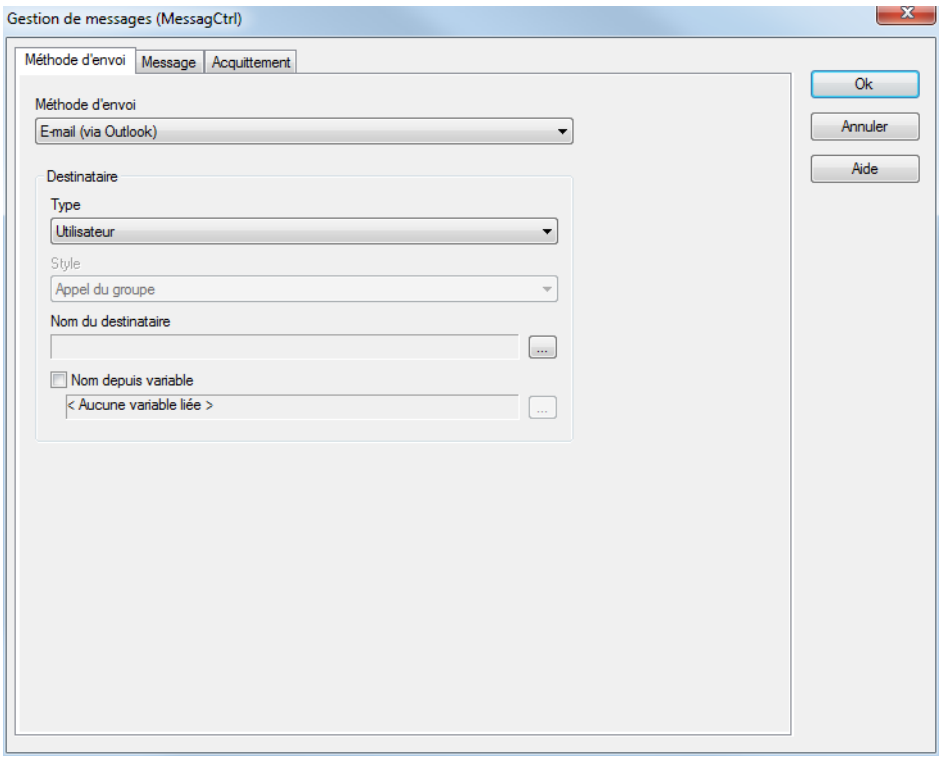
- an alarm (function for group, class or area)
- a limit value (property **Valeurs limite/Fonction**),
- a response matrix (function)
- a time control (via button or PFS)

> **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 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:



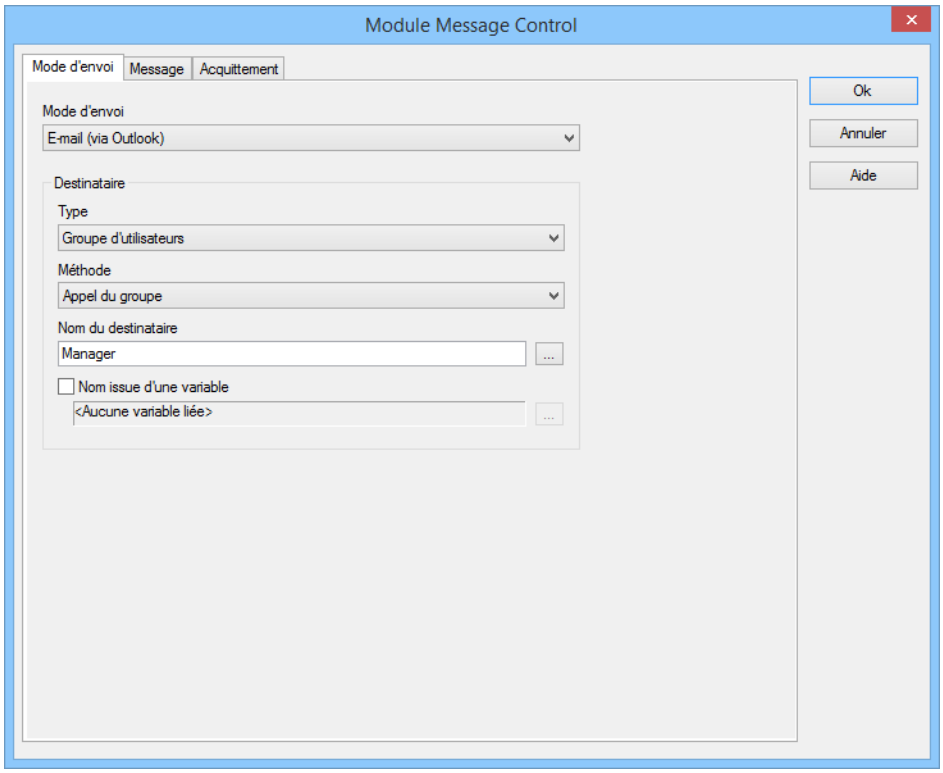
4. Configure
   - Sending mode (à la page 66)
   - Message (à la page 68)
   - Acknowledgment of receipt (à la page 81)
5. Close the dialog by clicking on **OK**.
   Note: The entries are validated in the tabs by clicking on **OK**. If incorrect configurations are discovered, the user is informed by 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 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.

> **Informations**
>
> In Runtime the function **Envoi d'un message** is always performed at the computer executing the process.

Sending mode

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



Parameters	Description
Sending mode	Selection of the sending mode from the drop-down list: The following are available: ▸ 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 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 16) of the workspace are shown.
Recipient	Recipient settings:
Type	Selection of recipient pattern from the drop-down list. Possible selection: ▸ User group: Sending to a group. Selection of a group in case of switching the option or via option **recipient name**. The sending mode for the group is defined in the option **Method**. ▸ User: Sending to an individual user (and the substitute person defined in their profile). Selection of a user in case of switching the option or via option **recipient name**.
Method	Only available if the option **Type** `user group` was selected. Selection of sending method from drop-down list. Possible methods: ▸ `Group call`: All users of the group will receive the message parallel. (Notice: in previous versions to zenon 7.00 this method corresponded the sending type **Group sending**.) ▸ `Successive call`: The message is delivered to the users according to the defined sequence (à la page 49) group successively until the first user positively acknowledges receipt. The group is only run through once. If there is no positive acknowledgement the sending is ended. ▸ `Infinite successive call`: In principle corresponds to the `Successive call`, however, includes a repetition: After the last recipient the group is being run through again beginning on the top. The message is being sent until it is positively acknowledged by the first recipient. 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 out through 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	Ouvre l'aide en ligne.

