

Ozeki Message Server 6 Product Guide

Last updated: 16th. March. 2006.
Number of pages: 185

Table of contents

TABLE OF CONTENTS	1
INTRODUCTION	4
Terms of use	5
Summary of changes	6
About Ozeki Message Server 6	7
Package contents	8
Ozeki Message Server	9
Ozeki Message Server Monitor	10
Ozeki Message Server Manager	11
OZEKI Excel SMS Client	12
Outlook SMS Plugin	13
Ozeki SMS Client	13
SMS TECHNOLOGY	15
The SMS network entities	15
SMS Network Architecture and Internal Protocols	17
SMS message types	18
SMS messaging with a mobile phone attached to the PC	20
Connect your phone to the PC with an RS232 serial cable	21
Connect your phone to the PC with an USB data cable	22
Connect your phone to the PC with Infrared (IR)	23
Connect your phone to the PC with Bluetooth (BT)	24
IP SMS technology	25
Common protocols - UCP/EMI	26
Common protocols - SMPP	27
Common protocols - CIMD2	29
Common protocols - Colour SMS	29
Common protocols - Premium rated SMS services	30
VPN configuration on Windows 2003	31
Service Provider Information	32
PRODUCT FEATURES	33
Ozeki Message Server in the Office	33
Basic features	34
Multi-user architecture	36
Addressbook	37
How IT Professionals Use the Software	38
The envelope system	38
Link and message management	39
Message routing	39
INSTALLATION / ADMINISTRATION GUIDE	43

Installing the server	43
Hardware and software requirements	43
Installation steps	44
Product activation	52
Server Configuration - Server Preferences	54
Driver Installation	57
How to turn on the SMPP communication logging	71
Plugins	77
System maintenance	79
Keep the system healthy by deleting old messages	79
Ozeki Performance Monitor	79
Manual uninstallation	79
DEVELOPERS GUIDE	81
General information	81
Autoreplying SMS applications	81
Application Mode	82
Programming APIs and languages	84
SMS API for C++ Developers	84
SMS API for Database developers (SQL)	85
SMS API for Webdevelopers (HTTP)	101
SMS API for JAVA developers	111
SMS API for PHP developers	113
SMS API for ASP developers	116
SMS API for Delphi developers	117
SMS API for Microsoft Access developers	120
APPENDIX	134
Appendix "A1" - Message types - OTA bitmap	135
Appendix "A2" - Message types - Operator logo XML	136
Appendix "A3" - Message types - Operator logo	138
Appendix "A4" - Message types - Ringtone	140
Appendix "A5" - Message types - Wap push	143
Appendix "B" - GSM service providers	145
Appendix "C" - SMS service center addresses	155
Appendix "D" - GSM operator and country codes for operator Logos	160
Appendix "E" - The 7 bit default alphabet of GSM phones	167
Appendix "F" - GSM error codes	173
Appendix "G" - Database plugin connection strings	175
Appendix "H" - References	185

Introduction

Organizations today are facing several unique communication challenges: the proliferation of electronic messages across their enterprise; the need to provide employees, customers, and partners with consistent information; the desire to better equip information workers with meaningful information to drive informed decisions; and the mandate to control costs without sacrificing application availability, security, or reliability.

The latest release of Ozeki Message Server is designed to help enterprises address these challenges. Ozeki Message Server 6 will help in maximizing the productivity of IT by reducing the complexity of creating, deploying, and managing communication applications. It empowers developers through a rich, flexible, modern development environment for creating secure, high performance communication applications.

Various types of messages (SMS text messages, binary data messages, etc...) can be sent from multiple platforms, applications, and devices to make it easier to connect internal and external systems. Deliver robust, integrated business intelligence solutions that help drive informed business decisions and increase productivity across your entire organization. Control costs without sacrificing performance, availability, or scalability.

Read on to learn more about the advancements Ozeki Message Server will deliver in three key areas: enterprise messaging, developer productivity, and cost control.

About this document

This document describes the Ozeki Message Server software. It gives an overview of the functionality, describes the user interface and provides all the information necessary for configuration. The document is intended for operator personnel involved in the installation, configuration and daily operations of the Ozeki Message Server Software. It provides useful information for application developers as well.

Terms of use

The information in this document is subject to change without notice and describes only the product defined in the introduction of this documentation. This document is intended for the use of Ozeki's customers only for the purposes of the agreement under which the document is submitted, and no part of it may be reproduced or transmitted in any form or means without the prior written permission of Ozeki. The document has been prepared to be used by professional and properly trained personnel, and the customer assumes full responsibility when using it. Ozeki welcomes customer comments as part of the process of continuous development and improvement of the documentation.

The information or statements given in this document concerning the suitability, capacity, or performance of the mentioned hardware or software products cannot be considered binding but shall be defined in the agreement made between Ozeki and the customer. However, Ozeki has made all reasonable efforts to ensure that the instructions contained in the document are adequate and free of material errors and omissions. Ozeki will, if necessary, explain issues which may not be covered by the document.

Ozeki's liability for any errors in the document is limited to the documentary correction of errors. OZEKI WILL NOT BE RESPONSIBLE IN ANY EVENT FOR ERRORS IN THIS DOCUMENT OR FOR ANY DAMAGES, INCIDENTAL OR CONSEQUENTIAL (INCLUDING MONETARY LOSSES), that might arise from the use of this document or the information in it.

This document and the product it describes are considered protected by copyright according to the applicable laws.

Ozeki logo is a registered trademark of Ozeki Informatics Ltd.

Other product names mentioned in this document may be trademarks of their respective companies, and they are mentioned for identification purposes only.

Copyright © Ozeki Informatics Ltd 2000-2005.

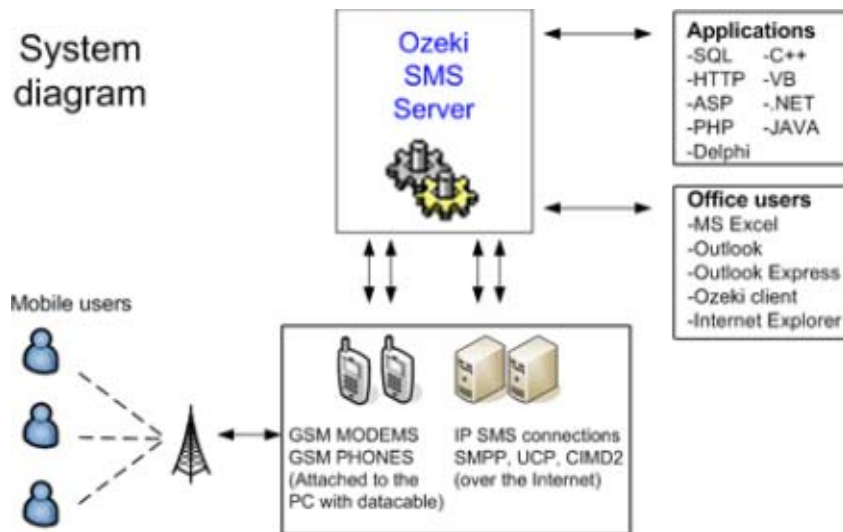
All rights reserved. © Ozeki Informatics Ltd.

Summary of changes

Date	Document version	Updated sections / changes
16th March, 2006	v1.0	The first version of this document is available

About Ozeki Message Server 6

Ozeki Message Server 6 - SMS Server is a powerful, flexible SMS Gateway application, that enables you to send/receive SMS messages to mobile devices with your computer. It has an easy to use user interface, and an excellent internal architecture. The application can use a [GSM mobile phone](#) attached to the PC with a phone-to-PC data cable or [IP SMS technology](#) to transmit and receive the messages. Ozeki Message Server works on Microsoft Windows XP,2000,2003 operating systems.



Office users

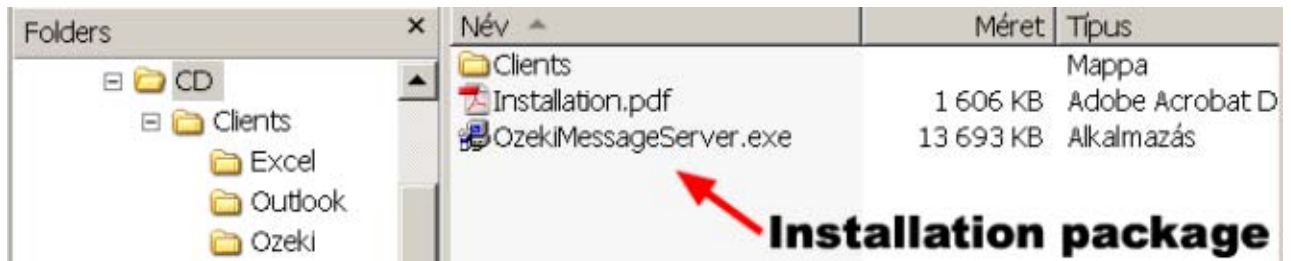
Office users can use [Microsoft Outlook](#), [Microsoft Outlook Express](#) and [Microsoft Excel](#) to send hundreds of messages to their clients. The messages and the phone numbers are stored in Excel files and an Excel Macro initiates the sending process. (The excel macro is included in the software package.)

Software developers

Software developers can integrate SMS messaging functionality into their applications very easily. For example if a message needs to be sent, it could be placed into a database table. The Ozeki Message Server monitors this table and delivers the message. The Message Server puts all received SMS in an incoming database table. Of course many other APIs are available in the software to support software development.

Package contents

If you purchase Ozeki Message Server, you can download a package that contains the following installation files:



If you click on OzekiMessageServer.exe, you can install the software just like you would install any other Windows application. The installation will create a new folder in C:\Program Files that will contain the following components.

Ozeki Message Server	The main message engine. This program operates GSM phones or IP SMS connections. It handles message queues and users. It runs as a system service in the background. It contains a built in HTTP server that is used by clients.
Ozeki Message Server Monitor	This application monitors the activity of the main message engine and displays a status icon in the windows taskbar.
Ozeki Message Server Manager	This Ozeki Message Server Manager can be used to configure and manage the main message engine. It has an easy to use graphical user interface (GUI), that can be used to access the message folders (queues) and to modify user and server preferences.
Excel SMS Client	An .xls file, that can be opened with Microsoft Excel to send SMS messages from Excel spreadsheets. This file contains the Excel Macro that is used for this purpose.
Outlook SMS Client	An application that can be installed on client computers to make SMS sending from Microsoft Outlook possible.
Ozeki SMS Client	A GUI application for SMS messaging with a user interface similar to popular e-mail clients.
Product documentation	Detailed documentation in PDF format with examples and case studies.

(The list of components may vary depending on your license.)

Ozeki Message Server

If you install the Ozeki Message Server a new service will be registered in your system. This service will communicate with the GSM phone attached to your PC or if you use an IP SMS connection it will communicate with your GSM provider through the Internet. To enable you to use the software on a 24 hours a day / 7 days a week basis, the service will run in the background all the time. If you restart your computer it will start automatically.

After installation you may check the properties of the new service by starting the Windows Service Manager.

To start the Windows service manager click the Start menu and select Control Panel/Administration tools/Services menu. (Figure 1)

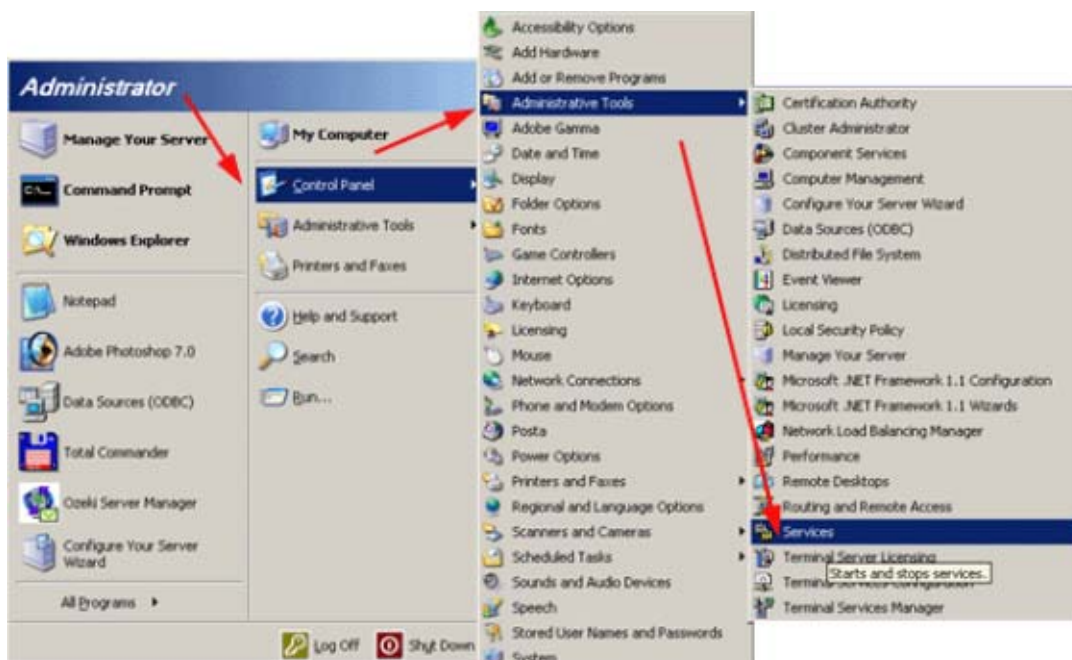


Figure 1 - starting the service manager

In the Service Manager you can find an entry called Ozeki Message Server. You may click on this link with the right mouse button to see it's properties. (Figure 2)

