# GTXT PC Software for SMS Messages User's Manual

Manual Part Number 96-00594



**Zed-3** 501 Valley Way Milpitas CA 95035 USA

+1-408-587-9333 http://www.zed-3.com

## Notice

The information contained in this document is subject to change without notice.

**Zed-3** makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Zed-3 shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Zed-3 assumes no responsibility for the use or reliability of interconnected equipment that is not furnished by Zed-3.

This document contains proprietary information which is protected by copyright. The contents of this manual and the associated software are the property of Zed-3, and all rights are reserved. No part of this document may be photocopied, reproduced, stored in any computer format, translated to another language, or publicly displayed without the prior written consent of Zed-3.

The information contained herein has been prepared by Zed-3 solely for use by Zed-3'ss employees, agents, and customers. Dissemination or use of the information or concepts contained herein to or by other parties is prohibited without prior written consent from Zed-3.

Zed-3, the Zed-3 logo, the Zed-3 mark, CN2x2, CN2x2P, CN2x4, CU3, GA90, GS8, GTXT, MDS, SE30P, SE30+, SE30X, SE150, SE500X, SEME, and SP3 are trademarks of Zed-3 and may be registered trademarks in certain countries. All other names may be trademarks or registered trademarks of their respective owners.

Release	Release Date	
1.2	01 October 2008	
1.3	5 November 2008	
1.4	29 January 2009	

## **Revision History**

## Contents

Chapter 1	Installation51.1Scope.51.2Basic Functionality61.3Installation61.4Running the program61.5The Main GTXT Window81.6About our Documentation9
Chapter 2	Configuration       11         2.1       Introduction       11         2.2       Configuring the Regional Options       11         2.3       Configuring the Connection to the GS8       12         2.4       Connecting to the GS8       13         2.5       Licensing       14
Chapter 3	Selecting the Recipient153.1Introduction153.2Editing a Local File of Names and Numbers153.3Filtering and Displaying Information for the Contacts163.4Choosing Recipients of the SMS173.5Hiding and Showing the Contacts Panel183.6File Locations18
Chapter 4	Messages and History       19         4.1       Introduction       19         4.2       Sending a Message       19         4.3       Receiving Messages       20         4.4       Using Tabs       21         4.5       History       22
Chapter 5	Scheduling a Message.235.1Introduction.235.2Configuring the Schedule.235.3Editing a Scheduled Message.245.4Deleting a Scheduled Message.24
Chapter 6	Linking to a Database256.1Introduction256.2Specifying the Database and Fields256.3Creating a Linked Message30

	6.4 6.5	Editing a Linked Message	. 33 . 33
Chapter 7	<b>Softw</b> 7.1 7.2 7.3 7.4	<b>vare API</b> About the API Notifying Message Data Structure Function Specification	<b>. 35</b> . 35 . 35 . 36 . 37
Appendix A	Enco	oding for SMS Messages	. 45
	A.1 A.2 A.3 A.4 A.5	Introduction 7 Bit Characters Default Alphabet Unicode Characters (16 bits) Long Messages	. 45 . 45 . 47 . 48 . 48
Appendix B	Crea	ting an Operator User on the GS8	. 49
Appendix C	Acro	nyms	. 51
Appendix D	Glos	sary	. 57
	Index	κ	. 65

## Chapter 1

## Installation

## 1.1 Scope

#### 1.1.1 Purpose

This manual is intended for users of GTXT and administrators of the network that includes the GS8. It also has details of the GTXT API so that software developers who want to write their own programs to communicate with the GS8 can do so.

The manual assumes you have a GS8 modular gateway that is installed and functional.

#### 1.1.2 For Experienced Users

Experienced users may need to review only the screen shot and subsequent description in section 1.5 on page 8. Other users should carefully study this annual to ensure you obtain the maximum benefit from the GS8 and GTXT.

#### 1.1.3 Not Included

This manual does not describe the features and functionality of the GS8. It does not explain how to install the GS8 nor does it contain complete technology details of the product. This manual does, however, describe some technology aspects to allow you to interface with the GS8.

This manual does not include information on warranty, service, or support. Consult the Zed-3 web site for details on these or consult documentation that may have accompanied the product.

The manual does not provide pricing, names of sales representatives, or names of distribution channels.

Access the Zed-3 web sites for all further information. Zed-3 is very open about its products and most of the manuals are available on line at <u>http://www.Zed-3.com.</u>

## 1.2 Basic Functionality

The GTXT application works with the GS8 modular gateway to be able to send and receive SMS messages.<sup>1</sup> It is a graphical user interface that can be used to:

- send a message immediately
- send a message at a predetermined time (a schedule)
- send a message where the contents are extracted from a database and at a time that depends on a change of data in the database (a link)
- receive a message sent to you as a reply to an earlier message that you had sent
- receive all messages sent to the GS8 modular gateway

You can send and receive ASCII messages (160 characters per message) or Unicode messages (72 characters per message).<sup>2</sup>

GTXT operates with or without Microsoft Outlook. When you use GTXT without Outlook, you can send messages to contacts that are stored in a local file. You can easily add, edit, and delete those contacts and create different files for these contacts.

When you use GTXT with Outlook, you can send messages to contacts stored in your Outlook contact files as well as to contacts stored in your local files. GTXT saves incoming and outgoing messages in Outlook data files.

## 1.3 Installation

This application runs under Windows XP SP3 and Windows Vista SP1. It works in conjunction wit Outlook 2003 and Outlook 2007. It operates on PCs with a single or dual monitor.

Unzip the application and run **setup.exe**.

GTXT uses the Microsoft .NET Framework. This is Microsoft's programming model for building applications for a range of business processes. If you do not have the .NET Framework installed on your PC, the installation software will automatically install it. This will take about ten minutes on a fast PC and much longer on older PCs. Please be patient while this installs!

After the .NET Framework is installed (or if you already had it installed), the installation software will install the GTXT application. This proceeds quickly. If you have Microsoft Outlook running, the installation program asks you to close it.

## 1.4 Running the program

After the software is installed, you can run it from the Start menu or you can restart Outlook and run GTXT from within Outlook.

<sup>1.</sup> SMS messages are often referred to as text messages. The term *SMS* is used throughout this manual. The term *message* is used in this manual to describe a communication. Each SMS has a limited number of characters, but a message may have many more. Therefore a message may be sent as many SMSs.

<sup>2.</sup> Appendix A, starting on page 45, describes in detail the encoding used to send an SMS message and the total number of characters available in a specific message.

### 1.4.1 Running as a stand alone application

From the Windows main start menu, select:

Start | Programs | Zed-3 | GTXT SMS

You cannot run the application as a stand alone application if you have Outlook running. Close Outlook, wait until Windows has fully exited the application, and click on Retry.

If you have Outlook installed on your PC and you run GTXT as a stand alone application, GTXT warns you that you cannot use the contacts from Outlook when you run the application stand alone. Click Continue to proceed.

### 1.4.2 Running from within Outlook

With Outlook open, select:

File | New | Zed-3 SMS Message

or press:

Ctrl-Shift-Z

You cannot access GTXT if you started GTXT as a stand alone application and then subsequently started Outlook.

The first time you run the software after you have installed it, the Microsoft .NET framework registers the application. This may take 20 seconds. Please be patient – it does not take so long on subsequent starts!

#### 1.4.3 Hiding the Program

To move GTXT to the system tray, click on the X on the title bar. The application is still active but is no longer shown on the task bar. To restore GTXT, click on the Z icon in the system tray.

#### 1.4.4 Exiting the Program

To exit GTXT, click on the **Close** button. If you have started GTXT within Outlook, GTXT will also close when you close Outlook.

**Caution** If you have any scheduled SMS or linked SMS, GTXT cannot send these when it is not running. Therefore, you should not close the application if you want these sent.

## 1.5 The Main GTXT Window

The main window is shown below:



This is a brief description of the elements of the GTXT window. These are listed starting with the title bar and continuing clockwise. Click on the links below to view a description of the feature or function.

*Title bar.* This shows you whether the GTXT software is connected to the GS8 and if not, the status of that connection; see section 2.4 on page 13.

Inactive tab to an individual. Click to make this active; see section 4.4.1 on page 21.

Active tab. Messages you compose will be added to this tab; see section 4.4.1 on page 21.

*Close tab.* Removes the tab from the GTXT window but does not delete the history associated with the recipient or recipients; see section 4.4.3 on page 21.

*Inactive tab to a group.* This indicates that GTXT sent a message to multiple contacts; see section 4.2 on page 19.

*Add new tab.* Click to create a new tab to reflect communication with a different contact or group of contacts; see section 4.4.1 on page 21.

*Pull down to show more tabs.* Display the list of tabs when there is insufficient space to view all of the tabs above the history pane; see section 4.4.1 on page 21.

*Select folder.* Choose which folder contains the recipient of the message; see section 3.2.1 on page 15 and section 3.3.1 on page 16.

*Filter button.* This filters the contacts from the specified folder that GTXT will show in the contacts panel; see section 3.3.1 on page 16.

*Show button.* This changes how GTXT presents (or shows) the contacts from the specified folder in the contacts panel; see section 3.3.2 on page 16.

*Contacts panel.* Displays selected contacts; see section 3.3 on page 16. To show this panel, click on the **Contacts** button; see section 3.5 on page 18.

*Search or add number.* By typing a few characters in this field, you can filter the contacts that GTXT displays in the contacts panel; see section 3.3.3 on page 17. You can add a phone number here to be a recipient of a message by typing a number; see section 3.4.2 on page 17.

*Z button.* Configure how GTXT communicates with the GS8; see chapter 2, starting on page 11. Also allows you to add and delete contacts from a local file; see section 3.2 on page 15.

*Hide button.* Hides the contacts panel; see section 3.5 on page 18.

*Contacts Button.* This is visible only when the Contacts panel is hidden. Click this button to show the contacts panel.

*Recipient pane.* Displays the contacts that will receive the message you are composing; section 3.4 on page 17.

Close button. Exit the GTXT application. See section 1.4.4 on page 7.

*Link button.* This feature allows you to send a message upon a change of a record in a database. The contents of the message can comprise data from other fields in the database. See chapter 6, starting on page 25.

*Schedule button.* This feature allows you to send a message in the future. You can send the message one time or at a recurrent period that you determine. See chapter 5, starting on page 23.

Send button. Send the message immediately; see section 4.2 on page 19.

*SMS and character counter.* GTXT shows how many SMS messages your text will use and how many characters remain to fill the current SMS; see section 4.2 on page 19.

*Message pane.* Displays the message that you are composing; see section 4.2 on page 19 for standard messages, see section 5.2 on page 23 for scheduled messages, and section 6.3.4 on page 31 for messages that are linked to a database.

*History pane.* Displays a standard message that you have sent or received; see section 4.2 on page 19 and section 4.2 on page 19. Displays a message that you have scheduled; see section 5.2 on page 23. Displays a message that is linked to a database; section 6.3.5 on page 32.

## **1.6** About our Documentation

### 1.6.1 Related Documentation

If you are installing or supporting the GS8, please obtain the **GS8 Quick Start Guide** and the **GS8 User's Manual** from the Zed-3 web site:

http://www.zed-3.com/web/EN/Software\_n\_Data.htm

On our web site, we also have application notes and white papers on the use and integration of the GS8 and GTXT into your work environment. These are available here:

http://www.zed-3.com/web/EN/Application\_Notes.htm

### 1.6.2 Colophon

This document was produced on personal computers using Adobe's FrameMaker for Windows.

The headings are set in Swiss 721, Bitstream's version of the Helvetica<sup>™</sup> typeface; the copy is set in Zapf Calligraphic, Bitstream's version of the Palatino<sup>™</sup> typeface; notices are set in Swiss 721 or News Gothic, Bitstream's version of the Kingsley-ATF Type Corporation typeface. The drawings were produced using Adobe Photoshop and Adobe Illustrator.

### 1.6.3 Documentation Feedback

Zed-3 appreciates any constructive feedback on all our documentation. If you have comments or error reports on any Zed-3 documentation, please submit your feedback to:

Technical Publications Department Zed-3 501 Valley Way Milpitas CA 95035 USA techpubs@zed-3.com

## Chapter 2

## Configuration

## 2.1 Introduction

You can select the language for the GTXT application. This changes the menus and messages of the software program and does not limit the languages of the messages you can send and receive.<sup>1</sup>

You need to let GTXT know where and how it can communicate with the GS8. Follow the instructions in this chapter to do this. You can use the GTXT application free to send your first 300 messages. After that, you need to obtain a license to continue to use the software. This section explains how to obtain and install a license.

The various elements of the GTXT window are shown in section 1.5 on page 8.

## 2.2 Configuring the Regional Options

### 2.2.1 Selecting the Language for the Application

- 1. Open the GS8 application, either within Outlook or stand alone (see section 1.4 on page 6 for details how to do this).
- **2.** Ensure the Contacts panel is visible. If it is not, click on the Contacts button on the bottom of the GTXT window.
- **3.** Click on the **Z** button in the bottom right corner of the GS8 application and select Configure. GTXT opens the Configure window as shown below.
- **4.** In the regional settings, use the pull down menu to select the language that you want to use for the GTXT application. The possible languages are:

English Russian Chinese (simplified) Spanish

<sup>1.</sup> See appendix A, starting on page 45 for details on the way languages are transmitted with the GTXT application and the GS8 modular gateway.