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 66).

Messages are individually configured:

- E-mail via Outlook and SMS via GSM or SMS gateway (à la page 68)
- E-mail message via SMTP (à la page 72)
- Voice message by audio file (à la page 76)
- Voice message via text to speech (à la page 79)

You can find details on configuration of the parameters for messages in the **Parameters for messages** (à la page 84) chapter.

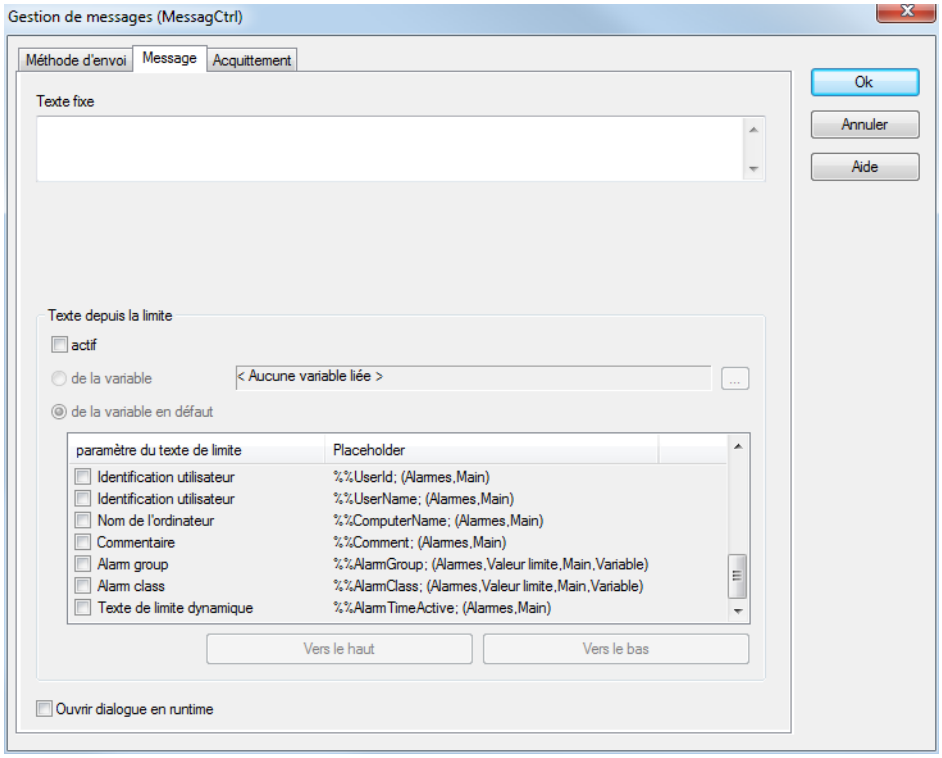
Informations

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

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

Configuration of the message text for:

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



Parameters	Description
Free message text	Integral part of the message. This text is searched for placeholders: ▸ **$**: indicates "formatted" text. The text is subsequently parsed for parameters. ▸ Text components are considered parameters if they are separated from the rest of the text with a semicolon (**;)** and end with a semicolon (**;)**. ▸ A Parameter (à la page 84) 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 86) of **limit value text parameters**. You can read more about the subject in the **Parameters for meesages** (à la page 84) chapter.
Text from limit value	Properties for additional parameters from limit values.
active	`Active`: Adds additional parameters from the limit value to the constant text. Configuration via options: ▸ **from variable** ▸ **from penultimate variable** ▸ **list limit value text parameters** Attention: If this option is active only limit value violations can trigger this function.
from variable	Parameters are taken over from the defined variable. Click on button **...** in order to open the dialog for selecting a variable.
from penultimate variable	Parameters are taken over from the variable whose limit value was violated.
list limit value text parameters	Selecting parameters from list by activating the corresponding checkbox. Sequence is definded by drag & drop with the mouse or the buttons **move entry up** and **move entry down**. Split placeholders: Displays placeholders which can be used in free message text. If the option **from variable** or **from penultimate variable** or the variable selection is changed the example entry will be adjusted. Placeholders can be copied. Structure: **%Variable%Parameter; (origin)** ▸ **%**: percentage sign as prefix for variables and parameters ▸ **Variable**: Variable. If no variable is indicated the following parameter refers to the main variable.

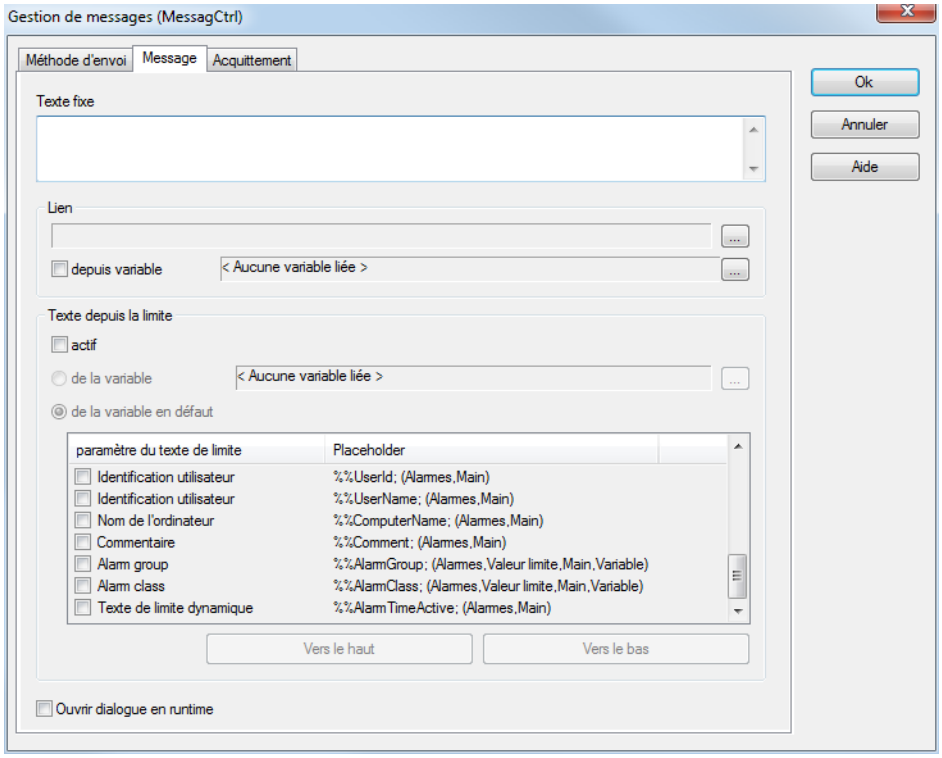
	▸ **Parameter**: Parameters ▸ ;: semicolon ends string. ▸ Origin: Only information, not used for free text. Indicates which variables can be used for the parameter : main variable via limit value main variable via AML additional variable via limit value or AML For details, see the **Parameters for messages** (à la page 84) and Parameter limit value text (à la page 86) chapters. If invalid parameters are used a corresponding error message is displayed in the message: **Invalid Parameter: 'Parameter'>.**
Move selected entry up	Places selected element one step higher.
Move selected entry down	Places selected element one step lower.
Show this dialog in the Runtime	`Active`: Opens this dialog in Runtime before executing the function. Attention - Behavior in Runtime: ▸ In Runtime this function is always performed at the computer executing the process. If the function is started on a client the dialog is displayed on the computer executing the process. ▸ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out. ▸ If the dialog is edited and closed with OK the changes will be saved and the function carried out. ▸ If the dialog is closed with **cancel** no message is being sent. This process is documented in the CEL provided for the property **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	Ouvre l'aide en ligne.