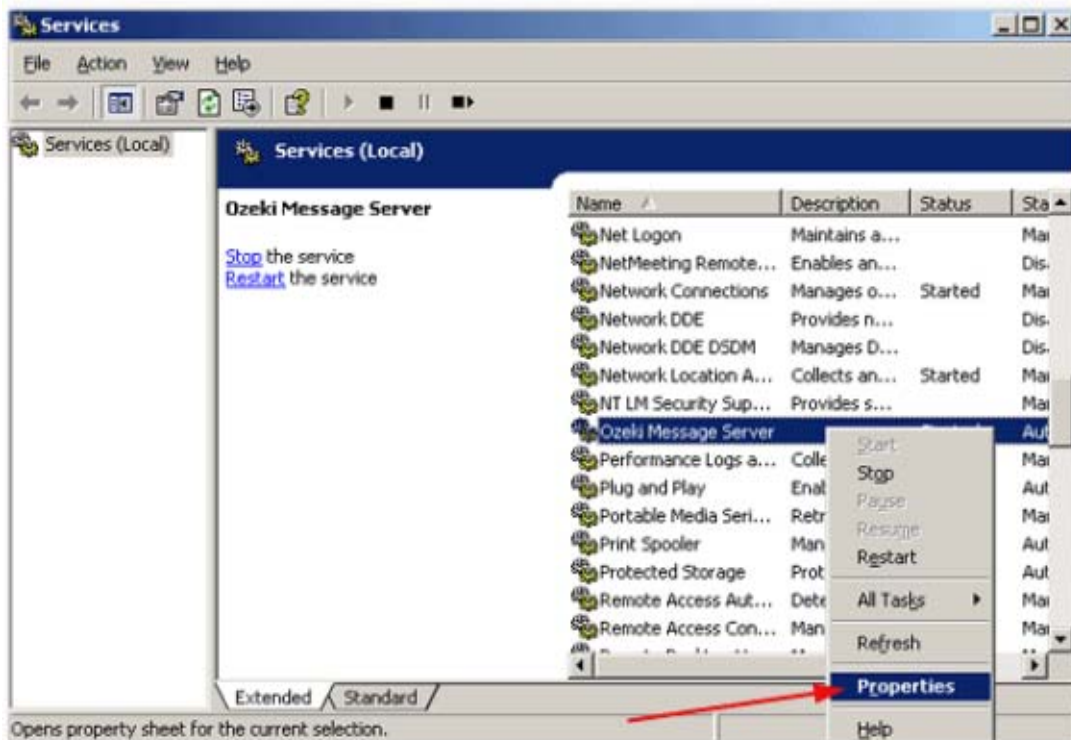


Figure 2 - Service properties

Ozeki Message Server Monitor

The **Ozeki Message Server Monitor** provides information about the status of the Message Engine. It is a taskbar icon (Figure 1.) application. If the message engine works properly, the taskbar icon signals green if the message engine stops it turns into red. After installation the Message Server Monitor is started by default. To start it manually you should click on the "Message Server Monitor" icon on your desktop.



Figure 1 - taskbar icon

If you click on this icon with the right mouse button, you can open a menu, that provides options for starting/stopping and configuration of the server. To perform the configuration you must start the Message server manager by clicking on the Manage Service menu item (Figure 2).

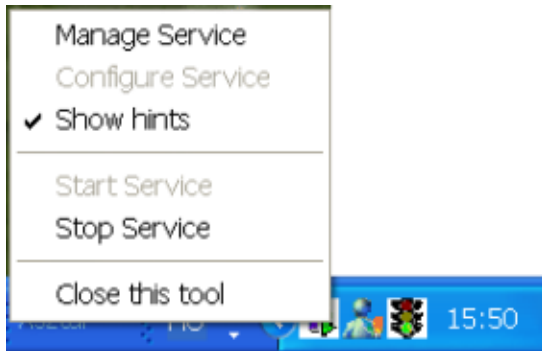
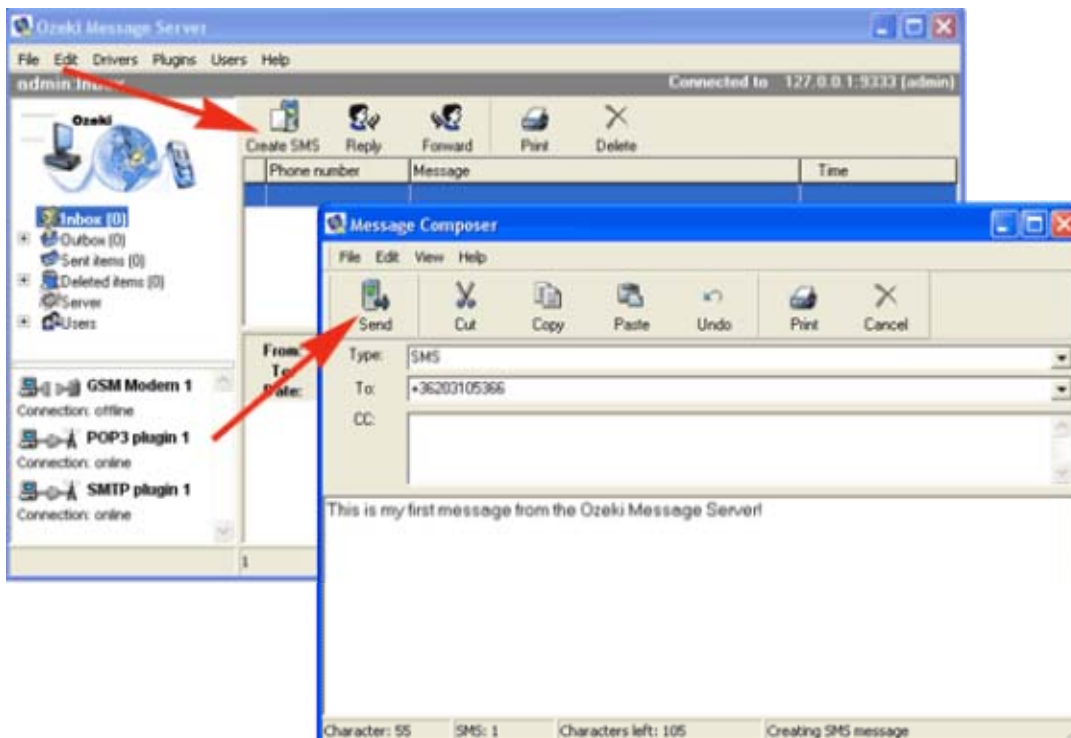


Figure 2 - Starting the message server manager.

Ozeki Message Server Manager

The Message Server Manager is the main user interface of the application. It can be used to configure the service, to send/receive messages, to maintain the addressbook and to get information from the server real time. It can be used a management console to manage the server remotely over the network. The Message Server Manager can be started from the Start menu or from the Server Monitor Taskbar icon. After startup the Ozeki Message Server Manager requires a login name and a password. The default user name and password is: **admin / abc123**. (**Note: It is important that you change this password after the first login.**)

On the main form of the Message Server Manager you can perform the necessary configuration and management tasks and you can send and receive SMS messages.



OZEKI Excel SMS Client

The Excel SMS client enables you to send SMS messages from Microsoft Excel. To achieve this goal, all you have to do is open two excel spread sheets: Messages.xls and Ozeki.xls. In the Messages.xls spread sheet, you need to fill the telephone numbers into the first column and the messages into the second column. In the Ozeki.xls excel sheet you will find a macro that can be started by clicking on the Send all button. This makro will connect to Ozeki Message Server 6 over the office LAN or over the internet and will send the SMS messages you have edited on the Messages.xls sheet. The parameters for this network connection can be edited on the Ozeki.xls spread sheet (Figure 1).

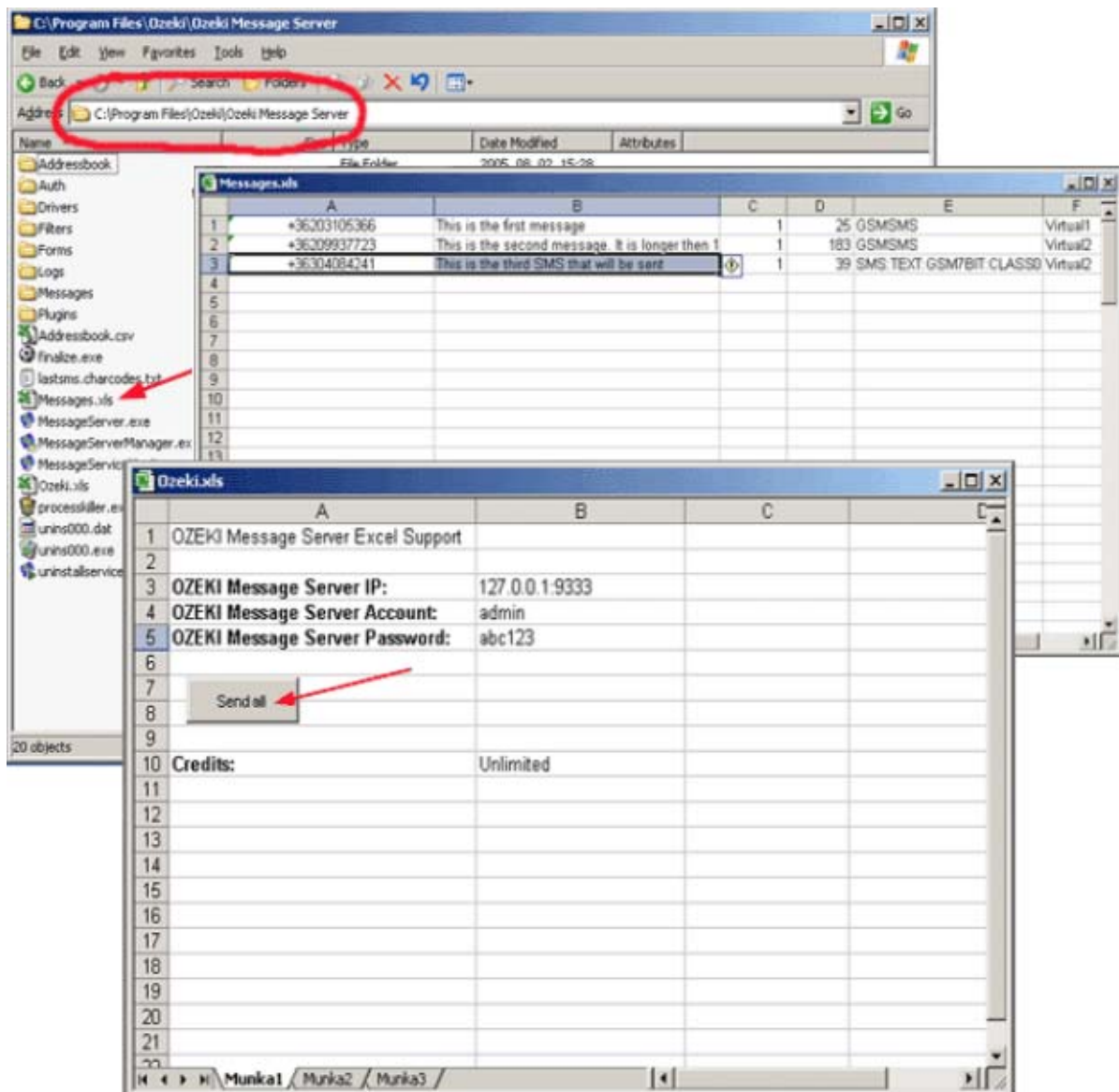
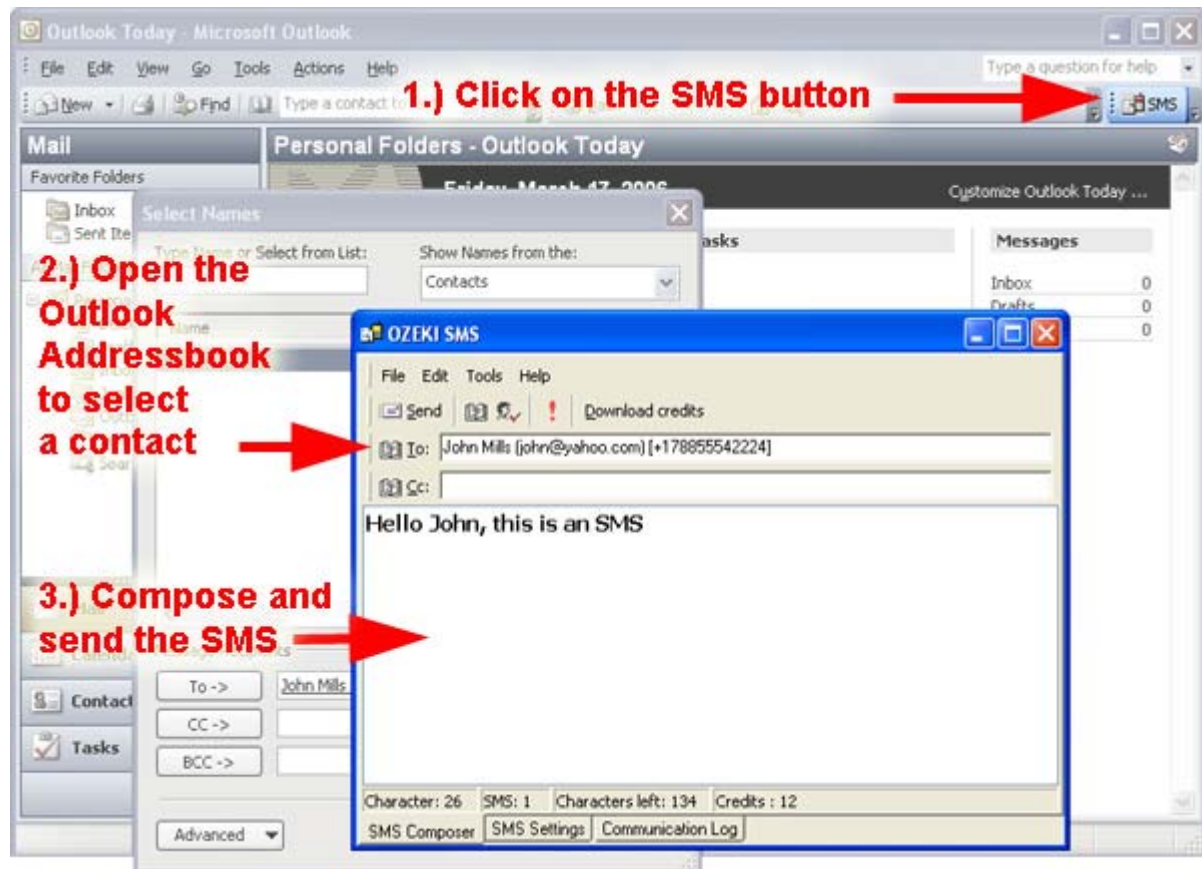