Configure	account on G58	×
GS8 Modul	ar Gateway	
-Location -	<ul> <li>IP Address</li> <li>Domain Name</li> <li>Port</li> <li>Name (optional</li> </ul>	al)
-Login	User Name Password	USofA × ••••• × Save Password ssages at startup (operator)
Licensing You can Enter you	Product I send 421 more SM r license key to kee	d : 001EEC1E14BC Copy ID S messages. p GTXT operational Register
- Regional -	Language SMS size	English (USA)
	Save	Close

#### 5. Click **Save**.

6. Close the GTX application by clicking on **Close**. Start the GTXT application again for the new language settings to take effect.

### 2.2.2 Selecting the Default SMS Size

Under the GSM specifications, the size of an SMS is 160 characters. You should normally leave the radio button to select the SMS size to 160 characters. However, some service providers have installed equipment that limits the size of the message to 140 characters. If you believe that your messages are limited to 140 characters, select 140 on this pane.

This parameter does not restrict the size of the message that you can send or receive. It adjusts the counter on the GTXT application that indicates how many characters are in each message. See section 4.2 on page 19 for information about the counter. The parameter also affects how long messages are split into multiple SMSs. See also section 4.2 and section A.5 on page 48 for more information.

## 2.3 Configuring the Connection to the GS8

- 1. Open the Configuration window as described in section 2.2 on page 11.
- 2. You can specify the location of the GS8 either by an IP address or an FQDN. We recommend you use an FQDN as it allows your use of the GS8 to be more portable.

Write the FQDN like this example:

my-GS8.my-company.com

Write the IP address like this example:

123.456.78.90

- **3.** The default port is 16200. Do not enter a comma. Use the actual port specified by your administrator.
- 4. Give this GS8 a name if you want, such as

My Office GS8

5. Select from which slots in the GS8 you want messages to be sent. We recommend you put your main GSM or CDMA module in slot 1. If you use only this slot, the recipients of your messages will always get messages from one number and they can associate that number with your business. If you send many messages, you may need to use multiple GSM or CDMA modules to send the data so that the time taken to send many messages is reduced. You should then enable multiple slots. GTXT will send messages in sequence to each of the slots, starting with slot 1 (if enabled) for the first message, slot 2 (if enabled) for the second message, and so on.

This selection has no effect on received messages. The GS8 will read SMS messages from any slot and will forward to GTXT users according to the rules programmed on the GS8.<sup>1</sup>

- 6. Enter the user name and password that you or your administrator configured on the GS8. (On the GS8 administration software, you configure this under SMS | GTXT Client.)
- 7. Select whether you want GTXT to read the messages stored on the GS8 at startup. When no GTXT users are logged into the GS8, the GS8 will save messages that are received. When you enable this selection, GTXT will retrieve those SMS messages when it starts. You normally enable this if you are performing an operator type of function and will distribute the messages to others who are using GTXT and the GS8. Multiple users can enable this selection and the first person to log in after the GS8 has stored messages will receive them.

If you are the only user of GTXT on the GS8, you should enable this selection. If there is a group of users, at least one of you should enable this selection.

8. Click Save.

## 2.4 Connecting to the GS8

After you close the configuration window, GTXT attempts to register with the GS8. GTXT displays the status of the connection on the title bar of the GTXT application. The displayed conditions are:

*Resolving domain name.* This indicates that GTXT is trying to find the IP address that corresponds to the FQDN that you entered.

**Unable to resolve domain name.** GTXT could either not communicate with a DNS server to resolve the domain name or the DNS server could not provide an IP address for the domain name. The possible faults are that you do not have connectivity to the network or you incorrectly typed the domain name.

*Authenticating.* This means that GTXT has found the GS8 and has sent the user name and password. GTXT is waiting for a positive response.

<sup>1.</sup> See appendix B, starting on page 49 for details how to configure an operator type of user on the GS8.

*Login failure.* The login credentials (user name and password) were rejected by the GS8. Check that you typed these correctly. If you are sure you have done so, contact the administrator of the GS8.

*Connected to <description>*. This is the normal indication. It will show the name if you have given one, the FQDN if you specified one, or the IP address.

If the GTXT software has connected to the GS8, you are now ready to send and receive SMS text messages.

## 2.5 Licensing

#### 2.5.1 License Choices

You can use the GTXT application freely to send 300 SMSs. After you have sent that number of messages, you must buy a license to continue to use GTXT. Zed-3 sells licenses that allow you to send:

- 1000 SMSs
- an unlimited number of SMSs

If you send only a few messages a day, the license for 1000 messages may be your best choice. If you use GTXT heavily, you should buy an unlimited license.

The license is tied to your particular PC. If you change your PC you must buy a new license for the other PC.

#### 2.5.2 Obtaining a License

- 1. Open the Configuration window as described in section 2.2 on page 11.
- 2. Click on **Copy ID** to copy the product ID to your PC's clipboard.
- **3.** Paste that product ID in an e-mail and send it to your reseller authorized to sell Zed-3 products.
- 4. You will receive the license by e-mail.

#### 2.5.3 Installing the GTXT License

- 1. Open the Configuration window as described in section 2.2 on page 11.
- 2. Paste the license in the window and click on **Register**.
- 3. GTXT will increase the number of messages you can send or set them to unlimited.

## Chapter 3

## Selecting the Recipient

## 3.1 Introduction

### 3.1.1 Distinction between Recipient and Contact

A recipient is a person (more specifically a phone number) to whom you are sending a message. A contact is a person (again, strictly a phone number) that resides in one or more files accessible to your computer.

### 3.1.2 Choices of Contacts

When you are using GTXT as a stand alone application, you can select the recipient of an SMS by choosing a contact from a local file of names and numbers or by directly typing in the number.

When you are using GTXT from within Outlook, you can select the recipient of an SMS by choosing a contact from an Outlook contact folder, a local file of names and numbers, or by directly typing in the number.

### 3.1.3 Sending Messages

You can send one SMS to one recipient or you can send the same SMS to multiple recipients. You can send many different SMSs to one or many different recipients.

The various elements of the GTXT window are shown in section 1.5 on page 8.

## 3.2 Editing a Local File of Names and Numbers

### 3.2.1 Adding Names to the Local Files

- 1. Ensure the contact panel is visible. Click on **Contacts** if it is not.
- 2. From the pull down list of folders at the top of the contacts panel, select Local.
- 3. Click on the **Z** button at the bottom and choose **Add**.
- 4. Type in the name and number of this contact.

- 5. Select a phone type from the list. Usually, this will be **Mobile**.<sup>1</sup>
- 6. Click Save.
- 7. Repeat to add as many contacts as required.

### 3.2.2 Changing or Removing a Name from the Local File

- 1. Ensure the contact panel is visible. Click on **Contacts** if it is not.
- 2. From the pull down list of folders at the top of the contacts panel, select **Local**.
- 3. Click on the **Z** button at the bottom and choose **Delete** or **Edit**.

## **3.3** Filtering and Displaying Information for the Contacts

### 3.3.1 Type Filter

- 1. Ensure the contacts panel is visible. Click on **Contacts** if it is not.
- 2. From the pull down list of folders at the top of the contacts panel, select the folder that contains the name you want to use, or select **All** if you are unsure in which folder the specific contact is located.<sup>2</sup>
- 3. Click on the **Filter** button at the top of the panel.
- **4.** Select which types of numbers you want to see displayed. For example, you usually cannot send an SMS to a home number or a fax number, so there is no point showing these. In general, you will show numbers that are **Mobile** and **Business 2**.<sup>3</sup>
- 5. GTXT will show only the contact types that are relevant.

#### 3.3.2 Show

- 1. Click on the **Show** button at the top of the panel.
- 2. Select what you want to see and click **OK**. The choices are:
  - **show icon** (GTXT displays a picture to the left of each contact)
  - **show phone** (GTXT displays the phone number of this contact)
  - **show type** (GTXT displays the type of this phone number)
- **3.** The selection you make on the **Show** button also affects how GTXT displays information about the selected recipient.

<sup>1.</sup> If the contact has a GS8 at their business to make and receive calls, you may wish to choose **Business 2** or **Radio** to indicate this number.

<sup>2.</sup> When you select **All**, the performance will be degraded as GTXT needs to search through many more names and numbers.

<sup>3.</sup> We recommend that you use **Business 2** for contacts that have a GS8 in their business that you can call or send SMS to.

### 3.3.3 Match

- **1.** Type in a few characters of the person or phone number you want to locate into the text field at the bottom of the contacts panel.
- **2.** GTXT will display only those names or number that match the characters that you have typed. For example, if you type **an**, then GTXT will show the names: Angela, Frank, and Wang. GTXT will look only in the selected folder.
- **3.** If you type some digits, GTXT will select to show the phone number regardless if you selected to have it already displayed with the **Show** button as described in section 3.3.2 above. For example, if you type 408, then GTXT will show the numbers 408-587-9333, 650-408-1234, and +86-1391-6408. GTXT will look only in the selected folder.
- **4.** To remove this match, delete the characters in the text field. GTXT will show all the contacts once more.

## 3.4 Choosing Recipients of the SMS

### 3.4.1 Adding a Contact

- 1. Ensure the contact panel is visible. Click on **Contacts** if it is not.
- 2. From the pull down list of folders at the top of the contacts panel, select the folder that contains the name you want to use, or select **All** if you are unsure in which folder the specific contact is located.<sup>1</sup>
- 3. Find the name you want, highlight it, and do either:
  - **a.** double click the name
  - b. right click and select Add Selected
- 4. You can select multiple names by doing one of the following:
  - a. highlight the first name and press Shift-click on a range of names
  - b. highlight the one of the names and press Ctrl-click on subsequent names
  - c. highlight any name and press Ctrl-A to select all names
- 5. When you have selected multiple contacts, use right click and then **Add Selected** to add them to the list of recipients.
- **6.** GTXT moves the name of the recipient to the recipient panel.

### 3.4.2 Adding a Number

- **1.** To add a number without creating a new contact in the local file or adding it to an Outlook file, type the number in the text field at the bottom of the contacts panel.
- **2.** GTXT will start to search through the contacts in the currently selected folder to see if there is a match, as described in section 3.3.3 above.

<sup>1.</sup> When you select **All**, the performance will be degraded as GTXT needs to search through many more names and numbers.

- 3. If GTXT cannot find a match, it will display all the contacts in the folder again and display a plus sign (+) to the left of the text field.
- 4. Click on the + to add that number to the list of recipients.
- 5. GTXT moves the number you typed to the list of recipients and clears the text field.

#### 3.4.3 Removing Contacts

You can remove contacts from a local file as described in section 3.2.2 above. You remove contacts from an Outlook folder using the tools incorporated into Outlook.

### 3.4.4 Removing Recipients of the SMS

- 1. Find the name you want in the recipient panel, highlight it, and do either:
  - **a.** double click the name
  - b. right click and select Remove
- 2. You can select multiple names by doing one of the following:
  - **a.** highlight the first name and press Shift-click on a range of names
  - b. highlight one of the names and press Ctrl-click on subsequent names
  - c. highlight any name and press Ctrl-A to select all names
- **3.** When you have selected multiple contacts, use right click and then select **Remove** to remove them from the list of recipients.

## 3.5 Hiding and Showing the Contacts Panel

Click on the **Hide** button to hide the panel. This occupies less space on your screen and is convenient if you have tabs open for the people you will communicate with. Click on the **Contacts** button to display the panel again.

## 3.6 File Locations

The contact files are stored in these locations. For Outlook:

C:\Documents and Settings\<local user>\Local Settings\Application Data\Microsoft\Outlook\outlook.pst

For GTXT local files:

C:\Documents and Settings\<local user>\Local Settings\Application Data\Zed-3\GTXT\My\_Local\_Contacts.xml

The path for local user is that determined by your login to Windows.

## Chapter 4

## **Messages and History**

## 4.1 Introduction

You can keep track of each of your SMS messages using tabs within GTXT. You can display many tabs to easily access a specific SMS interaction. After you close a tab, when you next start a communication with a specific contact, GTXT shows you previous communications with that contact.

The various elements of the GTXT window are shown in section 1.5 on page 8.

## 4.2 Sending a Message

- 1. Select the recipient or group of recipients as described in section 3.4 on page 17.
- 2. Enter the message in the message pane by doing one of the following:
  - a. click in the message pane and type the message
  - **b.** copy the text you want to send from another application (such as Word or Notepad), click in the message pane and press Ctrl-V (paste)
  - **c.** highlight the text you want to send in another application (such as Word or Notepad), then drag the text into the message pane (use Ctrl-drag if you want to retain the text in the original application)
- **3.** As you type, GTXT indicates how many characters remain to fully use this SMS. If you are typing regular text characters (similar to ASCII characters), an SMS can have 160 characters. If you are typing with Unicode (extended) characters (such as Chinese or Russian characters), an SMS can have 70 characters.
- **4.** If you exceed the number of characters that will fit into a single SMS, GTXT indicates you have two SMSs and restarts the counter for the number of characters that remain in the second message.<sup>1</sup>
- 5. You can continue to create a very long message that will comprise many SMSs.<sup>2</sup>

<sup>1.</sup> Appendix A, starting on page 45, describes in detail the encoding used to send an SMS message and the total number of characters available in a specific message.