E-mail message via SMTP

E-MAIL MESSAGE VIA SMTP

Configuration of the message text for:

- E-mail message via SMTP



Parameters	Description
Free message text	Integral part of the message. This text is searched for placeholders: ▸ **$**: indicates "formatted" text. The text is subsequently parsed for parameters. ▸ Text components are considered parameters if they are separated from the rest of the text with a semicolon **(;)** and end with a semicolon **(;)**. ▸ A Parameter (à la page 84) 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 86) of **limit value text parameters**. You can read more about the subject in the **Parameters for meesages** (à la page 84) chapter.
Appendix	Configuration of an appendix to the e-mail. Screenshots can be sent as appendix. Click on button **...** in order to open the dialog for selecting a file. This has to be deposited in the node Files/Graphics. When sending the message the selected file is added as attachment to the e-mail. Note: To send files from other folders with desired folder information, select the **from variable** option.
from variable	`Active`: Select a string variable. Click on button **...** in order to open the dialog for selecting a variable. The value of the string variables is read in Runtime and the content is analyzed. The path to the appendix is set according to the analysis: ▸ Absolute path: this is used without changes ▸ No absolute path: the value of the string variables is attached to the path to the zenon graphics folder When setting the path, a check is not carried out to see which expansion the file has. Thus in doing so, NO CHECK IS CARRIED OUT TO SEE WHICH EXPANSION HAS THE FILE. Variables can be sent desired file formats from the attachment path, in particular PDFs and Office documents from any desired path.
Text from limit value	Properties for additional parameters from limit values.
active	`Active`: Adds additional parameters from the limit value to the constant text. Configuration via options: ▸ **from variable**

	▸ **from penultimate variable** ▸ **list limit value text parameters** Attention: If this option is active only limit value violations can trigger this function.

from variable	Parameters are taken over from the defined variable. Click on button **...** in order to open the dialog for selecting a variable.
from penultimate variable	Parameters are taken over from the variable whose limit value was violated.
list limit value text parameters	Selecting parameters from list by activating the corresponding checkbox. Sequence is defined by drag & drop with the mouse or the buttons **move entry up** and **move entry down**. Split placeholders: Displays placeholders which can be used in free message text. If the option **from variable** or **from penultimate variable** or the variable selection is changed the example entry will be adjusted. Placeholders can be copied. Structure: **%Variable%Parameter; (origin)** ▸ **%**: percentage sign as prefix for variables and parameters ▸ **Variable**: Variable. If no variable is indicated the following parameter refers to the main variable. ▸ **Parameter**: Parameters ▸ ;: semicolon ends string. ▸ Origin: Only information, not used for free text. Indicates which variables can be used for the parameter : main variable via limit value main variable via AML additional variable via limit value or AML For details, see the **Parameters for messages** (à la page 84) and Parameter limit value text (à la page 86) chapters. If invalid parameters are used a corresponding error message is displayed in the message: **Invalid Parameter: 'Parameter'>.**
Move selected entry up	Places selected element one step higher.
Move selected entry down	Places selected element one step lower.

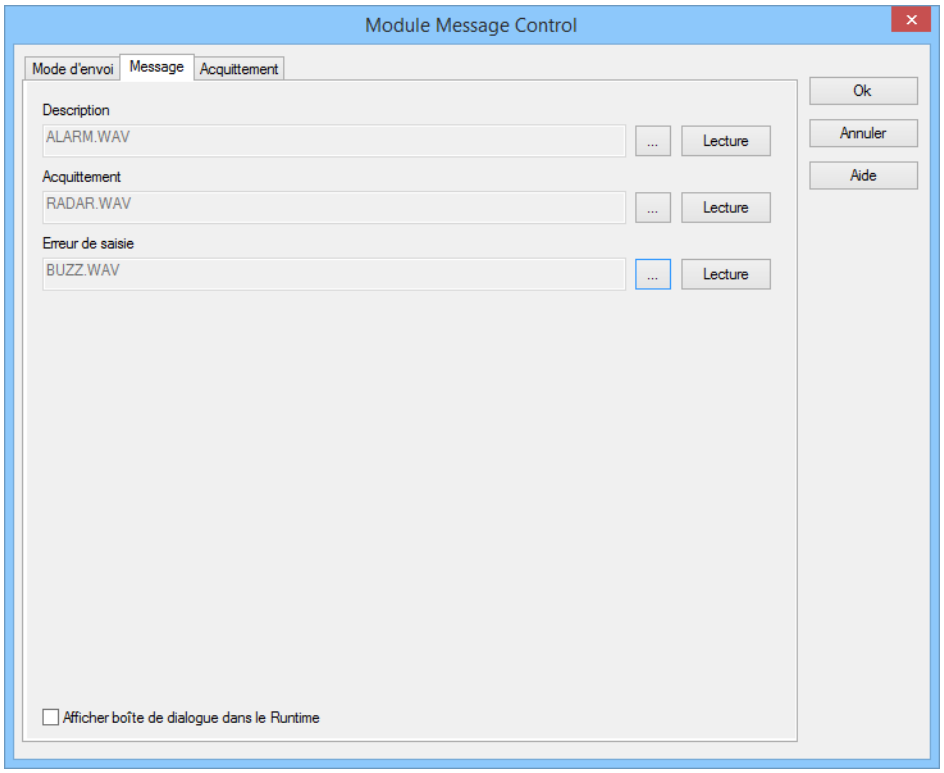
Show this dialog in the Runtime	Active: Opens this dialog in Runtime before executing the function. Attention - Behavior in Runtime: ▸ In Runtime this function is always performed at the computer executing the process. If the function is started on a client the dialog is displayed on the computer executing the process. ▸ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out. ▸ If the dialog is edited and closed with OK the changes will be saved and the function carried out. ▸ If the dialog is closed with **cancel** no message is being sent. This process is documented in the CEL provided for the property entries into CEL `all confirmations` or `only negative confirmations` was selected.
OK	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	Ouvre l'aide en ligne.