Figure 1 - Creating the excel sheet

Outlook SMS Plugin

The Outlook SMS Client plugin creates a button in Microsoft Outlook. If the button is pressed a message composer window is opened. In this window the SMS message can be created and by clicking on the send button it is sent over the network to the Ozeki Message Server engine for delivery.



The sent messages are saved in the sent items folder of Outlook.

If an incoming message arrives it is saved in the inbox folder of Outlook.

The user of this Outlook SMS Client can use the addressbook of Outlook to look up and select the appropriate phone numbers for delivery.

Ozeki SMS Client

The Ozeki SMS client is similar to any e-mail client, that communicates with a server. It can be used in your office network to enable your colleagues to send and receive SMS messages from their desktop computers using a central server. The Ozeki SMS Client has a very easy to use graphical user interface (Figure 1).

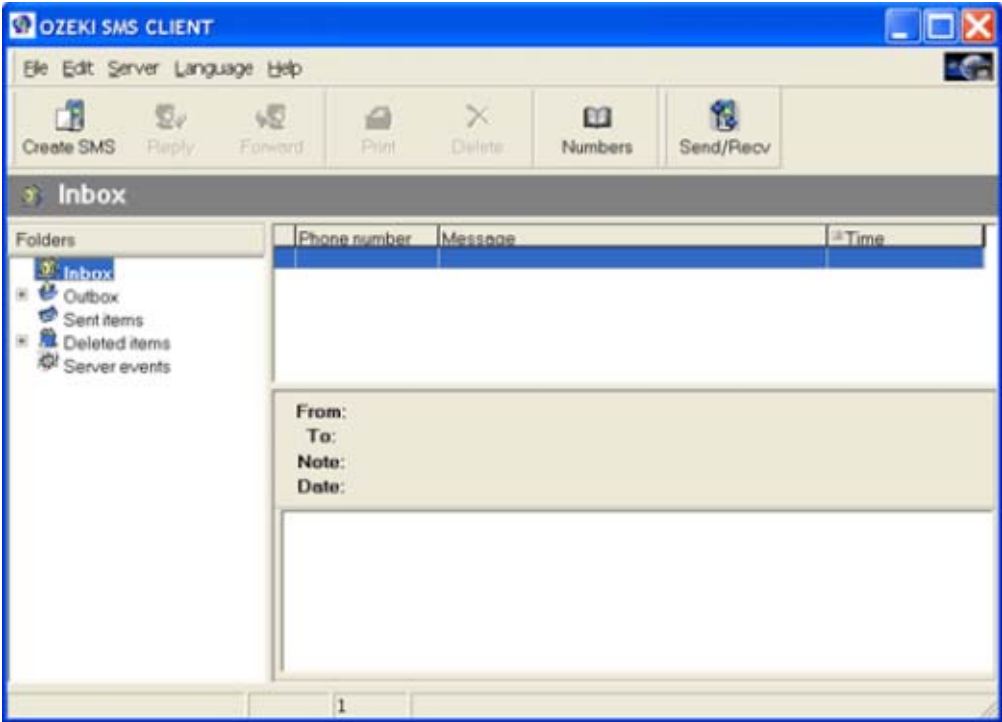


Figure 1 - Ozeki SMS Client

The communication protocol used between the Ozeki SMS client and the Ozeki Message Server is HTTP. The client connects to the server periodically to check for new messages and to send the messages that can be found in the outbox.

SMS Technology

The SMS technology was created in Europe, by GSM pioneers. The standardization process is lead by the European Telecommunications Standards Institute (ETSI). The SMS technology was created to provide an infrastructure for the transportation of short messages containing a maximum of 140 bytes (8 bit objects) of useful data in mobile telecommunication networks. The transportation is done in the GSM signaling path in traditional GSM networks, and as GPRS packets in GPRS networks. The messages are composed using the PDU specification. An SMS is a binary string, that contains all the necessary information to form the message header needed for transportation and the message body that contains the payload. The basic addressing scheme of SMS messages are mobile telephone numbers called MSISDN.

SMS technology fact sheet:

Standard:	GSM (ETS 03.40)
Transport technology:	GSM Signaling Path , GPRS
Transport protocol:	Short Message Protocol
Addressing scheme:	MSISDN
Message description language:	SMS PDU
User data length:	140 byte (Concatenated messages can exceed this size)
Basic character set:	7-bit SMS, 8-bit SMS, UCS2

The SMS network entities

SMS messages are created by mobile phones or other devices (e.g.: personal computers). These devices can send and receive SMS messages by communicating with the GSM network. All of these devices have at least one MSISDN number. They are called Short Messaging Entities. To understand their role in the GSM SMS network please take a look at the following figure:

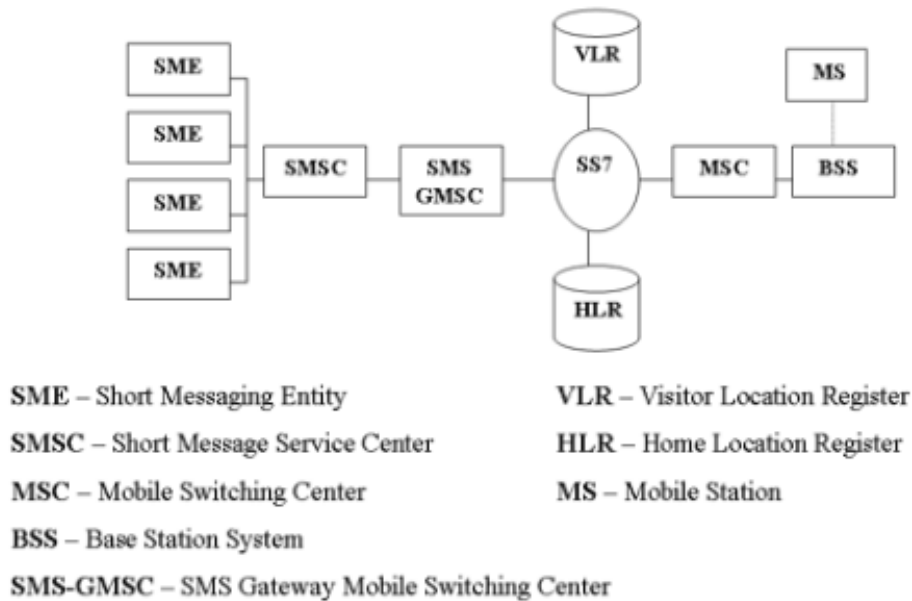


Figure 1. - Short Messaging Entities (SME) in the GSM network

The SME-s are the starting points (the source) and the end points (the receiver) for SMS messages. They always communicate with a Short Message Service Center (SMSC) and never communicate directly with each other. An SME can be a Mobile telephone. Depending on the role of the mobile phone in the communication we can talk about two kinds of SMS messages Mobile Originated (MO) messages and Mobile Terminated (MT) messages. MO messages are sent by the mobile phone to the SMSC. Mobile terminated messages are received by the mobile phone. The two messages are encoded differently during transmission.

An SME can also be a computer equipped with a messaging software, such as the Ozeki Message Server, that can communicate directly with the SMSC of the service provider. For this communication a mobile phone attached to the PC with a phone-to-pc data cable or a direct IP link can be used.

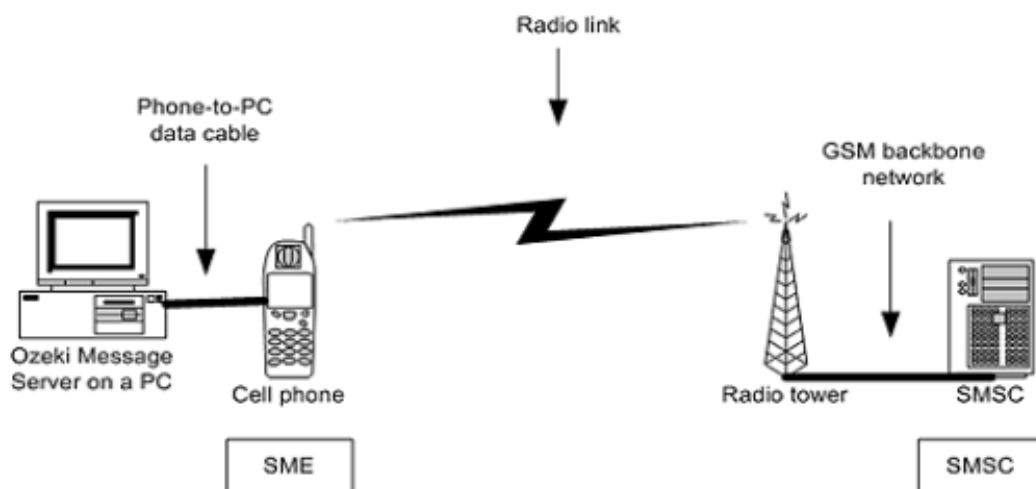


Figure 2 - Ozeki Message Server used with a mobile phone attached to the PC

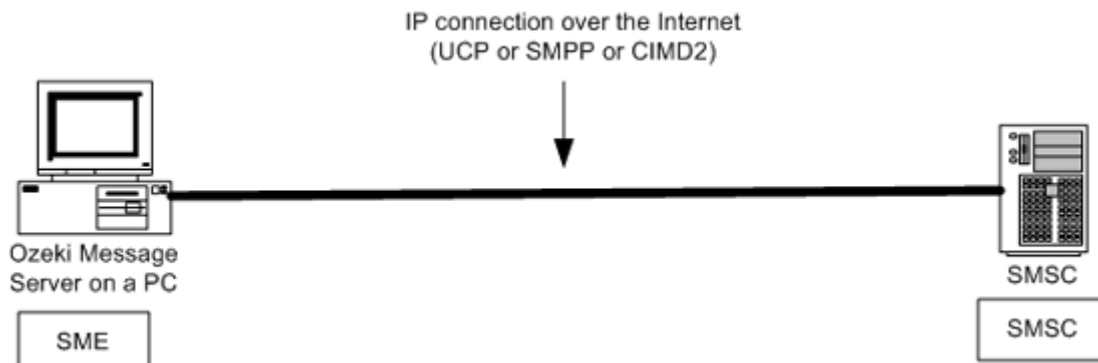


Figure 3 - Ozeki Message Server used with direct IP connection

SMS Network Architecture and Internal Protocols

In the SMS network operated by GSM service providers four layers of transport is used (Figure 1.):

- SM-AL (Application Layer)
- SM-TL (Transfer Layer)
- SM-RL (Relay Layer)
- SM-LL (Lower Layers)

Ozeki Message Server is located in the SM-AL layer. When sending an SMS the software creates protocol data units (PDU's) transported by the SM-TL layer. When a GSM phone attached to the PC receives an SMS, the message is also encoded according the SM-TL layer PDU specification. Ozeki Message Server decodes this PDU and makes the message readable for computer programs and computer users.