<sup>2.</sup> If the recipient is using a mobile phone, one communication split across many separate SMSs may be a distraction. We therefore advise you to limit a message to one or two SMSs. Depending on the service provider and the mobile phone, the message may appear as separate SMSs on the recipient's mobile phone or as a single SMS.

- 6. When you are satisfied with your message, click **Send** or press **Ctrl-Enter**.
- 7. GTXT sends the message to the GS8. If you are sending a message to multiple recipients, GTXT sends separate messages to the GS8 for each recipient. One port on the GS8 can send out approximately ten messages per minute.<sup>1</sup>
- 8. GTXT changes the status in the title bar briefly, to indicate it is sending the message.
- **9.** GTXT changes the name of the current tab to be the name of the recipient or, in the case of a message sent to a group of contacts, it displays the number of recipients followed by the word contacts.
- **10.** GTXT moves the message to the history pane and precedes it with the date and time. The format for the date and time is that which you have configured for your PC. GTXT shows this message in grey color.

## 4.3 Receiving Messages

#### 4.3.1 Conditions for the GS8 to Send a Message to You

When a recipient replies to your SMS, the GS8 may return the message to you. This depends on the way you (or your system administrator) have configured the GS8.

When you have a GS8 that is used by multiple people, and the GS8 receives a message, the default setting may be to send it to a specific person, for example, the operator.<sup>2</sup>

However, you can configure the GS8 so that when a message is received, then that message is sent to a specific GTXT user if that user had previously sent a message to that recipient within a specific time. You might select the time to be several minutes, depending on your application.

Alternatively, if you are the only user of a specific GS8, you may choose that all messages are received by your GTXT.

In any event, you must have GTXT active (running on your PC) to receive any messages.

#### 4.3.2 Showing a Received Message in GTXT

GTXT displays any received messages in the history pane and precedes it with the date and time. The format for the date and time is that which you have configured for your PC. GTXT shows this message in blue color.

If you have one communication with one recipient, then you will have only one tab open and active. All messages will appear under that tab.

<sup>1.</sup> If you have a single port on your GS8, and you are sending a message, for example, to 100 recipients, the time between the transmittal of the first and last messages could be 10 to 15 minutes (and longer if other users are using the GS8). To reduce this time, add multiple GSM or CDMA modules to your GS8.

<sup>2.</sup> See appendix B, starting on page 49 for details how to configure an operator on the GS8.

## 4.4 Using Tabs

You can use GTXT with just a single tab visible at any time. However, if you have communications with multiple recipients (or groups of recipients), you will want to use the tab feature on GTXT to organize these communications.

### 4.4.1 Creating a New Message

If you want to send a message to a different recipient or group of recipients, click on the + to add a new tab (see section 1.5 on page 8). Then select your recipient or recipients as usual and compose and send the message as usual.

The new message appears in a new tab, which is the *active* tab. All other tabs are *inactive*.

Continue to add tabs as required for different communications. When there is insufficient space along the top of the history pane to display all the tabs, GTXT shows a pull down triangle to permit access to the tabs you cannot see.

### 4.4.2 Receiving a Message When You Have Multiple Tabs

If you have multiple communications (and therefore multiple tabs), GTXT will place the message in all applicable tabs that you have open. The applicable tabs are those in which the recipient of an SMS is a member.

For example, suppose you have sent the following message:

- message 1 to Allen, Beth, and Charles
- message 2 to Beth and Danielle
- message 3 to Charles and Danielle
- message 4 to Beth only

#### When:

- **a.** Allen sends a reply, GTXT will display it in tab 1.
- **b.** Beth sends a reply, GTXT will display it in tabs 1, 2, and 4.
- c. Charles sends a reply, GTXT will display it in tabs 1 and 3.
- d. Danielle sends a reply, GTXT will display it in tabs 2 and 3.

### 4.4.3 Closing a Tab

To close a tab, click on the X that GTXT shows to the right of the contact's name on the tab. You cannot undo this action.

However, if you want to show the contents of the history pane again, create a new tab as described in section 4.4.1 above, and place the recipient or recipients of the messages on the tab that you closed into the recipient pane. GTXT shows the history associated with this contact or these contacts. This is described further in section 4.5 below.

## 4.5 History

### 4.5.1 Messages Saved in Outlook

Upon installation, GTXT creates sub-folders titled SMS under the Inbox, Outbox, and Sent Items folders of Outlook.

When you send an SMS, GTXT saves it under Personal Folders | Outbox | Sent Items | SMS.

When you receive an SMS, GTXT saves it under Personal Folders | Inbox | SMS.

When you have attempted to send an SMS that was not sent, GTXT saves it under Personal Folders | Outbox | SMS.

You can view, move, and delete these as you would with e-mail messages.

### 4.5.2 Messages Saved in GTXT

When you select a recipient for message as described in section 3.4 on page 17, GTXT automatically displays the history of your previous messages (sent and received) with this contact.

If you had sent a message to a group of contacts, and the contact you have selected was a member of that group, GTXT displays the messages that you sent to that group. However, GTXT does not display the messages you received from all members of that group. Instead, GTXT will display only those messages received from the selected contact, regardless whether they were sent in response to the group message or individual messages.

## Chapter 5

## Scheduling a Message

## 5.1 Introduction

This feature allows you to send a message in the future. You can send the message one time or at a recurrent period that you determine.

The various elements of the GTXT window are shown in section 1.5 on page 8.

## 5.2 Configuring the Schedule

- 1. Select the recipient or group of recipients as described in section 3.4 on page 17.
- 2. Create your message as described in section 4.2 on page 19.
- 3. Click on the **Schedule** button. GTXT opens the Schedule window as shown below.

SMS body:	List of recipients:
Please remember to bring when you come to Mike's you, Joe	the new water pump Garage today. Thank (510) 555-8765
Send SMS at Thur	sday 29 January 2009 🔽 day 06:00: <b>42 </b> time
Send	Once O Recur
Recurrence pattern	
Intraday	every 1 minutes,
○ Daily ○ Weekly	except from: set to set
○ Monthly ○ ○ Yearly	at add
Range of recurrence	
Start: Wednesday	v 28 January 2009 💌
No end date	
	Occurences
<ul> <li>End after</li> </ul>	

- 4. Enter a name for this schedule. This is used to access the schedule in future.<sup>1</sup>
- 5. Choose the time of day at which you want GTXT to send this SMS. GTXT displays the time format for you to enter the time in the format configured for your PC. If you are going to send the SMS only at one time, GTXT will send the message anywhere from one minute to 23 hours and 59 minutes in the future.
- **6.** Select whether GTXT will send this message once with a recurrence by using the radio button.
- 7. If you want the message only to be sent once, click **OK**.
- 8. When you want the message to be sent with a recurrence, select whether GTXT should send the message during a day (intraday), daily, weekly, monthly, or yearly by clicking on the appropriate radio button. GTXT changes the configuration panel to the right to match the period you selected.
- 9. Complete the pattern on the right pane.
- 10. Enter the range of the recurrence.
- 11. Click **OK**. GTXT moves the message to the history pane, but does not show any date or time.
- **12.** On the tab, GTXT shows a clock before the name you gave to this schedule. GTXT shows a recurrent symbol if the message is set to repeat.

## 5.3 Editing a Scheduled Message

You can modify the parameters in the Schedule window for a message you have configured.

However, you cannot change the text of the SMS nor can you change the recipients of the SMS. If you need to make a change to the contents or distribution list for a scheduled message, delete the message and create a new one.

To edit the schedule:

- 1. Click on the **Schedule** button. GTXT now shows you the choice of creating a new scheduled message or editing an existing message. Select **Edit**.
- 2. GTXT shows you a list of the scheduled messages.
- 3. Select the one you want and click Edit.
- 4. Alter the parameters as required and press **OK**.

As a shortcut, you can select the tab for the scheduled message and double click in the history pane. This opens the schedule for the selected message.

## 5.4 Deleting a Scheduled Message

- 1. Open the list of scheduled messages or the schedule window for the SMS you want to delete, as described in section 5.3 above.
- 2. Click on Delete.
- 3. Confirm that you want to delete the scheduled SMS.

<sup>1.</sup> If you do not enter a name, GTXT will assign the name **Untitled Schedule 1** for the first such scheduled message.

## Chapter 6

## Linking to a Database

## 6.1 Introduction

This feature allows you to send a message in which the contents of the message comprise data from fields in a database. The message is sent to a recipient whose phone number is also in the database. The message can be sent only when there is a change to the data in the database.

The various elements of the GTXT window are shown in section 1.5 on page 8.

## 6.2 Specifying the Database and Fields

### 6.2.1 Connecting to the Database

- **1.** From the main GTXT window, **do not** select the recipient (or group of recipients) as you usually do from the contacts panel.
- 2. Click on an empty tab or on the + symbol to open a new empty tab.
- 3. Click on the **Link** button. GTXT opens the Link window as shown below.

📕 Link to Database	×
Name of this link	
File Send Fields Preview Write back	
Database type MySQL	
Connection  IP Address  Domain name	
Port User name Database name	<b>-</b>
Password Table name	<b>V</b>
Connect	
Save	[ Cancel ]

4. Enter a name for this link. This is used to access the link in future.<sup>1</sup>

- 5. Select the type of database. The choices are MySQL, Microsoft SQL, or CSV. The CSV file is not really a database but can be very useful for simple linked messages. You can easily create a CSV file from a spreadsheet such as Microsoft Excel.
- **6.** Specify the location for the database (or for a CSV file, its directory). You can specify the location of the database either by an IP address or an FQDN. We recommend you use an FQDN as it allows your use of the GS8 to be more portable.

Write the FQDN like this example:

my-database.my-company.com

Write the IP address like this example:

123.456.78.90

- 7. Enter the port used for your database. The default port for MySQL database is 3306. Do not enter a comma. Use the actual port specified by your administrator.
- 8. Enter the user name and password that you or your administrator configured on the database.
- 9. Click on **Connect**.
- **10.** GTXT will show an error message if it cannot access the database, if it cannot open the database, or if the database is of the wrong format.
- **11.** If GTXT can open the database, it makes the connection panel grey. Select the database name and the table name from the pull down menu. The window will look like the figure below:

🚄 Link to Database	2			×
Name of this link	Caris ready 🛛 🗙			
File Send	Fields Preview	ſ	Write back	
Database type	lySQL -)			
Connection			1	
IP Address				
Domain name	dbase.garage.com	×		
Port	3306	×		
User name	Zed-3_Demo	×	Database name	auto_repair 💌
Password	test	×	Table name	mikes_garage 💌
Di	sconnect			
			]	
			_	
			S	ave Cancel

#### 6.2.2 Specifying the Conditions Under Which the Message is Sent

- 1. Click on the **Send** tab.
- 2. You can send the message under the following conditions:

<sup>1.</sup> If you do not enter a name, GTXT will assign the name **Untitled Link 1** for the first such linked message.

**Unconditionally.** GTXT will send the message without regard to the contents of the database. Use this if you want to send a message to a group of people and the message will leave at the same time. For example, you may want to tell all customers that you have stock of a new product (so the content of the message is identical). You may send a message to all your customers to tell them the balance of their account (so the content of the message is taken from the database).

**On conditions.** GTXT will send the message only when a field (or multiple fields) match requirements that you configure. Use this method if you want to send a message to a group of people only when something changes in the database. For example, you may want to tell a customer that his or her car is now repaired, is ready for collection, and the repair cost. You may want to send a message to your customers when their account reaches a certain credit limit.

*Program.* You can write SQL code and enter it here. This is for advanced users only. GTXT will send that code to the database and send the message only when the database returns TRUE. You are responsible for ensuring the code is properly formatted SQL code.

- 3. Click on the **Fields** tab.
- **4.** If you selected Unconditionally or Program in step 2 above, GTXT displays a list of the fields in the database. You can make no changes to this pane. This is shown below:

📕 Lin	k to Database	<u>×</u>
Name	e of this link Car is ready ×	
File	Send Fields Preview	Write back
	Field Name	Field Type
•	id	int(3)
	Car_Make	varchar(20)
	Car_Model	varchar(20)
	Name_of_Owner	varchar(30)
	Mobile_Number	varchar(12)
	Status	varchar(20)
	Repair_Cost	varchar(10)
	Date	varchar(30)
	Time	varchar(10)
*		
		Save Cancel

**5.** If you selected On Conditions in step 2 above, GTXT displays a list of the fields in the database and you can decide under what conditions GTXT will send the message. An example is shown below:

Nam	e of this link Car	is ready	×	
File	Send F	ields Previ	ew 🌒 Write b	ack
	Field Name	Field Type	Expression	Value
	id	int(3)		
	Car_Make	varchar(20)		
	Car_Model	varchar(20)		
	Name_of_Ow	varchar(30)		
	Mobile_Number	varchar(12)		
ø	Status	varchar(20)	=	READY
	Repair_Cost	varchar(10)		
	Date	varchar(30)		
	Time	varchar(10)		
*				

- **6.** For the Expression, you can type =, <, >, <=, or >=, !=, LIKE, NOT LIKE.
- 7. Type the value in the same row.
- 8. Press **Enter** after you write each expression or value.
- **9.** You can create multiple conditions. All conditions must be true for GTXT to send the message.
- 10. Click on the **Preview** tab.
- **11.** GTXT shows you those entries in the database that match all the conditions (if any) that you have configured in step 6 and step 7 above.