Voice message by audio file

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

- Modem
  or

- VoIP


Module Message Control
Mode d'envoi
Message
Acquittement
Description
ALARM.WAV
...
Lecture
Acquittement
RADAR.WAV
...
Lecture
Erreur de saisie
BUZZ.WAV
...
Lecture
Ok
Annuler
Aide
Afficher boîte de dialogue dans le Runtime

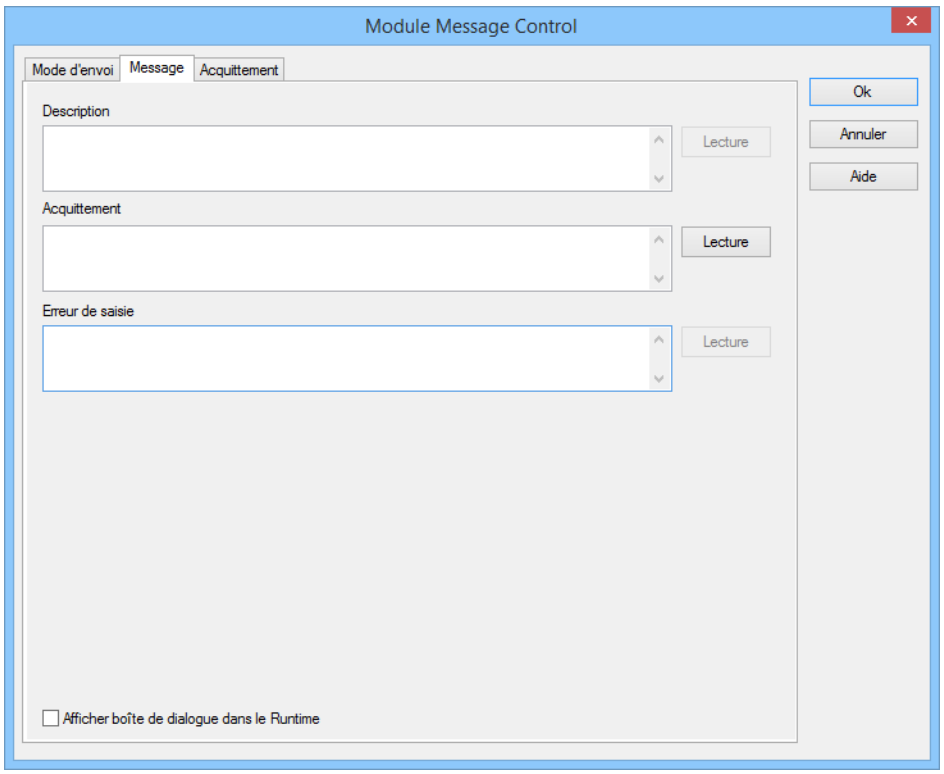
Parameters	Description
Description	Select the file that contains the description of the message. Click on button **...** in order to open the dialog for selecting a file. This has to be deposited in the node File/Multimedia. The file can be played for testing with the button **Play**.
Acknowledgement	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 Runtime	`Active`: Opens this dialog in Runtime before executing the function. Attention - Behavior in Runtime: ▸ In Runtime this function is always performed at the computer executing the process. If the function is started on a client the dialog is displayed on the computer executing the process. ▸ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out. ▸ If the dialog is edited and closed with OK the changes will be saved and the function carried out. ▸ If the dialog is closed with **cancel** no message is being sent. This process is documented in the CEL provided for the property entries into CEL `all confirmations` or `only negative confirmations` was selected.
OK	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	Ouvre l'aide en ligne.

Voice message via text to speech

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

- Modem
  or
- VoIP



Parameters	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 84). 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 84). 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 dynamic texts (à la page 84). The text can be played for testing with the button **Play**.
Show this dialog in the Runtime	▸ `Active`: Opens this dialog in Runtime before executing the function. Attention: Behavior in Runtime Note the following when calling up this dialog in Runtime: ▸ In Runtime this function is always performed at the computer executing the process. If the function is started on a client the dialog is displayed on the computer executing the process. ▸ If no entry is made the dialog is automatically closed after 30 seconds and the function is carried out. ▸ If the dialog is edited and closed with OK the changes will be saved and the function carried out. ▸ If the dialog is closed with **cancel** no message is being sent. This process is documented in the CEL if `all confirmations` or `only negative confirmations` was selected for the **Enregistrer dans liste d'événements** property (**Paramètres spécifiques au projet**group).