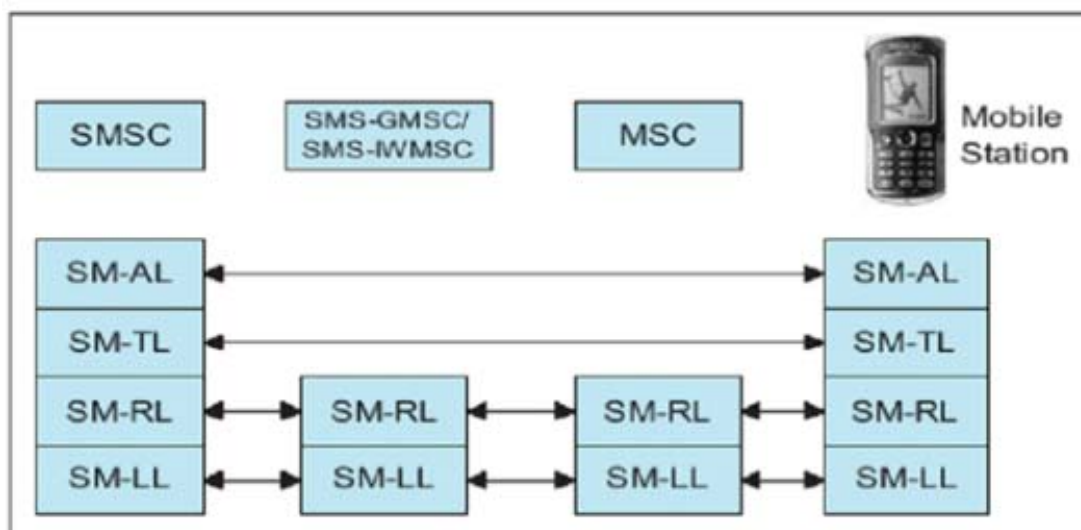


Figure 1. - SMS network layers

To understand how the SMS travels from the mobile phone to the SMSC please take a look at Figure 2. On this figure you can see which protocols are used and which GSM network entities take place in the communication process. As you can see, the mobile phone (Mobile station) transmits the SMS message to the GSM base station (BTS) through a wireless link. After this the message goes through the backbone network of the service provider. The Mobile Switching Center (MSC), the Home Location Register (HLR) and optionally the Visitor Location Register (VLR) is used to find out the appropriate Short Message Service Center

(SMSC) that will store and forward the message when the receiving party becomes available.

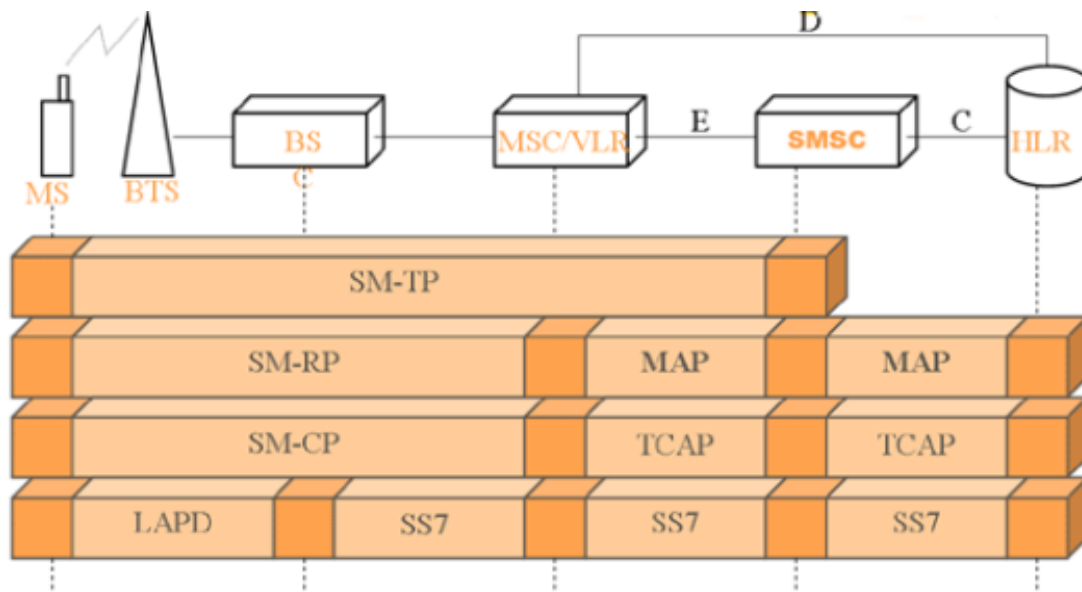


Figure 2. - SMS transportation

SMS message types

The Short Message Service (SMS) makes it possible to send and receive messages to and from mobile telephones. The messages can contain text or binary data. The text can comprise of words or numbers or an alphanumeric combination. A single short message can contain up to 160 characters when Latin alphabets are used, and 70 characters when non-Latin alphabets such as Arabic and Chinese are used. The latest SMS standards released by ETSI provide an extension to send longer text messages in SMS. This technology is called concatenated SMS messages.

An SMS message can also carry binary data. A binary SMS message has 140 bytes of used data. This user data can be split into two parts: the user data header and the actual data. The user data header can be used to inform mobile phones about certain message types. This way SMS can be used to carry operator logos, ringtones, telephone settings and WAP push messages.

Although the main goal of the SMS technology is to send and receive SMS messages between mobile telephones, it is possible to send and receive messages between a mobile phone and a computer as well. The Ozeki Messages Server was developed to help exploit this possibility. The software can be used to create automated SMS services or it can be used to simply send SMS messages from popular desktop applications such as Microsoft Excel and Microsoft Outlook.

Ozeki Message Server currently supports the following message types (Figure 1).

SMS:TEXT	A text message, that can contain alphanumeric characters, numbers from the default GSM alphabet . The basic message length is 160 characters. If the message length is longer it is sent
----------	--

	using more than one messages. In this case each SMS contains 153 characters.
SMS:TEXT:GSM7BIT:CLASS0	A text SMS message. Often referred to as FLASH SMS messages. It is different from traditional messages, because it is immediately displayed on the screen of the phone and it is not saved in the memory of the phone.
SMS:TEXT:GSM7BIT:CLASS1	A text SMS message. Class1 indicates that the message should be stored on the Device.
SMS:TEXT:GSM7BIT:CLASS2	A text SMS message. Class2 indicates that the message should be stored on the SIM card.
SMS:TEXT:GSM7BIT:CLASS3	A text SMS message. Class3 indicates that the message should be sent to the terminal equipment (TE) attached to the phone.
SMS:TEXT:UCS2	A text SMS message encoded using unicode characters. This message type is used if, arabic, chinese, or other special characters should be sent. A unicode text message can contain 70 characters. If the message text is longer and concatenated messages are used, 63 unicode characters are sent in one SMS message.
SMS:BINARY	A binary SMS message. The maximum message length is 140 octets.
SMS:BINARYXML	A binary SMS message with direct access to the PID, DCS and UDH fields in the SMS PDU.
SMS:WAPPUSH :application/vnd.wap.sic	A WAP push message containing a URL link to a WAP or Webpage
SMS:OPLOGO	An operator logo message.
SMS:OPLOGOXML	An operator logo message with direct access to the country code and operator code in the message.
SMS:RINGTONE	A ringtone message.
SMS:CALLERICON	A callericon message.
SMS:PICTUREMESSAGE	A picture message.
SMS:VCARD	A vcard message.
SMS:VCALENDAR	A vcalendar message.
DATACALL	A datacall message contains a binary file. This message type is not sent as an SMS. Instead it is sent using a modem data call to another modem. The file is sent using the zModem protocol.

Figure 1 - Messages types

SMS messaging with a mobile phone attached to the PC

If you plan to send/receive less than 15 000 SMS messages per day a good option is to attach a GSM phone or GSM modem to your PC. The GSM phone must be equipped with a SIM card that has good tariff for SMS messages. The GSM phone can be attached with a data cable, with an InfraRed port (IR) or with a BlueTooth (BT) connection. With this setup you can use a computer program such as Ozeki Message Server to send/receive SMS messages. The computer program in this case uses the attached device to communicate with the GSM network.

If a message is sent by the application running on the computer it is first sent to the attached GSM phone, and as a second step the GSM phone transmits the messages to the SMSC of the GSM service provider through a wireless link (Figure 1). When a message is received, the GSM phone stores the message in its memory or on the SIM card and sends a notification to the PC. When the program running on the PC receives this notification, it reads the appropriate memory cell and deletes the message from the phone to make room for the next incoming message (Figure 1).

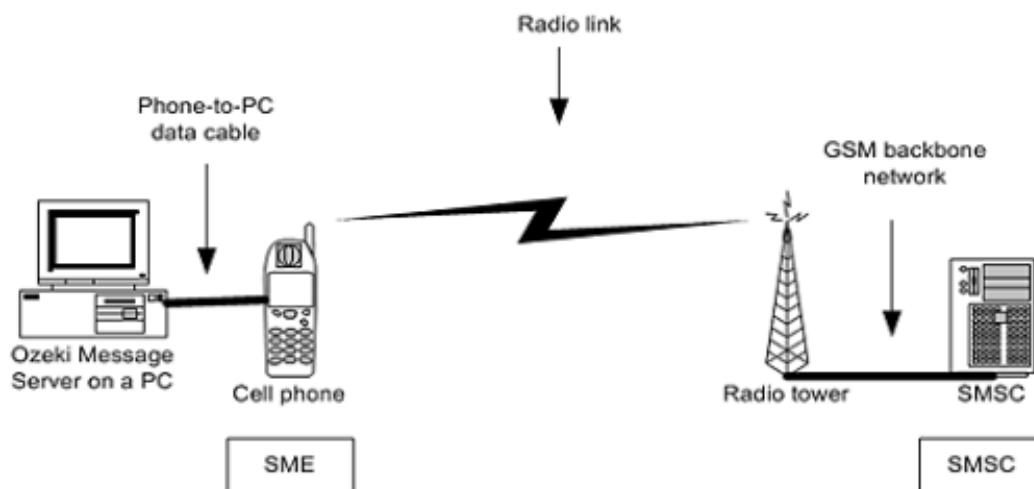


Figure 1 - IP SMS connection to send and receive large number of SMS messages.

Sending an SMS message using a cell phone takes about 5-6 seconds. Receiving takes about the same time. Good software, such as Ozeki Message Server allows you to attach more than one devices to your PC and to use them simultaneously to increase capacity.

The best option to connect a phone to the PC is to use a standard RS232 serial cable. USB cables, InfraRed and BlueTooth connections are not that reliable. You can read more about the connection methods on the following pages

Serial cable connection
USB cable connection
Infrared (IR) connection
Bluetooth (BT) connection

Connect your phone to the PC with an RS232 serial cable

In order to send or receive SMS messages from your PC, you can attach your phone to the PC with a phone-to-pc data cable (Figure 1). The cable is attached to the serial port of your computer. The serial port is located on the back side and has 9 pins. The name of the serial port in your operating system is COM1 or COM2.



Figure 1 - Attaching the device to the PC with a phone-to-pc data cable.

If you select from different alternatives, for example (Serial, USB, IR, BT connection), choosing the serial cable is the best option. An RS232 serial data cable is a simple and reliable tool. Industrial GSM modems used in environments, where performance and reliability is crucial are always connected using a RS232 cables.

If you wish to build an environment where it is important to operate all the time, we suggest to use an industrial GSM modem (Figure 2):



 <p>External GSM antenna</p> <p>RS232 serial cable</p> <p>SIM Card</p>	<p>WAVECOM</p> <p>Wavecom provides very fast and reliable modems for a good price. These modems need an external GSM antenna, an RS232 data cable. Just like the Siemens MC35 they can be connected directly to the power supply of your PC</p>
 <p>External GSM antenna</p> <p>RS232 serial cable</p> <p>SIM Card</p>	<p>NOKIA 30/ Nokia 32</p> <p>The Nokia 30 is also an excellent GSM phone. You can read more about in it's installation guide: Nokia_30_UG_en_.pdf</p>

Figure 2 - Industrial GSM modems

Connect your phone to the PC with an USB data cable

In order to send or receive SMS messages from your PC, you can attach your phone to the PC with a phone-to-pc data cable (Figure 1). The cable can be attached to the USB port of your computer. One advantage to using USB connections is that you can attach quite a large number of phones to the same PC with the help of USB hubs available at a low cost. One disadvantage is that you have to install a driver for your cable. Some of the drivers shipped with USB cables have errors and don't allow you to build a reliable system. If you install the driver software shipped with the USB data cable, a new "virtual" serial port will be register in your computer. The name of this serial port will be COM3 or COM4 or some higher number.



Figure 1 - USB cable connection

The USB cables available for cell phones are built on a special chip called USB - Serial Bridge. This chip is located inside the cable and is powered by the USB port. When you plug your USB cable into the PC the Windows OS will detect this chip and will ask for the USB driver.

Some USB cables can use the power current available in the USB port to charge the battery of the phone. If you purchase a cable with this option you must make sure, that the charging will happen automatically and no special software is required. If you have to install a software for charging there is a high probability that the installed software will lock the serial port and you won't be able to use the cable for SMS messaging and battery charging simultaneously.