### 6.2.3 Writing Back to the Database

1. Click on **Write back**. GTXT shows a window as follows:

📕 Link to Database		×
Name of this link Car is ready ×		
File Send Fields Preview	Write back	
On success O Do nothing		
Change Status	▼ to SENT ×	
O Program		
On failure O Do nothing		
Change	▼ to	
O Program		
If message length greater than one SMS Do not send message Do nothing		
Change	▼ to	
O Program		
	Save	Cancel

2. You can select whether GTXT will modify the database after it has sent (or attempted to send) the message. The three events are:

On success. If GTXT successfully sent the message to the GS8, it will execute this selection.<sup>1</sup>

*On failure.* If the GTXT application is unable to send the message to the GS8, it will execute this selection.

*If message length is greater than one SMS.* If the message is greater than can be transmitted in a single SMS, GTXT will execute this selection. One SMS is typically 160 ASCII characters or 70 Unicode characters. You may use this if you think that a longer message will disturb the recipient. You can decide whether GTXT should send the message or not.<sup>2</sup>

3. Within each event, you can select to do one of the following:

*Do nothing.* GTXT will make no modifications to the database when it executes that event.

*Change.* GTXT will send the appropriate SQL command to the database to change a variable. Use the pull down list to select the field that you want changed. Type in the new value of that field. GTXT does not perform any error checking, so if you indicate that you want a numerical field to be set to a character string, the database will reject that instruction and the field will not change.

You will usually use this choice if you need to indicate that an SMS has been sent (or if it has not). An example may be a car repair garage. When the vehicle has been repaired, the status may be set to READY. After the message has been sent, GTXT can change the status to SENT to indicate a message has been sent to the owner of the car.

*Program.* You can write SQL code and enter it here. This is for advanced users only. GTXT will send that code to the database. You are responsible for ensuring the code is properly formatted SQL code.

<sup>1.</sup> In the current version of software, the GTXT application only knows that the GS8 has received the message, not that it was successfully transmitted.

<sup>2.</sup> See appendix A, starting on page 45 for details of the characters supported by different messages and the length of messages.

4. When you have completed each of the tabs in the window Link to Database, click on **Save**.

## 6.3 Creating a Linked Message

#### 6.3.1 Message Text

- **1.** Connect to the database and configure the conditions (if any) under which the message will be sent as described in section 6.2 above.
- **2.** GTXT will display the screen as shown below. GTXT has replaced the list of contacts with the names and types of the fields in the database:

Zed-3 SMS Connected to gs8-calif.zed-3.com	
Car is ready × +	auto_repair.mikes_
	Car_Make varchar(20) Car_Model
	varchar(20) Name_of_Owner varchar(30)
	Mobile_Number varchar(12)
	Status varchar(20)
	Repair_Cost varchar(10)
	Date
Send Schedule Link Close	Insert Send to

- 3. Type text into the message window to create the message.
- 4. Specify a field that will be part of the message by doing one of the following:
  - highlight the field in the list on the right and double click on the field
  - highlight the field in the list on the right, right click, and select Insert
  - highlight the field in the list on the right, then click on the button labelled **Insert** below
  - in the message pane, right click and select the field from the list
- **5.** GTXT adds the field into the message at the cursor location. The field name is displayed in red characters between square brackets. (Remember to add space characters as required before or after the field.)

#### 6.3.2 Message Recipient

- **1.** Once you have created the message, highlight the field in the database that contains the number where GTXT should send the message and do one of the following:
  - right click, and select **Send to**
  - click on the button labelled **Send to** below

- **2.** GTXT displays the field name containing the recipient's phone number in the recipient pane.<sup>1</sup>
- 3. The window may look like this example:

Zed-3 SMS Connected to dbase.garage.com				<u>_     ×</u>
Car is ready × +		aut	o_repair.mikes_ <sup>id</sup>	Î
			Car_Make varchar(20)	
		1.	Car_Model varchar(20)	
			Name_of_Owner varchar(30)	
have repaired your [Car is Mot ready.Car_Make] [Car is	ile_Number		Mobile_Number varchar(12)	
<pre>ready.Car_Model] and it is now ready for you. The total bill is \$[Car is ready.Renair Cost]. Thank you: Mike's</pre>			Status varchar(20)	
Garage			Repair_Cost varchar(10)	
Characters number unknown			Date	-
Send Schedule Link	Close	Ins	ert	Send to

This may result in a message such as:

Hello Susan Thomas: We have repaired your Ford Escort and it is now ready for you. The total bill is \$123.45. Thank you; Mike's Garage

GTXT will send the message to the mobile phone number on record for Susan Thomas.

### 6.3.3 Sending the Message Immediately

- 1. Click on the **Send** button to send the messages.
- **2.** GTXT will access each row of the database and see if the fields in that row match the criteria (if any) that you configured in section 6.2.2 on page 26.
- **3.** If the fields match the expressions and values, GTXT will send build the messages using the data in the database and to the phone numbers that you have specified.
- 4. If you selected that the messages will be sent unconditionally, GTXT will send a custom message to each person.<sup>2</sup>

### 6.3.4 Scheduling a Linked Message

You may want to schedule a linked message so that:

a. the message is sent at a time more convenient or appropriate for the recipient

<sup>1.</sup> You cannot send a linked message to a contact contained in your local or Outlook folders. GTXT can send a linked message only to a number that is contained in the database.

<sup>2.</sup> If you have a message that will occupy more than one SMS, and you selected the option not to send such messages, TXT will not send the message.

- **b.** you want to send a message periodically, with customized content and possibly to different recipients each time
- **c.** you want GTXT to look at the database periodically and send a message when a field changes in the database

Follow the steps below:

- 1. Create a linked message as described in section 6.3.1 and section 6.3.2 above. Ensure you are satisfied with the message and the recipient number before proceeding because you cannot edit the link after you have scheduled it.
- 2. Click on **Schedule**. GTXT opens the schedule window.
- 3. Configure the schedule as shown in section 5.2 on page 23.
- **4.** If you want to send the message once (as in example (a) above) select the date and time for the message and click OK.
- 5. If you want to send the message periodically (as in example (b) above), click **Recur**, and then select if you want the message to be sent daily, weekly, monthly, or annually.
- 6. If you want to send the message when a field changes in the database (as in example (c) above), click **Recur**. Select how often you want GTXT to scan the database. If the data changes frequently, you may want GTXT to access the database every minute. You should choose a time that is appropriate for the application. When GTXT accesses the database, it may create much traffic on your data network and may use much processing resources.

### 6.3.5 How GTXT Displays the Linked SMS

After you have initiated the linked message, GTXT moves the message to the history pane, but does not show any date or time.

On the tab, GTXT shows a database symbol before the name you gave to this linked message as shown in section 6.3.2 on page 30. If you schedule the message, it appears as shown below:

Zed-3 SMS Connected to Mike's Garage			
C Link schedule: Unna × 2 Unsorted × + Link schedule named 'Unnamed schedule' starts at 1/29/2009 2: SMS body: Dear [Milpitas Cars: Name_of_Owner], your [Milpitas C Cars: Car_Model] is ready for pick up. The total bill is \$[Milpitas Car	41:00 PM ars.Car_Make] [Milpitas rs.Repair_Cost]. Thank you.		auto_repair.mikes_garage id int(3)
Mike's Garage. Link schedule was sent at 1/29/2009 2:41:41 PM Link schedule was sent at 1/29/2009 2:42:00 PM	14	Car_Make varchar(20)	
	14	Car_Model varchar(20)	
			Name_of_Owner varchar(30)
Mobile Number			Mobile_Number varchar(12)
			Status varchar(20)
			Repair_Cost varchar(10)
		. Au	Date varchar(30)
			Time
Characters number unknown			
Send (Schedule Link	Close	In	sert Send to

## 6.4 Editing a Linked Message

### 6.4.1 Message that is not Scheduled

To edit the link:

- **1.** Select the tab that has the linked message.
- 2. Click on the **Link** button.
- 3. Alter the parameters as required and press **OK**.

### 6.4.2 Message that is Scheduled

You can edit only the schedule, not the link.

- **1.** Select the tab that has the linked message.
- 2. Click on the **Schedule** button.
- 3. Alter the parameters as required and press **OK**.

## 6.5 Deleting a Linked Message

- **1.** Select the tab that has the link or scheduled link.
- 2. Click on **X** that is on that tab.
- 3. Confirm that you want to delete the message.

## Chapter 7

## Software API

## 7.1 About the API

This section describes how you can interface with the GTXT application programming interface if you want to write your own software to send and receive SMS and MMS messages.

The following operating systems are supported by the API:

• Windows 98, Windows 2000, Windows XP

The following programming languages are supported by the API:

• C++, MFC

The format of the library is as Microsoft VC++6.0 compiler with format of Regular DLL With MFC Statically Linked API. The functions are called in stdcall way.

## 7.2 Notifying Message

### 7.2.1 Function and Format

The GTXT API sends to clients the results of the requests that have been sent to server and event notifications from server via WM\_GTXT\_MESSAGE message.

The message format is:

WM\_GTXT\_MESSAGE

unsigned long wParam

unsigned long IParam

#### 7.2.2 Parameter wParam

The low-order word of wParam contains the type of the command, say:

GTXT\_EVENT\_CLIENT\_AUTH.

The high-order word of wParam contains the result of the request, the value may be:

RESPONSE\_SUCCEEDED: Request succeeds and you may get the result in the following part.

RESPONSE\_FAILED: Request fails, no usable value.

RESPONSE\_TIMEOUT: Request time-outs, no usable value.

### 7.2.3 Parameter IParam

There are two types of value, they are:

The dwFlag set when calling function GTXT\_RequestSendSMS is used to mark a specific message sending.

User remove a message by message ID.

## 7.3 Data Structure

#### 7.3.1 The Result of Message Sending

typedef struct tagGTXT\_SENDRESULT

{

unsigned short wDest;

unsigned short wResult;

#### } GTXT\_SENDRESULT, \*LPGTXT\_SENDRESULT;

wDest. The objective ID corresponding to the destination when sending a message.

*wResult.* The result of sending, GTXT\_SENDRESULT\_SUCCESSED means success, GTXT\_SENDRESULT\_FAILED means failure.

#### 7.3.2 Line Information

typedef struct tagGTXT\_LINEDATA

{

int nLine;

char szLine[GTXT\_STRLEN];

} GTXT\_LINEDATA, \*LPGTXT\_LINEDATA;

nLine. Line's ID.

szLine. Line's name.

#### 7.3.3 Message Receiving

typedef struct tagGTXT\_RECVSMS

{

unsigned long dwSMSID; char szSender[GTXT\_STRLEN]; int nLine; char szLine[GTXT\_STRLEN]; unsigned long dwTime; char szContent[GTXT\_SMS\_STRLEN]; } GTXT\_RECVSMS, \*LPGTXT\_RECVSMS; dwSMSID. Message ID received. szSender. Sender's number. nLine. Sender's line ID. szLine. Sender's line name. dwTime. Send time. szContent. The content.

## 7.4 Function Specification

### 7.4.1 Whether has uncompleted requests

Function Prototype: BOOL GTXT\_IsBusy(); Description: To find out if there is uncompleted requests. Parameters: None. Return value: Return TRUE if there is uncompleted requests, FALSE otherwise. Explanation: No.

### 7.4.2 whether connects to server

Function prototype: BOOL GTXT\_IsConnective();Description: To find out if the connection to server is built up.Parameters: none.Return value: TRUE if it has been connected to GS8, FALSE otherwise.Explanation: no.

### 7.4.3 Whether passes server's use authentication

Function prototype: BOOL GTXT\_IsAuthentic();

Description: To find out if the user has passed the authentication on the server. Parameters: none. Return value: TRUE if passed, FALSE otherwise.

Explanation: no.

#### 7.4.4 Set Message Notifying Window

Function prototype: void GTXT\_SetMessageWnd(HWND hWnd);

Description: Set message notifying window for results of requests or events.

Parameters:

• hWnd:The window HANDLE for receiving results of requests and event message notifications.

Return value: no.

Explanation: no.

#### 7.4.5 Connect to Server

Function prototype: BOOL GTXT\_ConnectServer(const char \*pszServer, int nPort);

Description: To connect to server.

Parameters:

- PszServer: The IP address of server.
- nPort: The port of server.

Return value: TRUE if succeeds, FALSE if the function fails or the connection exists already.

Explanation: TRUE only indicates the running success of this function, but not the success of connecting to server. Clients need to wait for WM\_GTXT\_MESSAGE message.

The low-order word of wParam is GTXT\_EVENT\_CONNECT\_SERVER, and the high-order word of wParam is the result of connecting. RESPONSE\_SUCCESSED on the high-order word of wParam means connection is built up. RESPONSE\_FAILED means connection fails.