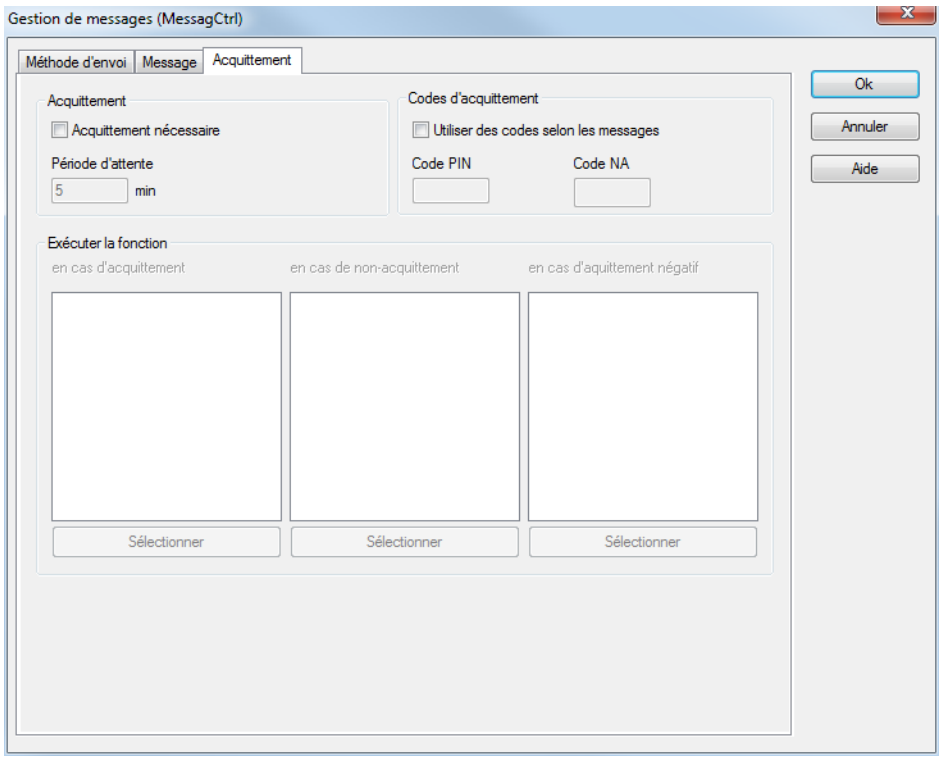
OK	Applique toutes les modifications effectuées sur tous les onglets, puis ferme la boîte de dialogue. Note: The entries are validated in the tabs by clicking on **OK**. If incorrect configurations are discovered, the user is informed by means of a message box. The user can then change the configuration 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	Ouvre l'aide en ligne.

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 79) and voice message via audio file (à la page 76)) can only be confirmed synchronously via the respective transmission medium. The use of other transmission methods for communication (such as email or SMS) is not possible. The message must be acknowledged immediately via the same connection which is still open.*

ACKNOWLEDGMENT SETTINGS



Parameters	Description
Acknowledgment of receipt	Settings for acknowledgment of receipt.
Acknowledgement 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.
Confirmation codes	Settings for acknowledgment 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 following options **PIN code** and **NA code** are used for confirmation.
PIN code	Individual code for confirming the message.
NA code	Individual code for rejecting the message.
Function execution	Selection of functions to be executed after a message has been confirmed, rejected or not acknowledged at all.
In case of confirmation	Functions which are executed if the message is confirmed. A click on the **Select** button opens the function selection dialog.
In case of lack of confirmation	Functions which are executed if there is no response to the message. A click on the **Select** button opens the function selection dialog.
In case of negative confirmation	Functions which are executed if the receipt of message is rejected. A click on the **Select** button opens the function selection dialog.
OK	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	Ouvre l'aide en ligne.

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 68)
- **E-mail message via SMTP** (à la page 72)
- **Voice message via text-to-speech** (à la page 79) via modem or VoIP

- **Partie dynamique du sujet** property:

CREATION OF A MESSAGE

- Text parts are separated from one another with a semi colon (**;**).
- **$**: marks text that contains the parameter.
- A parameter (à la page 84) can:
  - Be a key from the language table, such as **@MyText;**
  - Relate to a variable, when executing the function via AML or a limit value violation such as **%Var1;**
  - Be a compiled entry in the language table: for example **%@Var2+MultipleText**
- @ marks language switching
- **%** marks variables
- **%%** marks **limit value text parameters** (à la page 86) for variables

  A variable can be stated between the two percentage marks.
  If no variable is given, the value relates to the main variable.
  - Messages end with a semicolon (**;**).

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

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

LIST OF PARAMETER TEXT

Parameters	In the Runtime	Action
key: **@MyText**	A check is made to see if an entry in @MyText is present in the language table for the current language.	`Existing`: entry is added to the message. `Not existing`: According to the settings of the project properties **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. ▸ `Not existing`: 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 Text from limit value and free text

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

CONFIGURATION OF TEXT FROM LIMIT VALUE

To link limit values:

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

CONFIGURATION OF FREE MESSAGE TEXT

STRUCTURE

$FREETEXT;%VARIABLE%TEXT;

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

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

  If no variable is indicated the following parameter refers to the main variable.
- End of the character sequence: Semicolons (**;**).

EXAMPLE

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

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

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

LIST LIMIT VALUE TEXT PARAMETERS

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

confirmation)	receipt required (à la page 81) was activated.	receipt required (à la page 81) was activated.	receipt required (à la page 81) was activated.
%%Address (Address)	Address.	Address.	Address.
%%Unit Measuring unit:	Technical unit according to **Unité de mesure** property.	Technical unit according to **Unité de mesure** property.	Technical unit according to **Unité de mesure** property.
%%AlarmArea (Alarm area)	Alarm area of the main variable.	Alarm area of the main variable.	Alarm area of the additional variable.
%%LimitTimeAcknowledged (Time alarm is acknowledged)	--	Time stamp of the alarm object.	--
%%UserId (User identification)	--	User identification of the user who started the action. Conforms to property **Identifiant utilisateur**.	--
%%UserName (User name)	--	Complete name of the user who started the action. Conforms to property **Nom complet**.	--
%%ComputerName (Computer name)	--	Name of the computer on which the action was started.	--
%%Comment (Comment)	--	Comment of the alarm object.	--
%%AlarmGroup (Alarm/event group)	Alarm/event group of the alarm object (as text).	Alarm/event group of the alarm object (as text).	Alarm/event group of the alarm object (as text).
%%AlarmClass (Alarm/event class)	Alarm/event class of the alarm object (as text).	Alarm/event class of the alarm object (as text).	Alarm/event class of the alarm object (as text).
%%AlarmTimeActive (Time alarm is pending)	--	Period of time the alarm has been pending.	--

ERROR TREATMENT

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

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

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

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

8. Message Control in Runtime

In 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 84).