If you don't have an USB data cable available for your phone, but you can get a standard RS232 data cable, you can still use USB (Figure 2). In this case you should install an USB-Serial converter. The USB-Serial converter can be attached to the PC's USB port and the data cable can be attached to it. In this case the installation procedure will be similar to the procedure you would follow when you would install a standard USB data cable. First you attach the cable to the computer, then you install the driver for the cable. The driver will create the "virtual" com port you can use for SMS messaging.



Figure 2 - USB serial converter

Connect your phone to the PC with Infrared (IR)

You can attach your phone to your PC to send and receive SMS messages through an Infrared connection. Most mobile phones have IR ports and notebook computers are also equipped with it. An Infrared connection is a good option if you want to use Ozeki Message Server for short periods to send and receive messages. IR ports cannot be used to establish long lasting connections between the phone and the PC for two main reasons. There must be a line of sight between the two communicating parties, which is often broken if one of the devices are moved. The other problem is the way IR connectivity is implemented in GSM phones. Very often the phones switch off their IR port after a certain period (e.g. 10 minutes) automatically.

If you are in a situation, where IR is your choice, to use Infrared (IR), you must enable your infrared port on the phone. (Sometimes you have to turn it on in the PC as well). When this is done, you should place the phone within the range of the IR receiver on the PC. When the PC and the phone begin to communicate over IR, you will see an icon in the bottom right hand side of the screen (Figure 1).



Figure 1 - IR connection between the cell phone and the PC

After this icon appears a new "Virtual" serial port is registered into your PC. This serial port will have a standard name e.g.: COM3. When you configure Ozeki Message Server, you should select this newly created COM port.

Connect your phone to the PC with Bluetooth (BT)

You can attach your phone to your PC to send and receive SMS messages through a Bluetooth (BT) connection. The Bluetooth technology is based on radio signals, so this connection is wireless. One strong advantage to bluetooth over infrared connections is that no line of sight is required. If you have a bluetooth phone within range of your PC, it can be used to send and receive SMS messages. One problem with bluetooth is that several devices can be within range and a quite complicated pairing process is required when you establish the connection.

The bluetooth standard provides many services. Among these services there is a bluetooth serial connection service. To use bluetooth on your computer, it must be equipped with a bluetooth antenna. Bluetooth antennas in most cases are connected to the USB port of your PC. When you install the driver of a bluetooth antenna into, it will register "virtual" serial ports called COM3, COM4, COM5, etc. When a bluetooth capable cellphone is attached it can be accessed through one of these virtual serial ports.



Figure 1 - Bluetooth connection

To sum it up:

In order to use the bluetooth connection, a PC must have bluetooth support. This can be achieved with an USB bluetooth antenna and an appropriate driver installation. The mobile phone must be bluetooth capable and the bluetooth link must be activated. Following this the serial connection can be initiated from the PC by opening the appropriate virtual COM port. (Figure 1.)

IP SMS technology

If you plan to send or receive a larger number of SMS messages, it is worth to consider using the Internet (Figure 1). Over the Internet (or over a private IP network, such as a leased line), it is possible to connect directly to the GSM operator's SMS Center (SMSC). The SMSC is responsible for receiving, storing and delivering the SMS messages in a GSM telephone network. The benefits of direct connection to this service are great.

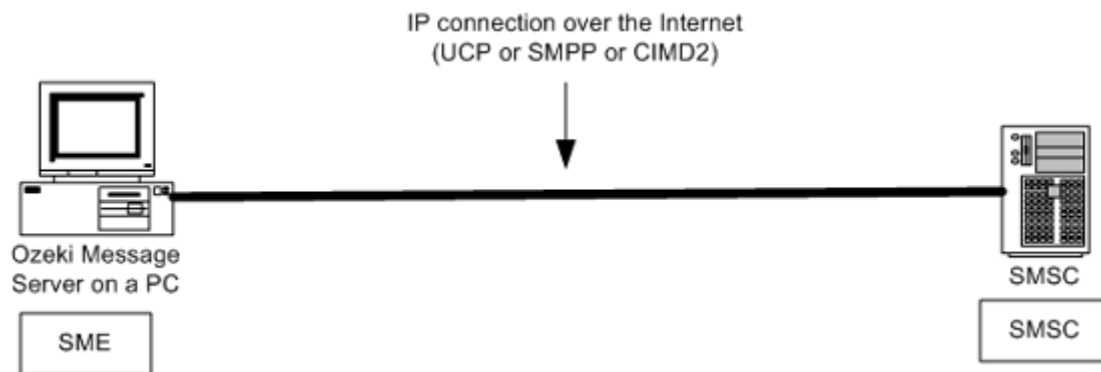


Figure 1 - IP SMS connection to send and receive large number of SMS messages

A large number of messages can be sent or received in a short time. The sender phone number can be freely modified. The availability of the service can be monitored.

If you work with great volumes of messages, this solution is cheaper, faster and more reliable than the alternative solution of using a GSM handy connected to a computer.

Ozeki Message Server enables you to connect directly to your service provider's SMSC through the Internet or a leased line using TCP/IP. In order to use this option, first you need to sign a contract with your GSM service provider, to get direct access to it's SMSC. If you have a contract you will receive their IP address, a port number and user name and password to access their service. Often they will ask to set up a VPN connection to their service. After a connection is available to the SMSC, an SMS protocol is required to be able to send and receive SMS messages. The most common protocols are UCP, SMPP and CIMD2. (Of course all of these are supported by Ozeki Message Server. In fact you can use more then one of these protocols simultaneously.) You can read more about these protocols on the following pages.

Universal Computer Protocol (UCP)
Short Message Peer to Peer Protocol (SMPP)
Computer Interface Message Distribution version 2 (CIMD2)

It is also important to mention, that some service providers use custom

protocols, for example XML over HTTP. If you wish to use Ozeki Message Server with your service provider, and your protocol is not yet included in our program, please contact us. There is a good chance we will add support to your service free of charge.

Common protocols - UCP/EMI

Ozeki Message Server has built in support for Universal Computer Protocol / External Machine Interface (UCP / EMI). This protocol is often used by GSM service providers for IP SMS connectivity. The currently used protocol versions (v3, v3.5, v4.0) can be used to transfer high volumes of SMS traffic. It is used for both premium rate and normal rate SMS messages. In order to use UCP to connect to an SMSC over the internet or other private TCP/IP network, a contract should be signed with the service provider. The TCP/IP link can be a leased line or the Internet.

One advantage to using UCP connections, is that the client is able to specify the sender address of the message. The sender address can be a custom phone number or a short word containing alphanumeric characters. Unfortunately some GSM service providers restrict the use of this option. Another advantage of UCP is that it makes it possible to send every message type the GSM network can carry. It can be used to send binary SMS messages (operator logos, ringtones), wap push messages and other messages as well. Concatenated SMS messages and various alphabets can also be used.

If you wish to use UCP to connect to your service provider, you need to get the following information from the service provider:

IP address of the SMSC
Port number of the SMSC
User name
Password

Some service providers require you to set up an IPSEC VPN tunnel when you wish to connect to their SMSC (Figure 1).

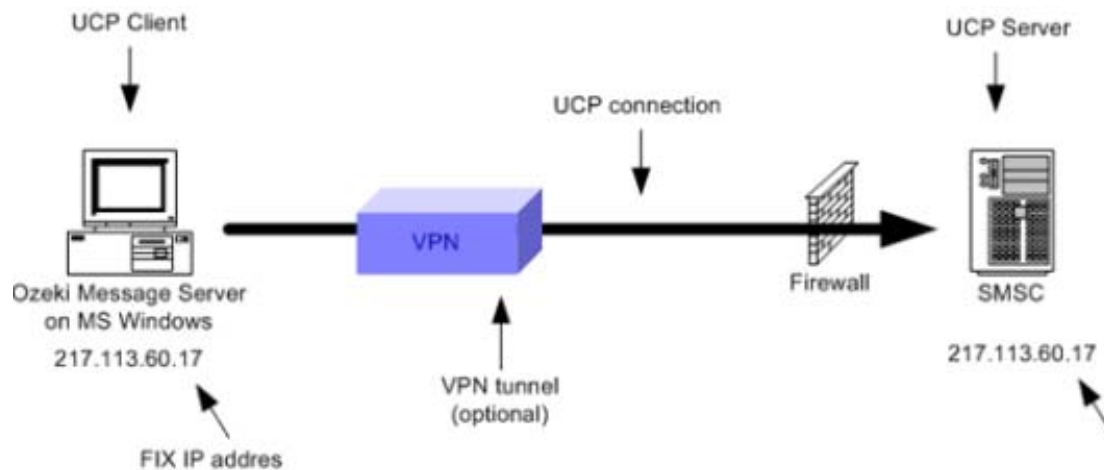


Figure 1 - UCP connection with VPN

In most cases you need a fix IP address, that you have to tell the provider so they can give you access through their firewall. In some situations, the service provider does not want you to be able to set up more then one UCP connections from your computer to their service. If this is the case, they might ask you to use a specific source port for the connection.

Common protocols - SMPP

The Short Message Peer to Peer (SMPP) protocol is an open, industry standard protocol designed to provide a flexible data communications interface for the transfer of short message data between External Short Message Entities (ESME), Routing Entities (RE) and Message Centers. SMPP is capable to carry any message type just like UCP/EMI.

SMPP can be used as a protocol that transfers messages between applications, such as **Ozeki Message Server** and the Short Message Service Center (SMS Center) of the GSM Service provider over an IP link. This link can be a leased line or the Internet. The IP connection between your PC and the SMSC might be protected by VPN (Figure 1.)

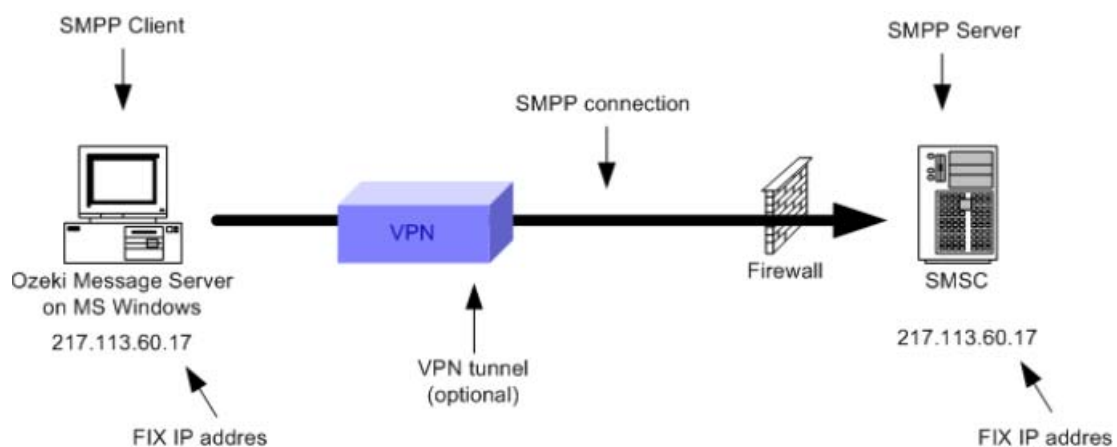


Figure 1 - SMPP connection between a PC and the SMSC

If you wish to use UCP to connect to your service provider, you need to get the following information from the service provider:

IP address of the SMSC
Port number of the SMSC
User name
Password

If you would like to get some insights about SMPP, you must be familiar with some of the following terms. A Message Centre (MC) is a generic term used to describe systems such as a Short Message Service Centre (SMSC), GSM Unstructured Supplementary Services Data (USSD) Server, or Cell Broadcast Centre (CBC).

A ESME typically represents a fixed network SMS client, such as a WAP Proxy Server, EMail Gateway, or Voice Mail Server. It may also represent a Cell Broadcast Entity (CBE). A Routing Entity (RE) is a generic term for a network element that is utilized for MC to MC, and ESME to MC message routing. A RE has the ability to emulate the functionality associated with both a MC and an ESME. To an ESME, a RE appears as a MC and to a MC, a RE appears as an ESME. A carrier may utilize REs to hide a network of Message Centres, presenting only the REs as the external interface point for ESMEs.