#### 7.4.6 Cut the Connection to Server

Function prototype: BOOL GTXT\_DisconnectServer();

Description: To cut off the connection to server.

Parameters: none.

Return value: TRUE if the connection is cut off, FALSE if fails or the connection does not exist. Explanation: no.

## 7.4.7 User Authentication

#### 7.4.7.1 Client Requesting

Function prototype: int GTXT\_RequestClientAuth(const char \*pszName, const char \*pszPassword).

Decription: To send the authenticate request for user.

Parameters:

- pszName: User name.
- pszPassword: User password.

Return value: GTXT\_ERROR\_NOTCONN if there is no connection to server. GTXT\_ERROR\_PENDING if the server has not given out the result. GTXT\_ERROR\_OK if succeeds.

Explanation: GTXT\_ERROR\_OK only means the function's success, but not the approving of authentication. Clients need to wait for WM\_GTXT\_MESSAGE message.

The low-order word of wParam is GTXT\_EVENT\_CLIENT\_AUTH, and the high-order word of wParam is the result of authentication. RESPONSE\_SUCCESSED on the high-order word of wParam means it has received server's response, call function GTXT\_ResponseClientAuth to get the result of authentication. RESPONSE\_TIMEOUT means the server has not given out a response within the time-out limitation, and the authentication is rejected, the function GTXT\_ResponseClientAuth returns failure at the time.

#### 7.4.7.2 Server Responding

Function prototype: int GTXT\_ResponseClientAuth(int \*pnAuth).

Description: To get the result of user authentication

Parameters:

pnAuth: the result of authentication, may be one of these:

GTXT\_AUTH\_OK: authenticate approved

GTXT\_AUTH\_FAILED: authenticate rejected

GTXT\_AUTH\_ALREADYLOGIN: the user has logon already.

Return value: GTXT\_ERROR\_NOTCONN if there is no connection to server, GTXT\_ERROR\_OK otherwise.

Explanation: See the explanation of function GTXT\_RequestClientAuth.

### 7.4.8 Line Information

#### 7.4.8.1 Client Requesting

Function prototype: int GTXT\_RequestLineStatus().

Description: To send a request for line information.

Parameters: none.

Return value: GTXT\_ERROR\_NOTCONN if there is no connection to server, GTXT\_ERROR\_NOTAUTH before authentication, GTXT\_ERROR\_PENDING if the server has not given out result, GTXT\_ERROR\_OK if succeeds.

Explanation: GTXT\_ERROR\_OK only means the function's success, but not getting the line information. Clients need to wait for WM\_GTXT\_MESSAGE message.

The low-order word of wParam is GTXT\_EVENT\_LINE\_STATUS, and the high-order word of wParam is the result. RESPONSE\_SUCCESSED on high-order word of wParam means receiving server's response, call function GTXT\_ResponseLineStatus next to get line information; RESPONSE\_TIMEOUT on high-order word of wParam means only part of response received within the time-out limitation, fail to get line information. Function GTXT\_ResponseClientAuth does not contain or contains part of line information.

#### 7.4.8.2 Server Responding

Function prototype: int GTXT\_ResponseLineStatus(LPGTXT\_LINEDATA pLineData)

Description: To get the line information

Parameters:

• pLineData: The line information.

Return value: GTXT\_ERROR\_NOTCONN if there is no connection to server; GTXT\_ERROR\_NOTAUTH before authentication; GTXT\_ERROR\_NOTDATA if no data; GTXT\_ERROR\_OK if succeeds.

Explanation: There may be several lines, if the function returns GTXT\_ERROR\_OK, you should repeat this function until it returns GTXT\_ERROR\_NOTDATA. Calling time refers to the explanation of function GTXT\_RequestLineStatus.

#### 7.4.9 Message Sending

#### 7.4.9.1 Client Requesting

Function Prototype: int GTXT\_RequestSendSMS(unsigned short \*pwDestID, const char \*pszDest, const char \*pszSMS, WORD wLine, unsigned long dwFlag).

Description: To send a request for message sending.

Parameters:

- pwDestID: The corresponding ID to destination. It's the only item when returns result.
- pszDest: Several destinations separated by semicolon ";". The number of destinations must be the same as the number of IDs.
- pszSMS: The content of message
- wLine: Line for sending
- dwFlag: ID generated by clients which marks a specific sending.

Return value: GTXT\_ERROR\_NOTCONN if there is no connection to server; GTXT\_ERROR\_NOTAUTH before authentication; GTXT\_ERROR\_OK if succeeds.

Explanation: GTXT\_ERROR\_OK only means the function's success, but not the sending result of the message. Clients need to wait for WM\_GTXT\_MESSAGE message.

The low-order word of wParam is GTXT\_EVENT\_SEND\_SMS, and the high-order word of wParam is the result. lParam has the dwFlag value set when calling function GTXT\_RequestSendSMS which indicates a specific message sending. RESPONSE\_SUCCESSED on high-order word of wParam means got server's response, call function

GTXT\_ResponseSendSMS to get the result of message sending; RESPONSE\_TIMEOUT on high-order word of wParam means only part of response received within the time-out limitation. As the function can send multiple messages to different destinations at one time, the sending to certain destination from where client does not receive the response is failed.

#### 7.4.9.2 Server Responding

Function Prototype: int GTXT\_ResponseSendSMS(LPGTXT\_SENDRESULT pSendResult, unsigned long dwFlag);

Description: To get the result of message sending

Parameters:

- pSendResult: The result of message sending
- dwFlag:The dwFlag value when calling function GTXT\_RequestSendSM indicates a specific message sending

Return value: GTXT\_ERROR\_NOTCONN if there is no connection to server; GTXT\_ERROR\_NOTAUTH before authentication; GTXT\_ERROR\_NOTDATA if no data; GTXT\_ERROR\_OK if succeeds.

Explanation: You may specify multiple destinations when calling function GTXT RequestSendSMS, and the return is only about one destination every time. If the return value is GTXT\_ERROR\_OK, you should keep on calling this function until you get GTXT ERROR NOTDATA. Calling time refers to the explanation of function GTXT RequestSendSMS.

#### 7.4.10 Message Updating

#### 7.4.10.1 Client Requesting

Function Prototype: int GTXT\_RequestUpdateSMS()

Description: To send a request to update messages

Parameters: none

Return value: GTXT\_ERROR\_NOTCONN if there is no connection to server; GTXT\_ERROR\_NOTAUTH before authentication; GTXT\_ERROR\_PENDING if server has not given out response yet; GTXT\_ERROR\_OK if succeeds.

Explanation: GTXT\_ERROR\_OK only means the function's success, but not you can receive the message. Clients need to wait for WM\_GTXT\_MESSAGE message.

The low-order word of wParam is GTXT\_EVENT\_UPDATE\_SMS, and the high-order word of wParam is the result. RESPONSE\_SUCCESSED on high-order word of wParam means got server's response, call function GTXT\_ResponseUpdateSMS next to get the message; RESPONSE\_TIMEOUT on high-order word of wParam means only part of response received within the time-out limitation, and the update operation fails.

#### 7.4.10.2 Server Responding

Function Prototype: int GTXT\_ResponseUpdateSMS(LPGTXT\_RECVSMS pRecvSMS)

Description: To get updated messages

Parameters:

• pRecvSMS: the structure of received message.

Return value: GTXT\_ERROR\_NOTCONN if there is no connection to server; GTXT\_ERROR\_NOTAUTH before authentication; GTXT\_ERROR\_ NOTDATA if no data; GTXT\_ERROR\_OK if succeeds.

Explanation: If there are multiple updated messages, the function returns GTXT\_ERROR\_OK, you should use this function to get more until it returns GTXT\_ERROR\_NOTDATA, calling time refers to the explanation of function GTXT\_RequestUpdateSMS.

#### 7.4.11 Message Removing

#### 7.4.11.1 Client Requesting

Function prototype: int GTXT\_RequestDeleteSMS(unsigned long dwSMSID).

Description: To send a request to delete a message.

Parameters:

• dwSMSID: Is the message ID got when receiving it.

Return value: GTXT\_ERROR\_NOTCONN if there is no connection to server; GTXT\_ERROR\_NOTAUTH before authentication; GTXT\_ERROR\_OK if succeeds.

Explanation: GTXT\_ERROR\_OK only means the function's success, but not deleting the message successfully. Clients need to wait for WM\_GTXT\_MESSAGE message.

The low-order word of wParam is GTXT\_EVENT\_ DELETE \_SMS, and the high-order word of wParam is the result. lParam contains the message ID. RESPONSE\_SUCCESSED on high-order word of wParam means got server's response, call function GTXT\_ResponseDeleteSMS next to get the result of delete operation; RESPONSE\_TIMEOUT on high-order word of wParam means the server has not given out a response within the time-out limitation, and the delete operation fails.

#### 7.4.11.2 Server Responding

Function Prototype: int GTXT\_ResponseDeleteSMS(unsigned long dwSMSID, BOOL \*pbDelete).

Parameters: Get the result of deleting a message.

Parameters:

- dwSMSID:The message ID of which going to be deleted.
- pbDelete: The result of delete operation. TRUE is success, FALSE is failure.

Return value: GTXT\_ERROR\_NOTCONN if there is no connection with server, GTXT\_ERROR\_NOTAUTH before authentication, GTXT\_ERROR\_NOTDATA if no data, GTXT\_ERROR\_OK if succeeded.

Explanation: Calling time refers to the explanation of function GTXT\_RequestUpdateSMS.

### 7.4.12 New Message Notifying

Function prototype: int GTXT\_ResponseNewSMS(LPGTXT\_RECVSMS pRecvSMS).

Description: To receive a new message.

Parameters:

• pRecvSMS: The structure of received message.

Return value: GTXT\_ERROR\_NOTCONN if there is no connection to server; GTXT\_ERROR\_NOTAUTH before authentication, GTXT\_ERROR\_NOTDATA if no data, GTXT\_ERROR\_OK if succeeds.

Explanation: Client will get WM\_GTXT\_MESSAG message.

The low-order word of wParam is GTXT\_EVENT\_UPDATE\_SMS, and the high-order word of wParam is invalid. Client call function GTXT\_ResponseNewSMS to get new message.

## Appendix A

## **Encoding for SMS Messages**

## A.1 Introduction

An SMS message sent on a GSM system uses 140 octets to transmit the message.

The default GSM alphabet uses characters that are 7 bits in length (described in section A.2, below). GTXT also support characters that are 16 bits in length (described in section A.4 on page 48). The 7 bit alphabet is mandatory for all GSM systems.<sup>12</sup>

The 7 bit characters are defined in ETSI TS 100 900 (GSM 03.38 version 7.2.0 Release 1998) "Alphabets and language-specific information," §6.2. The 16 bit characters are defined in ISO/IEC10646 "Universal Multiple-Octet Coded Character Set (UCS);" UCS2, 16 bit coding.

The 7 bit characters and the 16 bit characters allow you to send and receive messages in any language. Your PC must be properly configured to permit you to input the language of the message you want to create and to display a message you receive in that language. Consult your Windows documentation for further details.<sup>3</sup>

## A.2 7 Bit Characters

### A.2.1 Single Word Characters

The complete character set is shown in the table in section A.3 on page 47. 160 of the 7-bit characters are packed into 140 octets ( $160 \times 7 \div 8 = 140$ ). These characters are referred to as the *default alphabet* and approximate the ASCII character set. The differences between the default alphabet and the ASCII characters are indicated in the table.

<sup>1.</sup> The GSM specification permits the use of an 8 bit character set. Such usage is user defined and has never gained widespread acceptance. It is not supported on GTXT or the GS8.

<sup>2.</sup> Some service providers transmit SMS messages that are only 140 characters. See section 2.2.2 on page 12 for how you can indicate this to GTXT.

<sup>3.</sup> The language of the message is not related to the language that you use to control the GTXT application. Details on how you configure the language of the GTXT application are in section 2.2.1 on page 11.

The key to the table is as follows. The table shows the equivalent ASCII character (for that numeric value) where it differs from the GSM character:



Character 0A hex (10 decimal) is a line feed (LF). Any characters following LF that are to be displayed are shown on the next line of the message, commencing with the first character position.

Character 0D hex (13 decimal) is a carriage return (CR). Any characters following CR that are to be displayed are shown on the current line of the message, commencing with the first character position.

#### A.2.2 Double Word Characters

The escape character (1B hex, 27 decimal), indicates that the character uses two 7-bit words. There are eight of these characters. When you type any of these characters in GTXT, it consumes two characters. These characters are not sent as Unicode characters (see section A.4 on page 48), so the maximum number of characters in the message is still 160 characters.