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

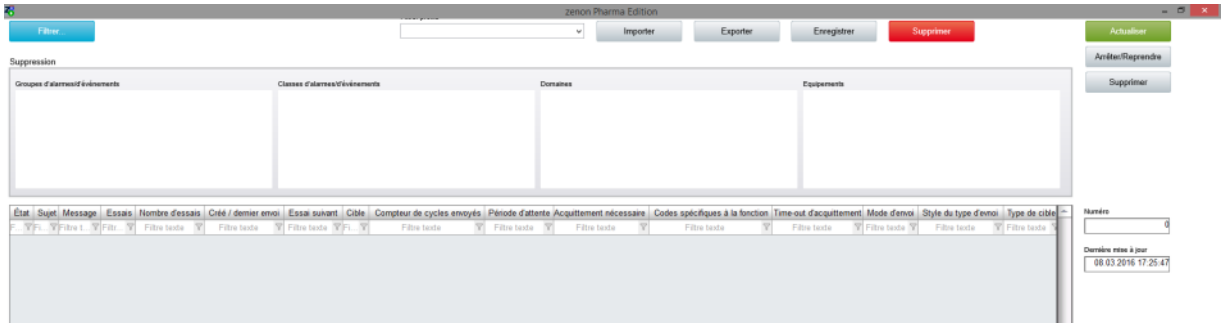
- a message configured for this is being sent
- its acknowledgement (à la page 94) is being analyzed
- the functions used for acknowledgement are executed

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

MANAGEMENT IN RUNTIME

The Message Queue can be managed with a screen of type Message Control (à la page 92).

Message Control is being closed when closing the Runtime or by the function Send Message: deactivate (à la page 84).



TIME STAMP IN ALARM MESSAGE LIST AND MESSAGE CONTROL

When the function Send message (à la page 64) is activated a copy of the alarm object is transmitted to the function. The message text is created and translated when the function is activated. If the function is activated via AML it will use a copy of the alarm object from the AML. So the content always corresponds to a current combination of process and alarm status.

MESSAGE QUEUE REMANENCE

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

- Current messages in the queue
- Suppressed alarm groups
- Suppressed alarm class
- Suppressed areas
- Suppressed equipment

NETWORK

You can find further information on the behavior in the network in the Network (à la page 99)chapter.

ACKNOWLEDGING

Every message can unambiguously be identified by a GUID. According to the configured media and sending mode messages can also be acknowledged through other media. For more information about the acknowledgement of messages see chapter Message acknowledgement (à la page 94).

LANGUAGE SWITCH

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

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

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

8.1 Connect screen of the type Message Control

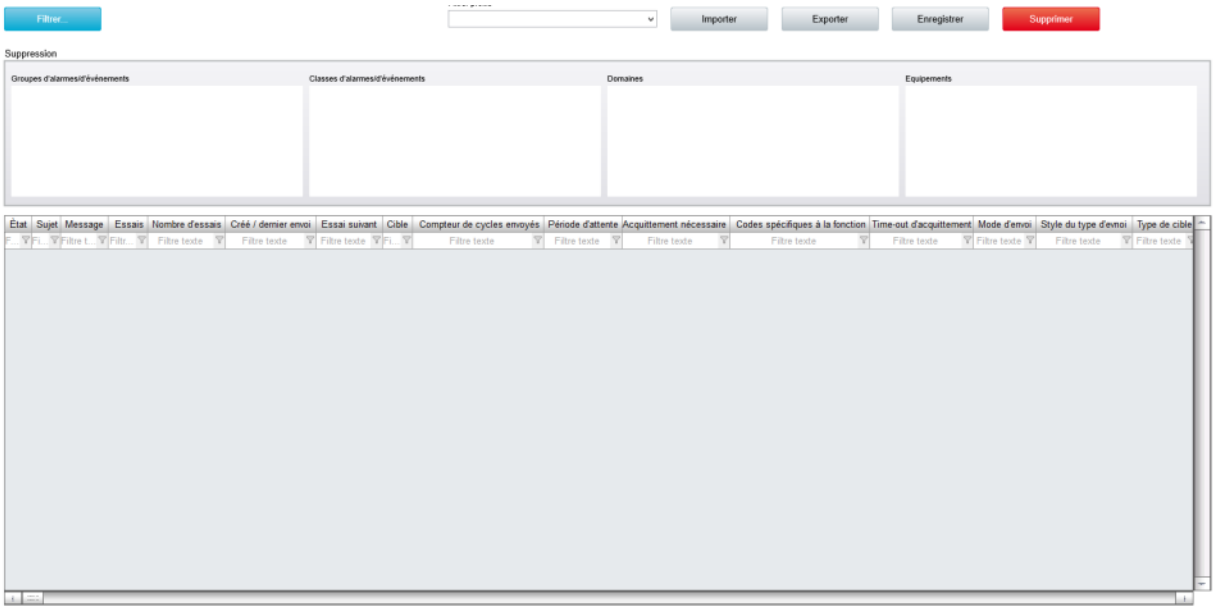
The screen of type `Message Control` makes a copy of the current message queue and control elements for analysis and filtering available in Runtime. The content of the elements displayed in the **message list** list field corresponds to the point of time when the screen is connected and is not automatically 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 which

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

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

The columns of the **message list** can be configured individually (see Filter (à la page 54)) and can be provided with individual column headings. These labelings are localizable.



Control elements	Description
Insert template	Ouvre la boîte de dialogue de sélection de modèle pour le type de synoptique correspondant. Des modèles sont fournis avec zenon, et peuvent également être créés par l'utilisateur. Modèle ajoute des éléments de contrôle prédéfinis à des emplacements prédéfinis sur le synoptique. Les éléments qui ne sont pas nécessaires peuvent également être supprimés individuellement après leur création. Les éléments supplémentaires peuvent être sélectionnés dans la liste déroulante, puis insérés dans le synoptique. Les éléments peuvent être déplacés et disposés individuellement sur le synoptique.
Buttons	Buttons for Runtime actions.
Refresh	Refreshes view. The current data are loaded onto the screen from the message queue. The point of time of the last update can be displayed with the field **Last update**.
Delete	Deletes selected entry from the list. It is only possible to delete messages which have not been sent yet.
Filter	Opens Dialog (à la page 54) to configure the filters for the view.
Stop/continue	Starts and stops the cyclic update of the list.
Window	Lists and fields which can be displayed in Runtime.
Number of messages	Number of currently existing messages.
Last Update	Point of time of last update.
Suppressed alarm/event groups	**Alarm/event groups** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Suppressed alarm/event classes	**Alarm/event classes** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Suppressed alarm/event areas	**Alarm/event areas** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Suppressed equipment	**Equipment** for which messages were suppressed and the alarms of which thus are not displayed in the message queue.
Message queue	List of messages. Contains messages which have been sent, confirmed or deleted or remain to be sent. The messages are only displayed and cannot be edited anymore. Column headings can be named individually (à la page 56) and are localizable by putting @ before them.