Figure 2 illustrates the context of SMPP in a mobile network:

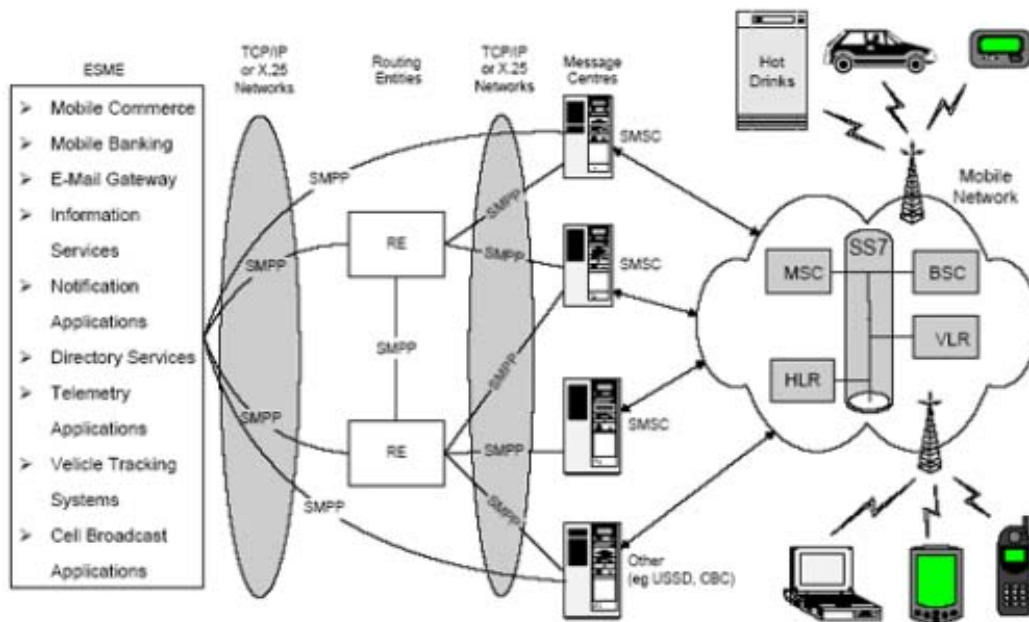


Figure 2 - SMPP inside the mobile network

Common protocols - CIMD2

The Computer Interface Message Distribution (CIMD) protocol was introduced by Nokia. The Nokia SMS Center includes an CIMD2 interface, that transfers messages between applications and the Nokia Short Message Service Center (SMS Center). CIMD2 is version 2 of the Computer Interface to Message Distribution protocol. The CIMD2 interface is suitable for client applications that send and retrieve messages over TCP/IP sockets, such as **Ozeki Message Server**.

An application is interconnected through the CIMD2 connection to a SMS Center. The main purpose of this interconnection is to transfer messages from the applications to the mobile stations (MS) and from the MSs to the applications. Other kind of information can also be conveyed over the interconnection, for example, status reports from the GSM/GPRS network to the applications.

CIMD2 is capable of transferring any SMS message type just like UCP/EMI and SMPP.

A CIMD2 connection can be used through an IP link. This link can be a leased line or the Internet. Optionally the link can be protected by VPN (Figure 1).

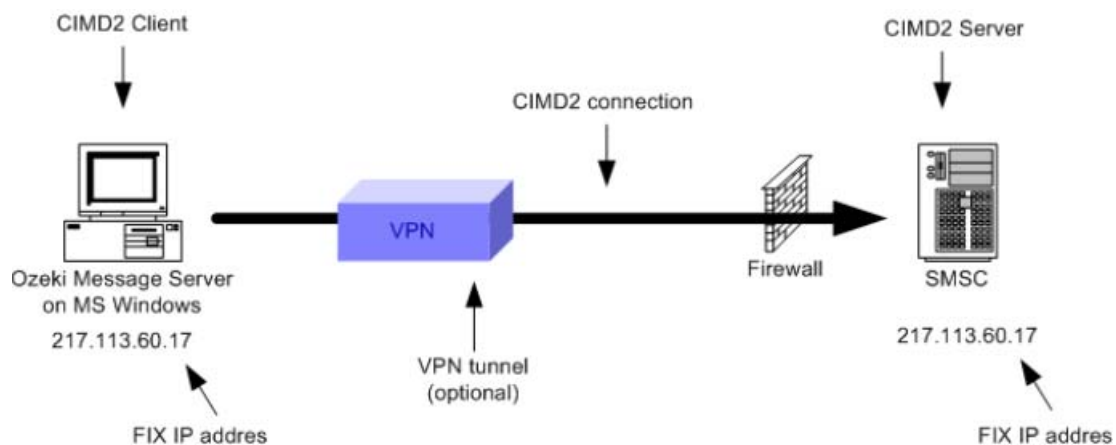


Figure 1 - CIMD2 connection with VPN protection

If you wish to use CIMD2 to connect to your service provider, you need to get the following information from the service provider:

IP address of the SMSC
Port number of the SMSC
User name
Password

Common protocols - Colour SMS

The Colour SMS protocol was developed by T-mobile Hungary for premium rate SMS services. It is an easy to use HTTP based SMS protocol. When a customer tries to access a premium rate SMS service by sending an SMS to a premium rate phone number, the SMSC transfers this message as an HTTP post request to the HTTP Server operated by the content provider. The content

provider returns a response to the HTTP request in XML format. This response is sent as an SMS message back to the customer. The HTTP server operated by the service provider is Ozeki Message Server 6 (Figure 1).

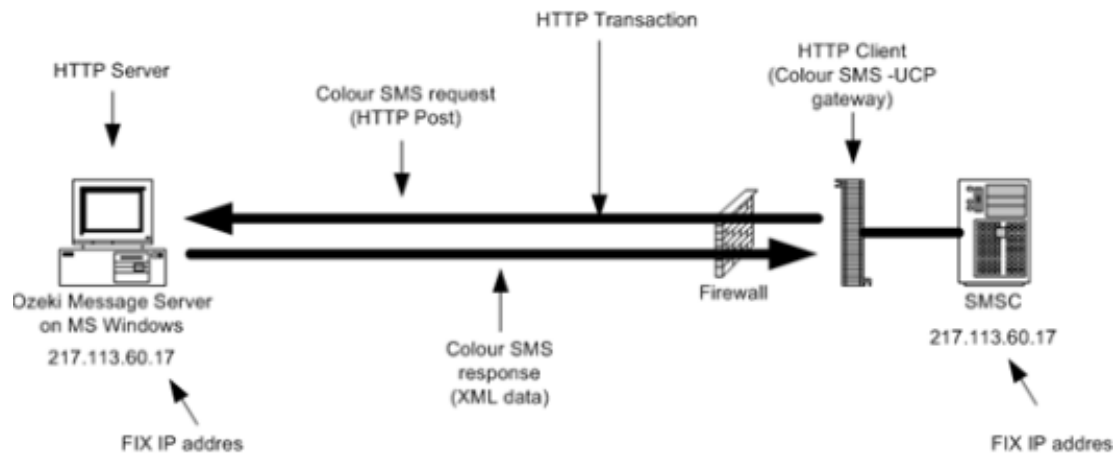


Figure 1 - Colour SMS transport

Due to the nature of the HTTP protocol, it is always required to return a response to incoming messages in a short period of time. The Colour SMS protocol is often used with other connection methods, such as UCP to make delayed responses to incoming messages possible. Using a UCP connection, response messages can be sent at a later time, which is useful if you would like to set up periodic SMS notifications, such as weekly news service.

Common protocols - Premium rated SMS services

Mobile Service providers often allow 3rd party content providers to create mobile services using premium rate SMS messages. The services can be games for prizes, tests, data services. Premium rate SMS messages make it possible to collect money from the users of these value added services. Since the amount of money collected is very small, these services are often referred to as micropayment services.

Today's most popular premium SMS services are downloaded games, logos, ringtones, background pictures and information services (e.g.: weather, news, programs).

These services work in the following way:

The person who would like to use the premium rate service, sends an SMS to a premium rate phone number. When the SMS arrives into the Short Message Service Center (SMSC) of the service provider, the SMSC recognizes that the destination phone number belongs to a "content server". As a next step, the message is transferred to a content provider's server through a TCP/IP connection. The content provider's server receives the SMS messages and generates a response SMS according to the implemented business logic. The response SMS travels back to the SMSC for transmission through TCP/IP just as

the request SMS was transferred to the content server. In most situations the content server and the SMSC communicate over the Internet (Figure 1).

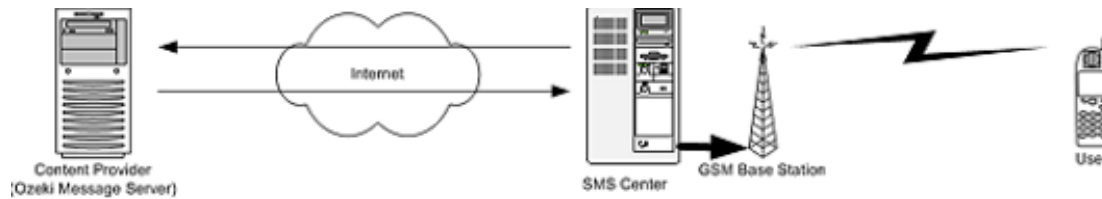


Figure 1 - Premium rated service architecture

The business logic of the content server can be simple or complex. One of the most common solutions is based on content interpretation. A simple algorithm compares the incoming message text to a set of keywords, and generates a response based on the keyword. Another common application collects the phone numbers of the received messages and uses these phone numbers to provide periodic information services, such as traffic report or weather report every morning. There are many other types of solutions as well. One thing they have in common, is that in every case an SMS must be sent back to acknowledge the service request within a limited response time (usually in 2 minutes).

Some useful facts:

- The premium rate SMS prices cannot be chosen freely. In most cases the content provider must select a price offered by the GSM service provider.
- One premium rate phone number can run several services, but can only operate with one price.
- It is not possible to send/receive premium rated SMS messages using SIM cards of foreign countries. It means that there is no such thing as premium rate roaming.
- When creating a solution, it is important to check if premium rated messages can be sent to our service from every GSM network in the country. Sometimes it is necessary to sign a contract with all GSM providers independently.
- Premium rated SMS service cannot be operated using a GSM phone attached to the PC with a phone-to-pc data cable.
- Premium rate SMS service phone numbers are often different than normal price phone numbers. They often use a special prefix, such as 06 90
- If there is no answer for the premium rate SMS the Mobile Service provider doesn't charge the customer.

VPN configuration on Windows 2003

This guide describes how you can configure an IPSEC/L2TP connection to a GSM service provider with preshared secret key to set up IP SMS connectivity:

http://pplan.calpoly.edu/plans_and_projects/ipsec.pdf

Service Provider Information

Our company provides detailed information to our customers about connecting Ozeki Message Server to any of the following service providers. The documentation contains the contact information of the administrative and technical people at these companies (names, direct telephone numbers, e-mail addresses, etc) and detailed instructions on how to set up the service. The set up documents contain step by step configuration instructions, communication protocol information, detailed protocol specifications and case studies along with the experiences and hardships we have faced when we have set up our own service to these providers. We also give working configurations for Linux and Windows on how to set up the VPN connections required by some of these operators.

[HUNGARY: Pannon GSM](#)

[HUNGARY: T-Mobile](#)

[HUNGARY: Vodafone](#)

[CZECH REPUBLIC: EuroTel](#)

[CZECH REPUBLIC: Oskar](#)

[CZECH REPUBLIC: T-Mobile](#)

Product features

The Ozeki Message Server Application is a Message Management System, that operates as a system service on Windows 2000, Windows XP and Windows 2003 computers. It is the successor of the successful Ozeki SMS Server 5 application. It can be used to send and receive SMS text messages and various other message types through the GSM network. Ozeki Message Server can be used by everyday people in the office, and by IT professionals.

Ozeki Message Server in the Office

Office users can use applications they are already familiar with sending SMS messages. For example [Microsoft Excel](#) can be used to send large numbers of messages with the help of an Excel Macro. Ozeki Message Server provides a plugin for Microsoft Outlook to make SMS messaging possible from the standard outlook interface and to be able to receive reminders about Outlook events. Of course there is a graphical user interface included in the software for those, who wish to use Ozeki Message Server on its own.

Ozeki Message Server in the hands of IT professionals

Ozeki Message Server was designed to help software developers and system integrators to add messaging functionality to their applications. The software supports SMS text messages and it can be used to send and receive other message types, such as zModem files. It provides connectivity plugins, and application programming interfaces (APIs), that make sending and receiving very easy from existing systems. One of the biggest advantages of the software, is that it has a powerful engine, that can handle large volumes of traffic. It can be configured to use more than one devices and connections to send and receive messages with load balancing and it provides advanced management of message queues. The software provides SMS connectivity via GSM phones attached to the PC's serial port and over TCP/IP links connected directly to the Short Message Service Center (SMSC) of the GSM service provider. For IP SMS connectivity the UCP, CIMD2 and the SMPP protocols are supported.

Ozeki Message Server in the Office

Ozeki SMS is a reliable mobile messaging platform, that enables intelligent delivery of relevant information for people using GSM handsets. It is a cost-efficient two-way messaging system for your office. It can be used for various purposes whether you are a small, medium size or large company. The software provides a friendly, easy-to-use interface that makes it possible to send and receive SMS messages. It also has a lot of pluggable modules that help if you want to do messaging from other applications.

If you use it, you and your employees will not forget any important events, vital information will reach you as fast as possible and your marketing and customer

relationship management team will perform better.

The "Out-of-the-box", office edition of the Ozeki Message Server application is capable of handling SMS messages in a similar way to your e-mail client. It provides functionality for sending, receiving, forwarding and replying to messages. It maintains a history of all the messages sent and received by the system and can be used in an office in two ways. It can be used in a single user mode from a standalone computer, or it can be used in the office network from several computers simultaneously (Figure 1).