Hexadecimal	Decimal	Character
1B 14	27 20	^
1B 28	27 40	{
1B 29	27 41	}
1B 2F	27 47	\
1B 3C	27 60	[
1B 3D	27 61	~
1B 3E	27 62	]
1B 65	27 101	€

The double word characters are:

## A.3 Default Alphabet

B7 B6 B5	00	000 001			010 011			100 101			110 111											
Bits		Non-	Lat	in	Numbe		ers and			Up	per	· Ca	se		Lower Case							
B4 B3 B2 B1		Chara	acte	ers			Sym	bol	s			Latin	Ch	ara	cters	S	La	tin	h Ch	arac	; te r	s
00 00	<b>e</b> 0	NUL D 0	10	DLE <b>A</b> 1	E 6 20	spa	<b>се</b> 32	30	0	48	40	i	64	50	Ρ	80	60	ż	' 96	70	р	112
0001	<b>1</b>	SOH 2 1	11	DC <sup>.</sup> – 1	1 7 21	!	33	31	1	49	41	Α	65	51	Q	81	61	a	97	71	q	113
0010	2	STX	12	DC: <b>Ф</b> 1	2 8 22		34	32	2	50	42	В	66	52	R	82	62	b	98	72	r	114
0011	<b>ן</b> 3	ЕТХ <b>4</b> 3	13	DC: <b>F</b>	3 9 23	#	35	33	3	51	43	С	67	53	S	83	63	с	99	73	s	115
0100	<b>é</b> 4	ЕОТ Э	14	DC4 <b>A</b> 2	1 0 24	¤	\$ 36	34	4	52	44	D	68	54	т	84	64	d	100	74	t	116
0101	<b>6</b> 5	ENQ 5	15	ΝΑł Ω 2	( 1 25	%	37	35	5	53	45	Е	69	55	U	85	65	е	101	75	u	117
0110	í 6	ACK ù 6	16	SYN <b>N</b> 2	1 2 26	&	38	36	6	54	46	F	70	56	v	86	66	f	102	76	v	118
0111	7	BEL Ì 7	17	ετε Ψ 2	3 3 27	•	39	37	7	55	47	G	71	57	w	87	67	g	103	77	w	119
1000	<b>6</b> 8	BS 5 8	18	CAΝ Σ 2	1 4 28	(	40	38	8	56	48	Н	72	58	x	88	68	h	104	78	x	120
1001	<b>9</b>	НТ Э	19	EON <b>O</b> 2	Л 5 29	)	41	39	9	57	49	I	73	59	Y	89	69	i	105	79	у	121
1010	L A	. <b>F</b> 10	1A	SUE E	3 6 2A	*	42	ЗA	:	58	4A	J	74	5A	Z	90	6A	j	106	7A	z	122
1011	<b>G</b> B	VT <b>7</b> 11	1B	ES0 esc* 2	; 7 2B	+	43	3B	;	59	4B	к	75	5B	Ä	[ 91	6B	k	107	7B	ä	{ 123
1100	<b>,</b> C	FF 7 12	1C	FS Æ 2	8 2C	,	44	3C	<	60	4C	L	76	5C	Ö	\ 92	6C	I	108	7C	ö	 124
1101	<b>с</b>	<b>R</b> 13	1D	GS <b>æ</b> 2	9 2D	-	45	3D	=	61	4D	М	77	5D	Ñ	] 93	6D	n	109	7D	ñ	} 125
1110	E	SO 14	1E	RS <b>ß</b> 3	0 2E		46	3E	>	62	4E	N	78	5E	Ü	^ 94	6E	n	110	7E	ü	~ 126
1111	F	SI 11 15	1F	US É 3	1 2F	Ι	47	3F	?	63	4F	0	79	5F	§	- 95	6F	0	111	7F	à	)EL 127

## A.4 Unicode Characters (16 bits)

Unicode characters consume two complete octets, so the total size of the SMS message is limited to 70 characters (140  $\div$  2). If the message comprises characters mostly from the default (7-bit) alphabet, the entire message is sent as Unicode if there is at least one Unicode character.<sup>1</sup>

The Unicode characters permit the transmission of special symbols, Chinese, Japanese, and Korean characters (sometimes called "Kanji" characters), Hebrew and Arabic characters, Greek, Russian, and other European characters, and those for most languages in the world.<sup>2</sup>

## A.5 Long Messages

Each SMS occupies 140 octets and as described above, that permits the transmission of nominally 160 characters of the default alphabet or 70 Unicode characters.

The GSM system sends messages that are longer than this length in multiple SMS messages. There are six octets used for overhead to link the two messages together. A mobile phone will take these multiple messages and display them as a single (large) message.

The six octets reduce the size of each message to 153 7-bit characters and 64 Unicode characters. When you are typing a long message in GTXT, the counter that indicates the size of the message displays the following:

Number of 7-bit characters in message	Display of counter on GTXT
0	SMS #1: 160 characters left
1	SMS #1: 159 characters left
160	SMS #1: 0 characters left
161	SMS #2: 145 characters left
206	SMS #2: 0 characters left
207	SMS #2: 152 characters left

There is a jump after the 161st character is typed to 145 characters remaining. This is because after you type character 161, GTXT "moves" eight characters from the first SMS to the second SMS. The second SMS can accommodate 153 characters so this leaves 153 - 8 = 145 characters remaining in the second SMS.

If you type a character that consumes two 7-bit spaces (as described in section A.2.2 on page 46) in the last character position of an SMS, GTXT will move both 7-bit words into the next message. The counter will then jump to show only 144 characters are remaining in that message.

<sup>1.</sup> The special double word characters described in section A.2.2 on page 46 do not force the message to be sent as Unicode.

<sup>2.</sup> Under Windows, the Unicode characters are encoded in two, three, or four bytes. The MSB of the first byte is set to one to distinguish the character from ASCII. An ASCII character has the MSB of the byte set to zero.

## Appendix B

## Creating an Operator User on the GS8

The creation of a user of GTXT to be the operator becomes necessary when you want to collect all incoming SMS messages through GTXT.

By default, the GS8 forwards an incoming SMS message to a GTXT user only if **all** of the following conditions are true:

- the GTXT user has previously sent a SMS message to this phone number in the last "x" seconds (where "x" is defined in **SMS | Configuration | Return Message**)
- no other GTXT user has sent a message to this phone number before this SMS is received
- the GTXT user is logged in

The SMS configuration window on GTXT is shown below:

<b>Z</b> ZZ	ED-3				
Monitor	Line Settings	Call Settings	SMS	Administration	Maintenance
2009-01-	08 21:01	Setting			
Route GTXT Client		бтхт	GTYT Acce		Enabled C Disabled
Configurate Send Test M	Message		Listen Por	t	16200
Received M	essages		Update Re	equest	Accept C Reject
		Return Message	If rece	ived within 180	seconds, return to sender
		Save Message	Which mea	ssages <mark>will be sa</mark> v	ed in GS8 when they can't be forwarded to destination
			C Messa	ges From GSM	
			C Messa	ges From SIP	
			All Mes	sages	
		Set			

If the GS8 determines that an incoming SMS message is not a reply message to any GTXT user, it will try to route the message according routing plan defined in the **SMS** | **Route** window. If the GS8 can not route a SMS message according to the routing plan defined in **SMS** | **Route**, it will store the message on the GS8 in the **SMS** | **Received Message** window. In this case, the only way to retrieve the message is through the administration UI.

Under most applications, it is not desirable to store any SMS messages on the GS8. Therefore, it is necessary to configure a user of GTXT that acts as the operator. This configuration is done on the GS8 and ensure that such a GTXT user will receive all messages that are not replies or that are not forwarded to any other user. To accomplish this, you need to modify the routing table in the **SMS** | **Route** window to route all the messages to a GTXT user. In the example shown below, a rule is defined to forward all incoming messages to GTXT user 16.

Monitor Line Setti	ngs Call Settings	SMS	Administration Mai	intenance				admin	Logout
009-01-08 21:01	Description	Source		Destination		Caller Pattern	Callee Pattern		
ammon Tasks	diperator	GSM/ALL		GTXT/16			ALL		
	Nick's cel	GSM/ALL		GTXT/16		510381.	ALL		
Route	Suz	GSM/ALL		GTXT/16		510258.	ALL		
GTXT Client	ray	GSM/ALL		GTXT/16		510676.	ALL		
Send Test Message	ben	GSM/ALL		GTXT/16		510213.	ALL		
Received Messages	self	GSM/ALL		GTXT/13		14085858533	ALL		
	Description		GSM I AL	L M sloti					
	Source Line		GSM IV AL	L M slot	□ 10	<b>D</b> 11	L 15		
	Source Line Caller Pattern Callee Pattern		GSM IV AL	L M slot	⊃ □ 10	Du	□ 12		
	Source Line Caller Pattern Callee Pattern Destination Line		GSM C st GSM C st IP 14	L F sloti	C 11	C 11	□ 12 C 14		
	Caller Pattern Caller Pattern Callee Pattern Destination Line		GSM F AL IP AL IP 14 GSM C al IP C 1 IP C 1 GTXT C 4	L K slot1 L 1 C 10 C 5 3 C 15	C 11 C 11 C 6 € 16	C 12 C 7 C 17	C 14 C 9		

## Appendix C

## Acronyms

Α	Ampere
AA	auto attendant
ac	alternating current
ACD	automatic call distributor
ACL	access control list
AES	advanced encryption standard
ALG	application level gateway
ANI	automatic number identification
ANM	answer message
API	application programming interface
ARP	address resolution protocol
ASCII	American standard code for information interchange
B2BUA	back to back user agent
b	bit
b B	bit byte
b B b/s	bit byte bits per second
b B b/s B/s	bit byte bits per second bytes per second
b B b/s BHCA	bit byte bits per second bytes per second busy hour call attempts
b B b/s BHCA CAS	bit byte bits per second bytes per second busy hour call attempts channel associated signaling
b B b/s B/s BHCA CAS CDR	bit byte bits per second bytes per second busy hour call attempts channel associated signaling call detail recording
b B b/s B/s BHCA CAS CDR CHAP	bit byte bits per second bytes per second busy hour call attempts channel associated signaling call detail recording challenge handshake authentication protocol
b B b/s B/s BHCA CAS CDR CHAP CID	bit byte bits per second bytes per second busy hour call attempts channel associated signaling call detail recording challenge handshake authentication protocol caller identification
b B b/s BHCA CAS CDR CHAP CID CLID	bitbytebits per secondbytes per secondbusy hour call attemptschannel associated signalingcall detail recordingchallenge handshake authentication protocolcaller identificationcalling line identification
b B b/s B/s BHCA CAS CDR CHAP CID CLID CND	bit byte bits per second bytes per second busy hour call attempts channel associated signaling call detail recording challenge handshake authentication protocol caller identification calling line identification
b         B         b/s         B/s         BHCA         CAS         CDR         CHAP         CID         CLID         CND         CNG	bit byte bits per second bytes per second busy hour call attempts channel associated signaling call detail recording challenge handshake authentication protocol caller identification calling line identification

CODEC	coder decoder
CoS	class of service
CPE	customer premises equipment
CRM	customer relations management
CSID	fax sender ID
CSV	comma separated variable
СТ	computer telephony
CTI	computer telephony integration
dB	decibel
dc	direct current
DHCP	dynamic host configuration protocol
DID	direct inward dial
DMZ	de-militarized zone
DND	do not disturb
DN	domain name
DNS	domain name service
DSCP	differentiated services code point
DSP	digital signal processor
DTMF	dual tone multi-frequency
E1	digital signal, level 1, specified by ITU-T (2.048 Mb/s)
EMC	electromagnetic compatibility
EMI	electromagnetic interference
ENUM	E.164 number
EPABX	electronic private automatic branch exchange
ESD	electrostatic discharge
ETSI	European Telecommunication Standards Institute
FCC	Federal Communications Commission
ft	feet
FQDN	fully qualified domain name
FTP	file transfer protocol
FXO	foreign exchange office (analog subscriber side interface)
FXS	foreign exchange subscriber (analog exchange side interface)
GB	giga-byte (1,073,741,824 bytes)
Gb/s	giga-bits per second
GMT	Greenwich Mean Time

GND	ground
GSM	global system for mobile communications
GUI	graphical user interface
h	hour(s)
HA	high availability
HDLC	high level datalink control
HTML	hyper text markup language
HTTP	hyper text transport protocol
Hz	Hertz
I/O	input/output
ICMP	Internet control message protocol
IE	information element
IETF	Internet engineering task force
IGMP	Internet group management protocol
IGP	interior gateway protocol
in	inch
IP	Internet protocol
IPSec	IP security
IPv4	Internet protocol version 4
IPv6	Internet protocol version 6
ISDN	integrated services digital network
ISO	International Organization for Standardization
ISP	Internet service provider
ITSP	Internet telephony service provider
ITU	International Telecommunication Union
IVR	interactive voice response
К	1024
k	kilo (1000)
КВ	kilo-byte (1024 bytes)
kb/s	kilo-bits per second
kHz	kilo-Hertz
km	kilometer
kW	kilo-Watt
L2TP	layer 2 tunneling protocol
LAN	local area network

LCD	liquid crystal display
LCR	least cost routing
LDAP	lightweight directory access protocol
LED	light emitting diode
LSB	least significant bit
m	meter
mA	milli-Ampere
MB	mega-byte (1,048,576 bytes)
Mb/s	million bits per second
MGCP	media gateway control protocol
MHz	mega-Hertz
MIME	multipart Internet mail extension
mm	millimeter
MPLS	multi protocol label switching
ms	milli-second
MSB	most significant bit
mW	milli-Watt
MWI	message waiting indicator
n/a	not applicable
NANP	North American numbering plan
NAT	network address translator
ns	nanosecond
NTP	network time protocol
OAM&P	operations, administration, maintenance, and provisioning
OSI	open systems interconnection
PABX	private automatic branch exchange
PAP	password authentication protocol
PBX	private branch exchange
PC	personal computer
РСВ	printed circuit board
РСМ	pulse code modulation
PDF	portable document format
РНВ	per-hop behavior
PIN	personal information number
POTS	plain old telephone service