Filter profiles	Profile administration
Profile selection	Opens the dialog for selecting a profile.
Save	Saves current setting as a profile.
Delete	Deletes profile.
Import	Opens dialog for importing profiles from a file.
Export	Opens dialog for exporting profiles from a file.

8.2 Acknowledgement of messages

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

Medium	To acknowledge - **-**: no - **+**: yes
E-mail message via Outlook (à la page 17)	**-**
E-mail message via SMTP (à la page 18)	**-**
SMS message via GSM modem (à la page 20)	**-**
SMS message via SMS gateway (à la page 23)	**-**
Voice message by audio file (à la page 26)	**+**
Voice message via text to speech (à la page 27)	**+**

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

Informations

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.

CONFIGURING THE CONFIRMATION

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

1. In the function Send Message (à la page 64) open the tab Acknowledgement of receipt (à la page 81).
2. Activate the checkbox **Acknowledgement required** (always active for voice messages).
3. Define the waiting period (Timeout) within which the message must be acknowledged.
   After the waiting period has expired the message will be sent to the next person on the list.
4. Configure the codes for confirmation (PIN) or rejection (NA code) as follows:
   a) properties of the user:
   Utilisateur Message Control -> **Code PIN** and **Code NA**
   or
   b) in the function via:
   checkbox **Use message-specific codes** -> **PIN code** and **NA code**

   If codes are configured both via properties and the function the codes of the function are valid.
   Hint: If codes are configured via the function it is recommended to make a note in Message text (à la page 68) for the recipient to use the right code for activation or rejection.
5. Define the functions which are to be executed after receiving the code for:
   a) message was confirmed
   b) message was neither confirmed nor rejected
   c) message was rejected
6. Close the dialog by clicking on the **OK** button.

ACKNOWLEDGEMENT IN RUNTIME

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

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

Thus it is also possible to acknowledge written messages through other sending modes than those configured. The following must be the case for this:

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

E-ACKNOWLEDGING E-MAILS

An e-mail is acknowledged as follows:

- The response is being sent using the reply function of the e-mail program of the recipient.
- The original subject must not be changed in the email. This means: The contents defined in Outlook in **Sujet (ID)** or for SMTP in **Sujet pour les e-mails sortants** must remain unchanged. These will identify and allocate the message during receipt in zenon.
- However the subject must have at least three characters placed in front of it as a response. For example **RE:** or **AW:**.
- The response text must contain:
  - GUID: At the start of the message. There must be no other character, including spaces, in front.
  - Semi-colon (**;**): Separator.
  - Text: Contains PIN code or NA code.

  for example: **43d3c61d-ccc9-4c76-bc2c-61c2d12b0db3;0246**
- In Message Control e-mails for Message Control are identified by means of the subject. Attention: The response subject must be marked as response. To do this, the message is checked on receipt to see if it starts with at least three characters such as **RE:** or **AW:**.
- Message Control analyzes GUID and code and subsequently executes the functions set in the function Send Message (à la page 64).

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

ACKNOWLEDGING SMS

An SMS is acknowledged as follows:

- The response is being sent using the reply function of the user's telephone.
- The response text must contain:
  - GUID: At the start of the message. There must be no other character, including spaces, in front.
  - Semi-colon (**;**): Separator.
  - Text: Contains PIN code or NA code.

  for example: **43d3c61d-ccc9-4c76-bc2c-61c2d12b0db3;0246**
- Message Control analyzes GUID and code and subsequently executes the functions set in the function Send Message (à la page 64).

> **Informations**
>
> *SMSs to Smartphones can also be received and acknowledged with the **Notifier App by zenon**. To do this, the App sends the response as an SMS in **Alarm-ID;PIN** or **Alarm-ID;NA** format.*
>
> if the medium e-mail was configured a message sent via SMS can also be acknowledged via e-mail.

VOICE MESSAGE

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

- The telephone of the recipient must support DTMF.
- The receiving modem of the equipment must support DTMF.
- The message is read to the recipient with the text of the **description**
- The recipient acknowledges the message via key signals with **PIN code** or **NA code**.
- The recipient can**:**
  - have messages repeated:
    by pressing the #key on the telephone**.** The previously-sent message (description, confirmation or error message) is repeated.
  - re-set sent code:
    by pressing the *key on the telephone**.** The code (PIN or NA) previously sent by the recipient is revoked. The playback of the message starts with the **description** again.
- Message Control analyzes the code and subsequently executes the functions set in the function Send Message (à la page 64).

Note: Voice messages always must be acknowledged by telephone.

8.3 Voice messages in Runtime

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

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

   Note: In the list of messages, the name of the audio file is displayed in the **Message** column; with sending by audio file, the message text is displayed with Text-to-Speech.
3. Connect 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.
4. 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.
5. 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.
6. 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.
7. 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**
8. 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 59) is being synchronized and remanently saved.

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

USERS

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

NO RESOURCE SHARING

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

- E-mail (SMTP): The server and standby server use different email accounts.
- E-mail (Outlook): The server and standby server use different Outlook profiles.
- GSM: The server and standby server use different modems and SIM cards with different telephone numbers.
- SMS (SMS gateway): The Server and Standby Server use different 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