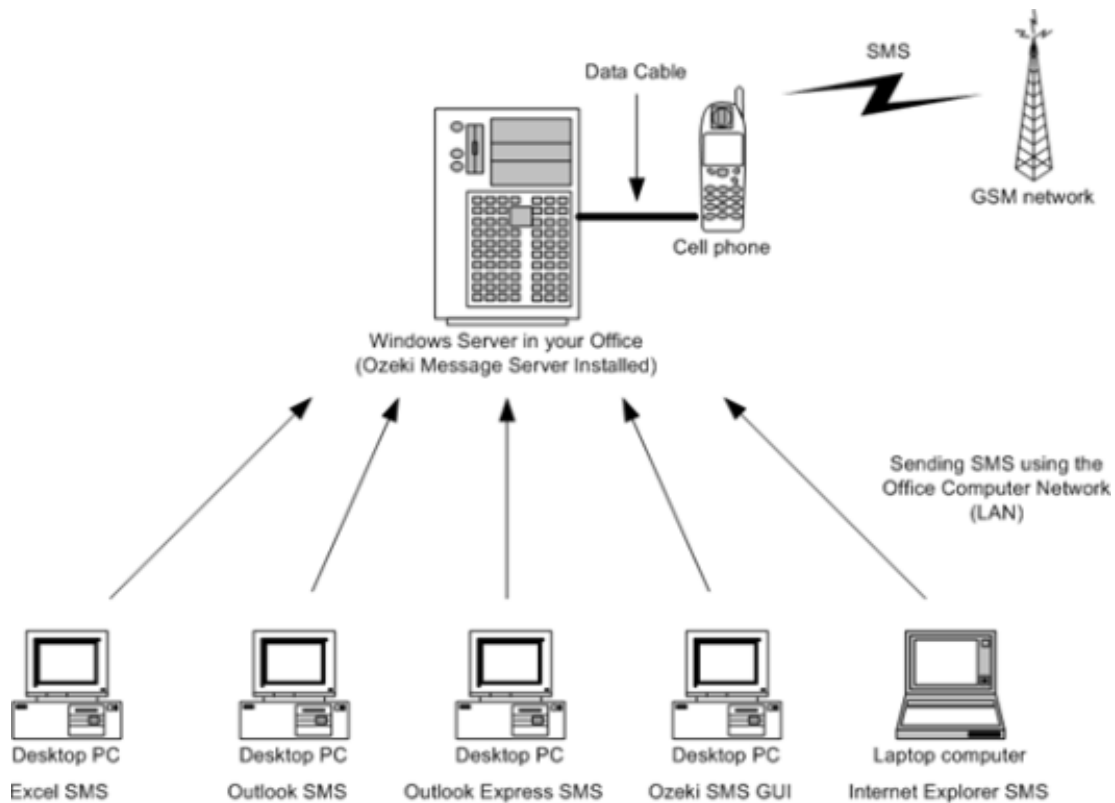


Figure 1 - Ozeki Message Server in the office

If you have a computer network in your office, you can use Ozeki Message Server to send SMS messages from any PC. After installation user accounts can be created in the server, and these accounts can be used to attach Microsoft Excel, Microsoft Outlook, and the other office clients to the server to send large or small number of messages through the office server, that has a mobile phone attached to it.

Basic features

Ozeki Message Server 6 provides the following basic features for Office users:

Intelligent Message Sending	You can send SMS messages from the built in Ozeki Graphical User Interface (GUI) or from MS Excel or MS Outlook. In all of these cases the outgoing messages are collected into a message queue, and sent as fast as
-----------------------------	--

	possible. In certain situations, for example on New Year's Eve, the GSM network fails. In these cases it is not possible to send SMS messages. The intelligent message sending capability of Ozeki Message Server helps in these situations. If a message cannot be sent for any reason, the software tries to re-send it a couple of times. Between each retry a certain amount of time (e.g. 10 minutes) is spent, to wait for the network to be available again. If there are six failures, the message is considered undelivered and the users are notified.
Receiving Messages	You can receive SMS messages in the Ozeki GUI and in MS Outlook or MS Outlook express.
Message Forwarding	The built-in user interface makes it possible to forward any incoming message to other phone number(s).
Message Replies	The reply feature of the built in GUI is similar to the e-mail clients' user interfaces. It is very easy to reply to any incoming SMS message.
Message Templates	To speed up work, you can define default messages in advance. These predefined messages are called message templates. When you want to send an SMS you can choose a template and edit it.
Address book	Ozeki Message Server 6 provides a central addressbook for all clients. This address book can be synchronized with an SQL database or an LDAP directory. You can read more about this feature on the addressbook page .
Group Messaging	If a large number of SMS messages with the same content needs to be sent, the group messaging feature provided by Ozeki Message Server can make your job easier. You can send messages to a lot of phone numbers (up to a few thousand) with only a few clicks.
Message History	Log files created by Ozeki Message Server, you can keep track of your messages anytime. A list of all messages, which have been sent and received by you are saved in a separate log file. This file can help you to remember what have you sent, when and to whom your message have been sent to.
Message Folders	In case of having a lot of incoming messages it is a good idea to organize them into folders. Ozeki SMS provides an option to create new folders and to save your messages to them.
Integrated Printing	You can print your messages and log files to your default system printer.
Multiple Device Support	For one GSM module it takes around 5-6 seconds to send an SMS message. You can speed this up by attaching more than one module to your computer. If you have two phones attached to send messages, you get twice the speed. Of course there is an option to switch to IP SMS technology seamlessly if the capacity of standard GSM phones are insufficient.
Integrated Microsoft Excel Support	You can use the built-in Excel support of Ozeki SMS. This allows you to send your SMS messages directly from Excel, simply by pressing a button. This feature is created with an Excel Macro that loads all the messages into the server and uses your Excel worksheet as the addressbook.
MS Outlook Support	You can use an Outlook plugin or you can use a standard e-mail client (e.g. Microsoft Outlook Express) to send SMS messages. If you have a computer network, equipped with several clients, this option provides a comfortable

	way to add SMS-messaging using your existing e-mail infrastructure
Remote User Interface	If you have Ozeki Message Server installed on one of the computers in your office, you can install its remote user interface to other computers in your network. The remote user interface makes it possible to send and receive messages by anyone who has a user name and a password configured on the server.

Multi-user architecture

An important element of the Ozeki Message Server, is that it has a multi-user architecture. Each user that wants to send out messages connects to the engine with a user name and password. More than one user can use the SMS service simultaneously, but they cannot disturb each other, by reserving too much system resources. Each user has a separate message queue, thus they all get the same priority when they want to access the sending devices. For example one user can place 10 000 outgoing messages into his outbox for delivery, and another can place 3 messages just after this. When the message engine decides which message to deliver first, it will take one message from the queue of 10 000 messages, and the next one will be taken from the queue of 3 messages. This way the user who wanted to send out only a few messages will not have to wait for the 10 000 messages to be delivered.

MESSAGE PRIORITIES

Ozeki Message Server 6 has a built in priority system to support simultaneous user access. This is very useful if you use the service for BULK SMS messaging and other everyday tasks. The priority system is based on message queues. Each user has its own message queue. And the engine takes one outgoing message from each queue if more than one queues have outgoing messages (Figure 1).

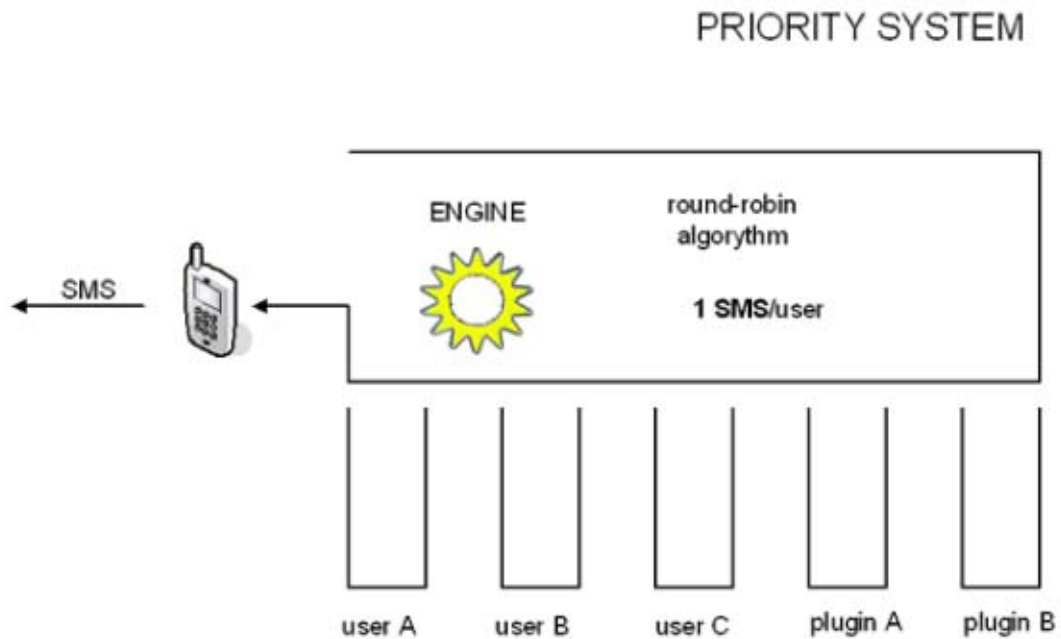


Figure 1 - Message queues

It is important to mention, that each plugin has its own user account as well. This means you can have various messaging processes supported by plugins sharing the same resource. You don't have to worry about one of the users or plugins overloading the system. If you attach more than one GSM devices you can increase the capacity of the server. In this case you can use operator names to define message routing.

Here is an example, that describes how this works:

If user A inserts 1000 SMS messages into his outgoing queue, and user B inserts only 1 SMS. The SMS of user B will not wait for the 1000 SMS messages to be sent out. Instead it will only wait 1 SMS to be sent by user A. It will be the second SMS in the delivery chain.

Addressbook

Ozeki Message server uses a central address book to store phone numbers. The central addressbook is stored on the server and it helps to find phone numbers of contacts previously entered in your system. This addressbook can be accessed by all clients. If the Outlook plugin is installed, the Outlook addressbook can also be used to select the destination phone numbers when composing SMS messages. Often it is possible to synchronize your phone with outlook. In this case all your telephone numbers, saved to your computer, will be available in OZEKI SMS as well.

The addressbook service of Ozeki Message Server 6 can get records from an SQL database and an LDAP directory as well.

How IT Professionals Use the Software

Ozeki SMS is a computer program with an open architecture. It can be attached to existing business applications or it can be used on its own. Ozeki SMS was developed to make it easier to send information to mobile phones both automatically and manually. The offered interfaces make it possible to use it from any software system and programming language. It supports software developers using different environments.

SQL-SMS gateway

One of the most popular scenario of usage requires a GSM Mobile phone and an SQL database server. In this case the user connects the GSM phone to his PC with a serial (or USB) data cable. After this is done he creates two database tables in his database. One of the tables called ozekimessageout will be used for sending and the other called ozekimessagein will be used for receiving. He then configures Ozeki Message Server 6 to do the message routing between the database tables and the GSM Mobile phone. After the configuration is done the user can send a message by simply inserting a record into the ozekismout database table. If an incoming message arrives it is automatically saved into the ozekismsin table.

HTTP event notification

System administrator often uses Ozeki Message Server to monitor the activity of their IT network. The typical way they do this is that they create small scripts for system monitoring and use an HTTP Client (command line tool), such as wget on Linux to send SMS messages when the server behaves unexpectedly. The HTTP client makes an HTTP GET request to the SMS server. The request contains all the information including the recipient telephone number and the message text. The Ozeki Message Server uses this information to send the appropriate SMS message.

The envelope system

Ozeki Message Server is a messaging software that can handle various message types, such as SMS text messages, SMS binary messages, WAP messages, etc... . The addressing scheme, the content and other properties of these messages are different for each message type.

Ozeki Message Server provides store and forward functionality and makes it possible to track each message individually. It also exposes enough information to find out when and how the message was sent or received. To make this functionality work for all kinds of messages, the software uses an envelope system similar to the envelope system used by your postal service. When the user passes a message to the software, the Ozeki application puts it in an XML envelope.

The message engine can read and handle these envelopes regardless of their content. Based on the addressing and message type identification printed on the

envelope, the engine can route the envelope to the appropriate destination. Envelopes can contain almost any message type. SMS, E-mail, MMS, Windows MSN messages, ICQ messages, although not all of these types are implemented in the current software version.

Link and message management

The Ozeki Message Server is a software designed to operate 24 hours a day and 7 days a week.

In a non-stop environment various situations need to be handled. For example the network link between the SMSC and the server can be broken or a GSM phone attached to the PC can stop responding after a certain amount of time, because its battery died. To handle these situations, the software uses advanced link management. This means, that the message engine constantly monitors the transport links and it takes action if the link is broken. For example if a TCP/IP network connection is lost, it automatically reconnects the SMSC. If the software notices that the network link is up again, it performs the necessary login sequence required by the SMSC after reconnection.

Another feature introduced to increase reliability is advanced message management. If a message cannot be sent for any reason (there is no GSM network coverage, or the SMSC is overloaded, etc), the engine will try again at a later time. The basic algorithm used can be adjusted. By default the engine tries to send a message six times before it marks it undeliverable and there is a one minute period between the tries.