PPP	point to point protocol
PRA	primary rate access
PRI	primary rate interface
PSTN	public switched telephone network
PSU	power supply unit
PTT	post, telephone, and telegraph administration
QoS	quality of service
RAM	random access memory
RARP	reverse address resolution protocol
RFC	request for comments
RMA	return material authorization
ROM	read only memory
ROW	rest of the world
RSVP	resource reservation protocol
RTC	real time clock
RTF	rich text format
RTP	real time transport protocol
RU	rack unit (1.75 in, 44.45 mm)
Rx	receive
RxD	receive data (or received data)
S	second
SDP	session description protocol
SE system	system exchange (either the SE150 or the SE500)
SEME	system exchange for me
SIP	session initiation protocol
SLA	service level agreement
SMDI	simplified message desk interface
SME	small or medium (size) enterprise
SMTP	simple mail transfer protocol
SNMP	simple network management protocol
SNTP	simple network time protocol
TAPI	telephony application programming interface
ТСР	transmission control protocol
TCP/IP	transmission control protocol / Internet protocol
TFTP	thin file transfer protocol

TIFF	tagged image file format
TOS	type of service
TTL	time to live
Tx	transmit
TxD	transmit data (or transmitted data)
U	see RU
UAC	user agent client
UAS	user agent server
UDP	user datagram protocol
UI	user interface
UL	Underwriter's Laboratory
UM	unified messaging
UPS	uninterruptible power supply
URL	universal reference locator
URI	uniform resource identifier
USOC	universal service ordering code
UTC	coordinated universal time
V	Volt
Vac	Volts with alternating current
VAD	voice activity detection
Vdc	Volts with direct current
VF	voice frequency
VLAN	virtual local area network
VM	voice mail
VoIP	voice over IP
VPN	virtual private network
Vpp	Volts peak to peak
W	Watt
WAN	wide area network

## Appendix D

## Glossary

**10BaseT.** A physical media specified by the IEEE 802.3 standard for supporting Ethernet with a maximum transmission rate of 10 Mb/s. 10BaseT consists of copper twisted-pair cable normally used for wiring ordinary telephones. Ethernet is a common technology used for connecting computers into a local area network (LAN).

**100BaseT.** A local area network transmission standard that supports a data rate of 100 Mb/s. Also known as *Fast Ethernet*; similar in function to 10BaseT.

*Address Resolution Protocol (ARP).* Defines the process of mapping an Internet Protocol address (IP address) to a physical machine address recognized in the local network. An *ARP table* maintains the correlation between each MAC address and its corresponding IP address within a network. ARP provides the protocol rules for defining this correlation and converting addresses in both directions. ARP is described by RFC 826.

*Advanced Intelligent Network (AIN).* A telephone network architecture that separates service logic from switching equipment. AIN encourages competition among service providers by allowing the addition of new services without requiring the re-design of existing switches.

*Authentication.* The process of ensuring that digital data transmissions are delivered to the intended receiver. Authentication also assures the receiver of the integrity of the message content and source. The simplest authentication method requires a user name and password to gain access to a specified account. Authentication protocols can be based on secret-key encryption or on public-key systems using digital signatures.

*Authorization.* The process of verifying the identity of a person or device. Authentication is commonly performed through logon passwords; knowledge of the password is assumed to guarantee that the user authenticity. Internet business and many other transactions may require a more stringent authentication process, such as the use of digital certificates issued and verified by a Certificate Authority as part of a public key infrastructure. Logically, authentication precedes authorization, although they may often seem to be combined.

*Bridge.* A device that connects network segments that use the same protocol, such as Ethernet. A bridge forwards traffic between network segments on the basis of data link layer information; these segments would require a common network layer address.

Called Party. The person or device that receives a phone call or data transmission.

Calling Party. The person or device that initiates a phone call or data transmission.

*Contact.* A device where a user can be reached.

*Decryption.* Decryption is the process of converting encrypted data to its original form. See Encryption.

**De-Militarized Zone (DMZ).** A computer host or small network placed between a company's private network and the outside public network to prevent outside users from gaining direct access to a server that contains company data. The term is derived from the geographic area between two opponents where fighting is prohibited. A DMZ is an optional, more secure approach to a firewall and effectively acts as a proxy server as well.

*Direct Inward Dialling (DID).* DID is a service that allows users that are connected to a common server (such as a media exchange or a PBX) to receive calls from sources external to the server without the intervention of an auto attendant or operator. Under DID, each user is assigned a unique telephone number, as opposed to the typical PBX setup that assigns extensions that are based on a common telephone number.

*Domain Name System (DNS).* Defines the manner that the Internet translates names of network nodes into addresses. SIP uses DNS to resolve the host names of endpoints to IP addresses.

*Dynamic Host Configuration Protocol (DHCP).* A communication protocol that defines a method where network administrators manage and automate Internet Protocol (IP) address assignment within an enterprise network. DHCP allows you to move network devices from one subnet to another without administrative attention. If using DHCP, you can connect IP phones to the network and become operational without having to manually assign an IP address and additional network parameters. The ZIP 4x4 phone complies with the DHCP specifications documented in RFC 2131 and are DHCP-enabled by default.

Advanced Encryption Standard (AES). Also known as Rijndael, this is a block cipher adopted as an encryption standard by the U.S. government and developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen. It has been analyzed extensively and is now used widely worldwide as a replacement for its predecessors the Data Encryption Standard (DES) and the triple DES (3DES). AES was announced by the NIST on 26 November 2001 after a 5-year standardization process and became a standard on 26 May 2002. It is one of the most popular algorithms used in symmetric key cryptography and is acceptable by the US government for classified information up to Top Secret.

*Encryption.* The process of converting data into a form that can be read only by the intended receiver. Decryption is the process of converting encrypted data to its original form. Traditional encryption schemes utilize the same key to encrypt and decrypt data. Public-key encryption schemes require two keys: a public key, which anyone may use, and a corresponding private key, which is possessed only by the person who created it. With this method, anyone may send a message encrypted with the owner's public key, but only the owner has the private key necessary to decrypt it. PGP (Pretty Good Privacy) and DES (Data Encryption Standard) are two of the most popular public-key encryption schemes.

*Ethernet.* The most widely-installed local area network (LAN) technology. Originally developed at the Xerox Corporation Palo Alto Research Center, Ethernet is specified in the IEEE 802.3 standard. Ethernet is a best-effort delivery system that uses CSMA/CD technology. Ethernet can be run over a variety of physical media, including coaxial, twisted pair, and fiber optics.

*Firewall.* A device, located at a network gateway server, that protects the resources of a private network from external entities. Typically comprises a set of related programs or a dedicated computer equipped with such security features as logging, reporting, alarms, and a control mechanism.

*File Transfer Protocol (FTP).* An application layer standard Internet protocol that uses the TCP/IP protocols to exchange files between computers on the internet. Commonly used to transfer web page files from the creator to a server or to download programs and other files from a computer to other servers. Described by RFC 959.

*Foreign Exchange Office (FXO).* An analog interface that connects to the Public Switched Telephone Network (PSTN) central office and is the interface offered on a standard telephone.

*Foreign Exchange Station (FXS).* An analog interface that connects directly to a standard telephone and supplies ring, voltage, and dial tone.

FTP Account. An SE system identifier that points to an FTP address.

*Fully Qualified Domain Name (FQDN).* The portion of an Internet URL that identifies the server program that an Internet request addresses. The FQDN includes the second-level domain name (such as Zed-3.com) and any other levels (such as www.Zed-3.com).

*Gateway.* A network point that serves as an entrance to another network. Computers that control traffic within an enterprise's network or at the local Internet Service Provider (ISP) are gateways. IP datagrams are transferred from network to network through gateways until it reaches its final destination. See Router.

*Hub*. A hub network topology consists of a backbone (main circuit) to which a number of outgoing lines can be attached, each providing at least one connection port for attaching devices. As a network product, a hub may include a group of modem cards for dial-in users, a gateway card for connections to a local area network, and a line connection.

*Hypertext Transfer Protocol (HTTP).* An application layer protocol that defines a set of rules for exchanging files (text, images, sound, video, and other multimedia files) on the Internet. Described by RFC 2068.

*Incoming call.* A call originated by a source that is external to the enterprise.

*Internet.* A worldwide computer network system in which users at any one computer can, with permission, exchange information from any other computer and sometimes talk directly to users at other computers; also known as the "Net." Originally designed by the Advanced Research Projects Agency (ARPA) of the U.S. Defense Department in 1969 so that a communication signal could withstand a nuclear war and serve military institutions worldwide. First known as the ARPAnet, the internet has evolved into public, cooperative, and self-sustaining facility accessible to billions of people worldwide.

*Internet Control Message Protocol (ICMP).* A message control and error-reporting protocol between a host server and an Internet gateway that enables hosts to send error or control messages to other hosts. ICMP is an integral part of IP and must be implemented by every IP module. Instances for sending an ICMP message include datagram processing errors, the datagram cannot reach its destination, or when the gateway has insufficient buffering capacity to forward a datagram. The ZIP 4x4 phone supports ICMP as documented in RFC 792.

*Internet Engineering Task Force (IETF).* The organization that defines standard Internet operating protocols such as TCP/IP. The IETF is supervised by the Internet Society Internet Architecture Board (IAB). IETF members are drawn from the Internet Society's individual and organization membership. Standards are expressed in the form of Requests for Comments (RFCs).

*Internet Key Exchange (IKE).* The method for exchanging encryption and authentication keys over an unsecured medium, such as the Internet.

*Internet Protocol (IP).* A network layer protocol that sends datagram packets between Internet nodes. IP is a connection-less protocol, implying that there is no continuing connection between communicating endpoints. IP provides addressing, type-of-service (ToS) specification, security, fragmentation and reassembly features. The most widely used IP version is Internet Protocol Version 4 (IPv4). The ZIP 4x4 phone supports IP as defined in RFC 791.

*Internet Protocol Version 6 (IPv6).* The latest level of the IP that is included in many projects, including the major computer operating systems. The most obvious improvement in IPv6 over IPv4 is the lengthening of IP addresses from 32 bits to 128 bits. All servers that support IPv6 will also support Internet Protocol Version 4.

*Intranet.* A restricted-access network that works like the Web, but isn't on it. Usually owned and managed by a corporation, an intranet enables a company to share its resources with its employees without making available confidential information to everyone with Internet access.

*IP Address (version 4).* A 32-bit number that identifies each sender or receiver of information sent across the internet. An IP address has two parts: the network identifier and the identifier of a specific device on the network. On the Internet itself – between the router that moves packets between points along the route – only the network part of the address is examined.

*IP Security (IPSec)*. Security standard produced by the Internet Engineering Task Force (IETF); a protocol suite that provides all necessary elements for secure communications-authentication, integrity, and confidentiality-and makes key exchange practical even in larger networks.

*Jitter.* Jitter is the deviation in some aspect of the pulses in a high-frequency digital signal or the period frequency displacement of the signal from its ideal location. The SE system utilizes Jitter Buffers to compensate for jitter.

*Key Management.* The management and handling private keys used for signing or encryption. The only reasonable way to protect the integrity and privacy of information is to rely upon the secure use of these keys, including the activities of selection, exchange, storage, certification, expiration, revocation, changing, and transmission. Most of the work in managing information security systems lies in the key management.

*Lightweight Directory Access Protocol (LDAP).* A software protocol that enables anyone to locate organizations, individuals, and other resources such as files and devices in a network, whether on the public Internet or on a corporate intranet. LDAP is a "lightweight" (smaller amount of code) version of Directory Access Protocol (DAP), which is part of X.500, a standard for directory services in a network. LDAP is lighter because, in its initial version, it did not include security features. Described by RFC 2251.

*Load balancing.* The mapping (or re-mapping) of work to processors, with the intent of improving the efficiency of a concurrent computation.

*Local Area Network (LAN).* A group of computers and associated devices that share a common communications line and the resources of a single processor or server within a limited geographic area, such as an office building. 10BaseT Ethernet is the most commonly used form of LAN. A hardware device called a hub serves as the common wiring point, enabling data to be sent from one machine to another over the network. LANs are typically limited to distances of less than 500 meters and provide low-cost, high-bandwidth networking capabilities within a small geographical area.

*Media Access Control (MAC) Address.* A hardware number that uniquely identifies a computer or other device. Within an Ethernet configuration, the MAC address is a 6-octet address assigned to the network interface card. When your computer is connected to the Internet, a correspondence table relates your IP address to your computer's physical (MAC) address on the LAN. The MAC address is used by the Media Access Control sublayer of the Datalink Control (DLC) protocol layer. Each physical device type has a different MAC sublayer.

*Network Address Translation (NAT).* A standard for translating a secure IP address used within one network to a different IP address known from another network. In addition to providing Internet access for trusted networks with privately assigned IP addresses, NAT conserves on the number of global IP addresses required by a network because each machine in a network does not require a registered IP address.