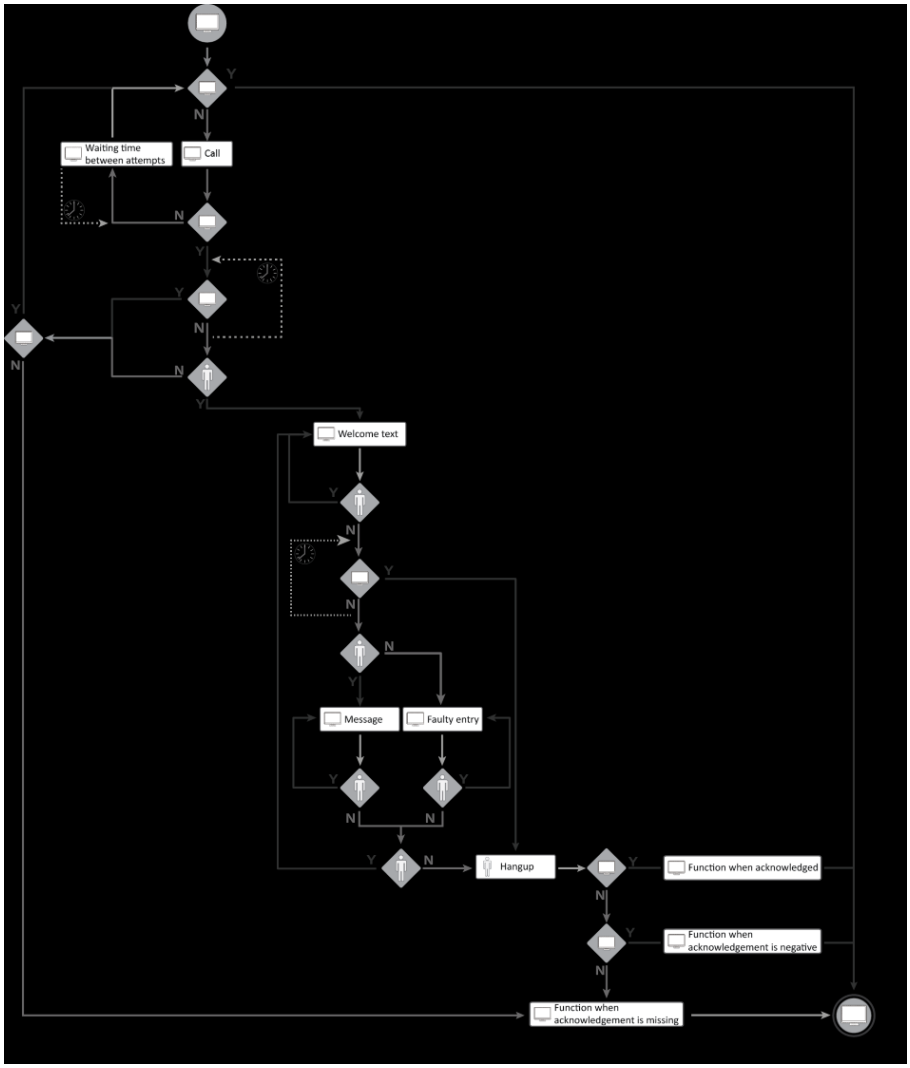
BEHAVIOR IN CASE OF FAILURE

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?
   - **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.
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** Further with **Confirmation** process.
   - **N**: Further with **incorrect entry** process.
5. Validation: Received code *?
   - **Y**: Further with **text description** process.
   - **N**: Further with **End call** process.

End call

1. Validation: Has **PIN** been received?
   - **Y**: Execute linked function for **PIN**, if present.
   - **N**: Further to **NA** process.
2. Validation: Has **NA** been received?
   - **Y**: Execute linked function for **NA**, if present.
   - **N**: Further 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.
    Further with check: Has the defined value for the **Nombre maximum d'essais** function been reached?
  - **N**: Further 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 Diagnose Viewer in a separate module **[Message Control]**. Message details can be found in section LOG entries (à la page 108).

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 12).

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 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 modem requires a PIN / is the PIN incorrect? (PIN error messages in LOG) ▸ Too many attempts to log on with an incorrect PIN? (PUK error message in LOG) In this case, the connection between zenon and the modem is only possible again once the PUK has been entered manually and thus the lock been lifted.	▸ Check COM port and modem configuration. ▸ Check PIN, if necessary release manually with the PUK.
Connection is established, but sending and receiving do not work.	▸ Has a connection first been established between the modem and the GSM network? In most cases, the connection with the network is only established after accessing the manufacturer software! ▸ Despite this the modem is not connected with the network (check with	▸ 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?

	manufacturer software): Look for a location with better reception. ▸ The SMSC telephone number that has been entered is not correct. Before entry in zenon or the first time a connection is established, check, also using the **Vérification connexion** property. 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 diagnose viewer.

- Miscellaneous messages (à la page 108)
- E-mail via SMTP/POP (à la page 116)
- Voice over IP (à la page 120)
- SMS (à la page 122)
- OpenSSL Library (à la page 125)

9.2.1 Miscellaneous messages

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

- Debug (à la page 109)
- Deep Debug (à la page 111)
- MSG (à la page 115)
- Warnings (à la page 116)

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 cancelled.
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: 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	A call is active, it is closed now.

	terminated 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 Benutzer (%s) for the group (%s) will be notified, previous Benutzer was:%s	The next user [name] from user group [name] is notified, the previous user was user [name].
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.

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 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 will be updated!	The project received a MessageControl-StateChange telegram for 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.
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 is 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.
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.

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.

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 119)
- Warning (à la page 120)

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 cancelled.
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: 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	A call is active, it is closed now.

	terminated 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 Benutzer (%s) for the group (%s) will be notified, previous Benutzer was:%s	The next user [name] from user group [name] is notified, the previous user was user [name].
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.

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 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 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:	The SMTP server has not confirmed the AUTH PLAIN data with OK. The login has failed. (The response is

	[response from the SMTP server]	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.)

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!
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.
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] from GSM 7-Bit-Alphabet. Baselanguage: [byte as HEX], Extensionlanguage [byte as HEX]	A character contained in the 7-bit-SMS data section could not be decoded. The employed alphabets basis table and extension table are attached as language identifiers (see Standard 3GPP TS 23.038 V10.0.0 (2011-03).)
ERROR	Insufficient data to decode 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 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 server]	The SMTP server has not confirmed the AUTH PLAIN data with OK. The login has failed. The response is attached to the LOG message.
ERROR	SMTP AUTH should be used, but the Server neither supports AUTH LOGIN nor the minimum implementation AUTH PLAIN	SMTP authentication should be used, however, during the EHLO response the server has not listed the required commands in the list of supported extension commands. The following commands are required: AUTH LOGIN and AUTH PLAIN, the latter is the minimum implementation according to RFC 2554.
ERROR	[SMTP / POP3] not connected	The attempt was made to send data to the outgoing mail server or to retrieve data from the incoming mail

		server, although the server is not connected.
ERROR	The Server did not send an OK-Response to the MAIL command: [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.
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.