*Network Interface Card (NIC).* A circuit board or card that is installed in a computer for the purpose of connecting the computer to a network. Personal computers on a LAN usually contain a NIC designed for a specific LAN transmission technology, such as Ethernet.

*Network Time Protocol (NTP).* A protocol that synchronizes computer clocks on an IP network. Described by RFC 1305.

*Per-Hop Behavior.* The differential treatment an individual packet receives while being routed through a network, as implemented by queue service or queue management disciplines. These per-hop behaviors are useful and required in network nodes to deliver differentiated treatment of packets regardless of end-to-end construction or intra-domain services. Per-Hop behavior is discussed in RFC 2474 and Per-Hop Identification codes are defined in RFC 3140.

*Point-to-Point Protocol (PPP).* A protocol that defines a communication method between two computers using a serial interface. Typically used to connect a personal computer to a server through a phone line.

*Proxy Server.* A server that acts as an intermediary between a workstation and the Internet to provide a caching service and ensure security and administrative control for the enterprise. The proxy server is invisible to the workstation; all Internet requests and returned responses involving the workstation appear to be directly with the addressed Internet server.

*Public Switched Telephone Network (PSTN).* The world-wide collection of voice-oriented public telephone networks. Also referred to as plain old telephone service (POTS).

*Quality of Service (QoS).* The concept that transmission rates, error rate, and other characteristics over a network or the Internet can be quantified, improved, and guaranteed (to a certain extent) in advance. QoS is particularly concerned with the continuous transmission of high-bandwidth video and multimedia data.

*Real-Time Transport Control Protocol (RTCP).* The protocol companion to RTP that provides error, session control, and identification data about a transport session. Described by RFC 1889.

*Real-Time Transport Protocol (RTP).* An Internet protocol standard that specifies a method for programs to manage the real-time transmission of multimedia data over unicast or multicast network services. RTP combines its data transport with an control protocol (RTCP), which makes it possible to monitor data delivery for large multicast networks. The ZIP 4x4 phone supports RTP as a media channel for voice and video applications as described in RFC 1889.

*Repeater.* A device that receives a digital signal on a transmission medium and regenerates the signal for the next leg of the medium. Repeaters overcome attenuation losses caused by free-space electromagnetic-field divergence or cable loss over electromagnetic media. A series of repeaters facilitates the extension of a signal over long distances.

*Request for Comments (RFC).* A formal document issued by the Internet Engineering Task Force. Some RFCs are informational in natural while others become Internet standards. RFCs are the result of committee drafting and subsequent review by interested parties. No further comments or changes to an RFC are permitted once it becomes an Internet standard; changes to internet standards are enacted through subsequent RFCs that supersede or elaborate on all or parts of existing RFCs.

*Reverse Address Resolution Protocol (RARP).* A protocol by which a physical machine in a local area network can request its IP address from a gateway server's Address Resolution Protocol table or cache. RARP is described in RFC 903.

*RJ-45.* A single-line digital transmission interface. Resembling a standard phone connector, an RJ-45 connector is twice as wide (with eight wires) and is used for connecting computers to local area networks (LANs) or phones with multiple lines.

*Router.* A device that determines the next network point to which a data packet should be sent on the to its final destination. Routers also act as traffic cops, allowing only authorized machines to transmit data into the local network so that private information can remain secure. In addition to supporting these dial-in and leased connections, routers also handle errors, maintain network usage statistics, and handle security issues.

*Routing Information Protocol (RIP).* A widely used protocol for managing router information within a self-contained network, such as a corporate LAN. A gateway host sends its entire routing table to it closest neighbor every 30 seconds, which in turn passes the table to its neighbor; this process is repeated until all hosts in the network have the same information. RIP is described by RFC 1058.

Server. A computer program or device that provides services to other computers.

*Session Description Protocol (SDP).* An ASCII-based protocol that describes multimedia sessions and their related scheduling information, including information transport session participant port numbers and contact addresses. The ZIP 4x4 phone uses SDP for session descriptions as documented in RFC 2327.

*Session Initiated Protocol (SIP).* An Internet standard protocol that defines a method of initiating an interactive user session involving multimedia elements, such as voice, chat, gaming, video, and virtual reality. SIP is a request-response protocol that deals with requests from clients and responses from servers through any transport protocol, such as UDP or TCP. Described by RFC 3261, SIP can establish, modify, or terminate multimedia sessions or Internet telephony calls.

*Simple Network Management Protocol (SNMP).* The protocol that governs network management and the monitoring of network devices and their functions. SNMP is described formally in RFC 1157 and in a number of related RFCs.

*Simple Network Time Protocol (SNTP).* A protocol that synchronizes computer clocks on an IP network. Similar to NTP with fewer features. Some IP phones use SNTP for their date and time synchronization functions. Described by RFC 2030.

SIP Endpoint. An internet host that understands the SIP protocol.

*SIP Registrar.* A SIP Registrar is a UAS that responds to REGISTER requests and maintains a list of bindings that are accessible to proxy servers and redirect servers within its administrative domain.

*SIP Server.* A network device that performs special functions at the request of SIP endpoints. Servers typically act in response to SIP endpoint requests, but can also initiate functions on their own. RFC 3261 defines three types of SIP servers: SIP Proxy servers, Redirect servers, and Registrar servers.

*Subnet Mask.* A number that, when applied to an IP address, can identify the subnetwork where the address resides. For example, within a class B network, a subnet mask of 255.255.255.0 specifies that the first two portions of the decimal dot format are the network number, while the third portion is a subnet number. The fourth portion is the host number. A subnet mask of 255.255.0.0 species a class B network that does not have a subnet.

*Switch.* A network device that selects a path or circuit for sending a unit of data to its next destination. A switch may also include the function of the router. A switch is generally a simpler and faster mechanism than a router, which requires knowledge about the network and how to determine the route. A switch is usually associated with layer 2 of the OSI communications model.

*Transmission Control Protocol (TCP).* A set of communications protocols that, when used with Internet Protocol (IP), support peer-to-peer connectivity functions for both local and wide area networks. TCP/IP is a communications protocol which allows computers with different operating systems to communicate with each other and controls how data is transferred between computers on the Internet.While IP handles the actual delivery of data, TCP tracks the data packets into which a message is divided for efficient routing through the internet. The ZIP 4x4 phone supports TCP as described by RFC 793.

*Trivial File Transfer Protocol (TFTP).* An Internet software utility for transferring files that, while simpler to use than FTP, provides fewer features. TFTP is used where user authentication and directory visibility are not required. The ZIP 4x4 phone uses TFTP to download configuration files and software updates from the TFTP Server, as described in RFC 1350.

*Trunk Port.* A device that allows a switch to bundle traffic from several VLANs through a single physical port, sorting the various packets by the VLAN identifier (VID) in their frame headers.

*Trusted Network.* Networks inside your network security perimeter. Only known and approved sources are allowed access to a trusted network.

*Tunnelling.* The transmission of data intended for use only within a private network through a public network in such a way that the routing nodes in the public network are unaware that the transmission is part of a private network. Tunneling is generally done by encapsulating the private network data and protocol information within the public network transmission units so that the private network protocol information appears to the public network as data. Tunneling allows the use of the Internet, which is a public network, to convey data on behalf of a private network.

**Universal Resource Locator (URL).** A standard method of specifying the location of an internet resource. Also referred to as a location or address, URLs specify the location of files on servers. A general URL has the syntax protocol://address. For example, http://www.Zed-3.com/index.htm specifies that the protocol is http and the address is www.Zed-3.com/ index.htm.

*Unshielded Twisted Pair (UTP).* Also known as 10BaseT. This is the standard cabling used for telephone lines. It is also used for Ethernet connections.

**Untrusted Network.** Networks that are outside of your security perimeter; Private and shared networks over which you have no control over the administration or security policies. Firewalls deal with the problem of communicating with these networks while protecting your trusted network.

*User Agent.* A SIP logical entity that can act as both a user agent client (UAC) and user agent server (UAS). The role of UAC and UAS, as well as proxy and redirect servers, are defined on a transaction-by-transaction basis. For example, the user agent initiating a call acts as a UAC when sending the initial INVITE request and as a UAS when receiving a BYE request from the calling station. Similarly, the same software can act as a proxy server for one request and as a redirect server for the next request.

*User Agent Client.* A SIP logical entity that creates a new request, and then uses the client transaction state machinery to send it. The role of UAC lasts only for the duration of that transaction; if a piece of software initiates a request, it acts as a UAC for the duration of that transaction. If it receives a request at a later time, it assumes the role of a user agent server to process that transaction.

*User Agent Server.* A SIP logical entity that generates a response to a SIP request. The response accepts, rejects, or redirects the request. This role lasts only for the duration of that transaction; if a piece of software responds to a request, it acts as a UAS for the duration of that transaction. If it generates a request later, it assumes the role of a user agent client to process that transaction.

**User Datagram Protocol (UDP).** A communications protocol that offers a limited amount of service when messages are exchanged between computers in a network that uses IP. An alternative to TCP, UDP uses the IP to transfer a data unit between computers without dividing it into packets and reassembling it at the other end. Because UDP does not provide the sequencing of packets, the application program that uses UDP must be able to verify that the entire message has arrived in the proper order. UDP is in the Transport Layer (or Layer 4) of the OSI communication model. The ZIP 4x4 phone supports UDP as defined in RFC 768 for SIP signalling.

*Virtual Local Area Network (VLAN).* A Local Area Network that maps workstations on a logical basis (such as department or primary application) rather than by physical location. VLAN members are not identified by their location on a physical subnetwork but through the use of tags in the frame headers of their transmitted data. VLANs are described in the IEEE 802.1Q standard.

*Virtual Private Network (VPN).* A method of using the Internet to provide remote offices or individual users with secure access to their organization's network. A virtual private network can be contrasted with an expensive system of owned or leased lines that can only be used by one organization. The goal of a VPN is to provide an organization with the same capabilities, but at a much lower cost. VPNs are possible because of technologies and standards such as tunneling, screening, encryption, and IPSec.

*Wide Area Network (WAN).* A geographically dispersed telecom network. Although a WAN may be privately owned, the term usually implies the inclusion of public networks.

## lndex

## Symbols

.NET framework		•																		.6	5	
----------------	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	----	---	--

## Numerics

16 bit characters										.4	8
7 bit characters										.4	5

## A

acronyms
active tab
add new tab8
addresses, see contacts
alphabet, default for messages 45, 47
API, see software API
application notes9
Arabic characters
ASCII characters
,

## В

buttons
close
contacts
filter
hide
link
schedule
send
show
Z button9

## С

characters
16 bit
7 bit
double word46
Unicode
see also languages
Chinese
close
button9
caution
move to system trav 7
close tab 8
configuration
description 11–14
FODN 12
CS8 user name and nassword 12
ID address 12
IP address
licensing
name of GS813
port address
user name and password14
Z button

connect to database
connection status
authenticating 13
connected14
login failure14
resolving domain name13
unable to resolve domain name .13
contacts
button
distinction from recipient15
file locations
hide9
local file15
Outlook folder15
panel9
select folder8
counter, SMS characters9, 19, 48

## D

database, see linking a message	
default alphabet 45, 47	7
documentation, see manual	
double word characters46	5

### Ε

encoding for SMS messages 45-48
English
exit GTXT9

### F

failure, database write back
file locations
filter button
folder
file locations
select button
FQDN
configuration
definition

## G

## Η

Hebrew characters
nide button
nide GTXT7

history pano	С
mistory pane	 5

## I

inactive group tab8
inactive tab
installation6
IP address12

## J

Japanese characters	
---------------------	--

## Κ

## L

## Μ

manual
acronyms 51–56
colophon
feedback
related documentation9
message pane9
messages
see receiving messages
see sending messages
see SMS messages
Microsoft
.NET framework6

Outlook see Outlook

### Ν

number field, add recipient
-----------------------------

## 0

operator
enable for startup messages13
Outlook
folder selection

## Ρ

password	13
phone numbers, see contacts	
port address	13
preview tab (link to database)	28
product ID	14
pull down to show tabs	.8

## R

receiving messages
retrieval at startup13
see also SMS messages
recipient
distinction from contact15
pane9
selecting
regional options11
Russian11, 19, 48

## S

sales offices
schedule button
scheduling a message
caution regarding closing GTXT7
description
screen shot
configure window11
link window
main GTXT window8
schedule window
SDK, see software API
search field9
select folder
send button9
send tab (link to database)
sending messages
character counter9, 19, 48
immediately
linked to a database
multiple SMSs 19, 48
on a schedule
send button
see also linking messages

see also SMS messages
setup6
show button
slot number, on GS813
SMS messages
Chinese
default alphabet
encoding and languages 45–48
message size 12, 45
text messages6
Unicode characters
see also receiving messages
see also sending messages
software API
Spanish
success, database write back
system tray7
5 5

## Т

tabs
active8
add new8
close
group8
inactive
pull down8
text messages, see SMS messages
title bar
hide (X)7
indication8

## U

Unicode ch	aracters	 			 	19,	48
user name		 			 		.13

### W

web site addresses5	,
white papers	)
write back	,

#### Z Zt

Z button					
configuration	 	 	 	 	.11
location	 	 	 	 	9