Message routing

A basic feature of Ozeki Message Server 6 is message routing. It collects SMS messages from various sources, such as a user typing in a message using the built in graphical user interface (GUI), a database or an external application. Once the messages are collected they are organized into message queues in the message engine. The message engine (Figure 1), handles the sending devices. Each sending device whether you use a GSM modem or an IP SMS connecting has a driver assigned. If more then one devices (or connections) are used for sending messages more instances of the drivers need to be installed.

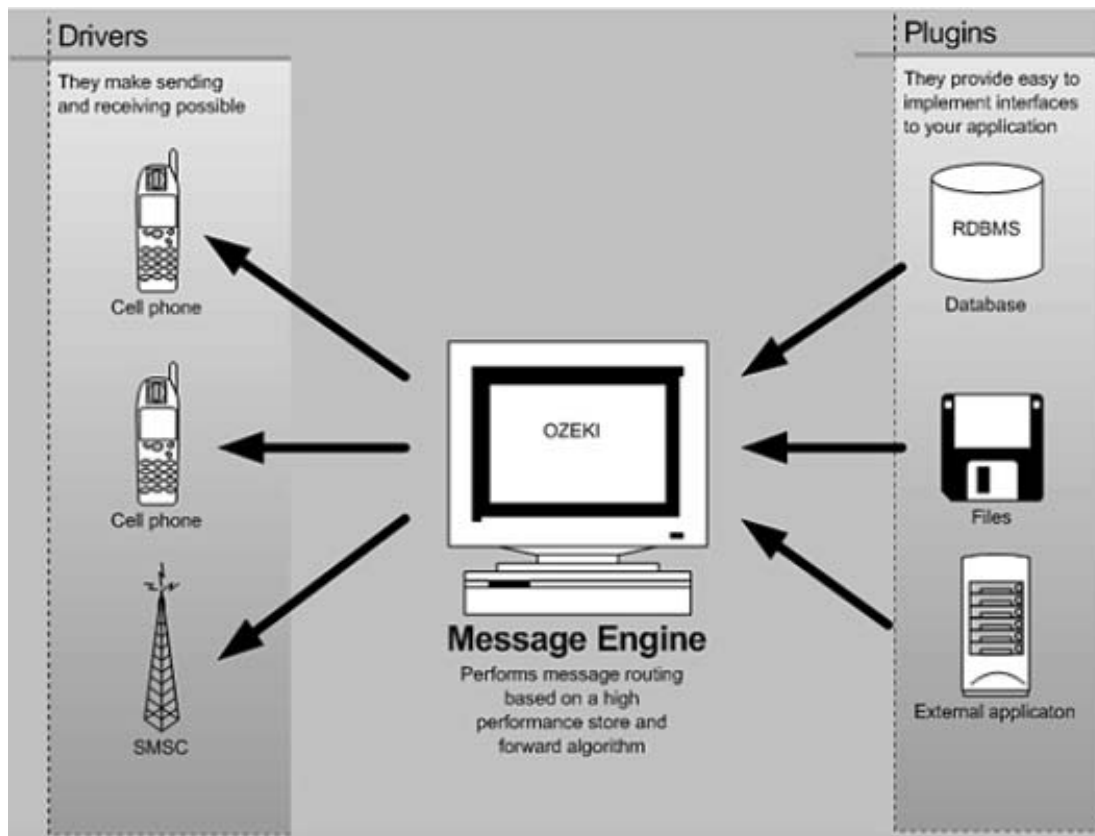


Figure 1 - Message Engine

When you install a driver, it can be seen in the Server Manager Application below the folder options. If more than one connections are installed, more than one instances are displayed (Figure 2).

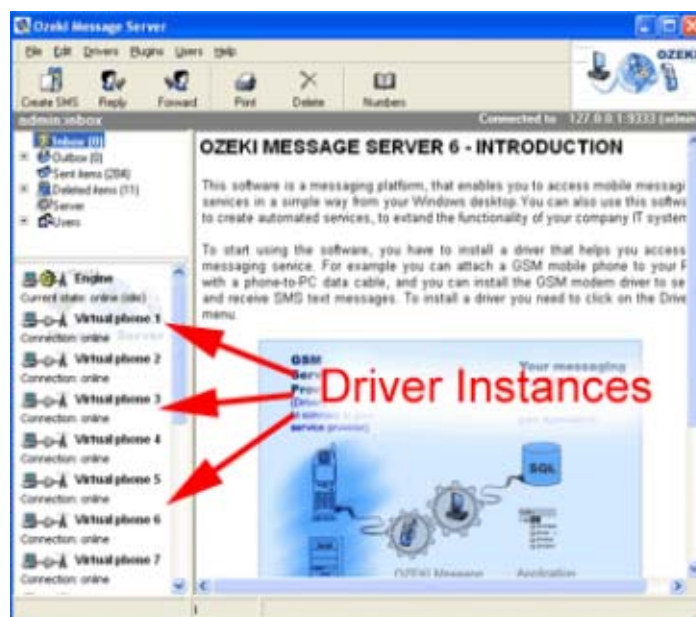


Figure 2 - Driver instances

When you install a driver, you configure it using the driver configuration form. On this form there is a special field, called **operator name** (Figure 3). An operator name is a unique name identifying the driver. You can put any name you like, in this field. The operator name has a key function in message routing. It can be used to tell the engine which driver should be used to send a particular SMS message.

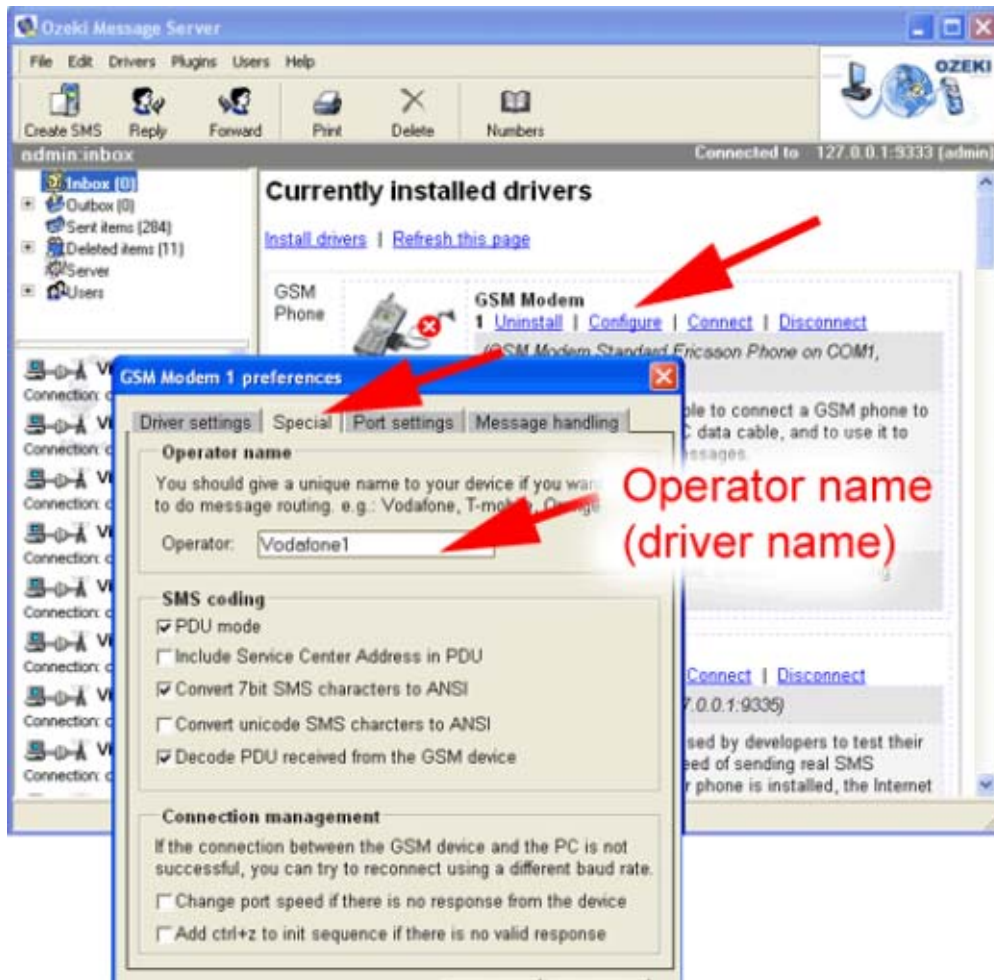


Figure 3 - Operator name

If you have more than one driver installed, you can select which driver you wish to use by selecting the appropriate operator name in the user interface when you compose the message (Figure 4). If you use a database plugin you have a special field set up in the outgoing message table called operator name. In this field you can put the name of the operator.

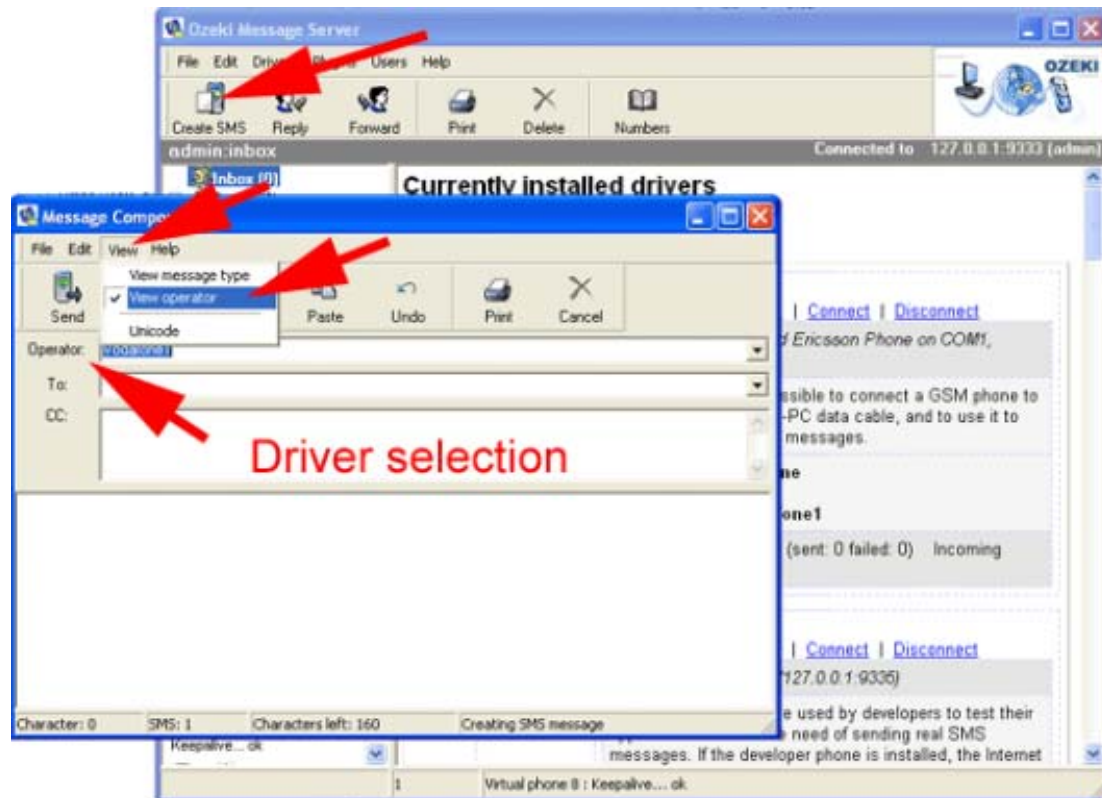


Figure 4 - Composing a message

The user properties form and the plugin configuration forms always contain an operator name field. If you specify this field, it means the user or plugin will only send outgoing messages using this connection

Load balancing

Load balancing can be achieved by using the same operator name for more than one drivers.

Installation / administration guide

This section contains useful technical information about the installation, configuration and maintenance of the software. It is intended for IT personal familiar with Windows systems.

The chapter includes information about the system requirements to run the software on Windows 98, Me, NT, 2000, 2003 and XP operating systems. It provides information on how to set up a standard GSM mobile phone to transmit and receive SMS messages and on how is it possible to configure IP SMS services.

Installing the server

[Ozeki Message Server](#) is a software-based solution designed for Microsoft Windows (Figure 1). The installation of this software follows the standard Windows installation guidelines. Basically you have to start the installation by downloading the install package from our website. The downloaded package is a simple .exe file, that can be started, by clicking twice on it, in Windows Explorer.

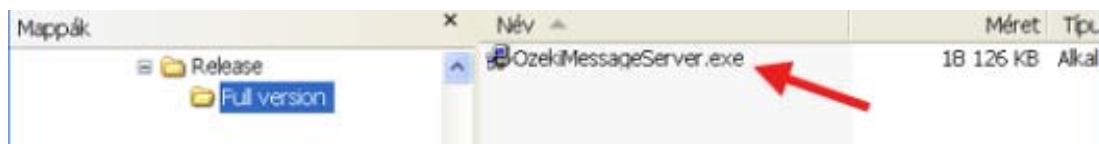


Figure 1 - start the installation

During the installation, various selections can be made. You will find step-by-step information in the following section. If you want to quit the **Setup** program without installing or uninstalling, just click **Cancel** and the **Exit Setup** window will offer you the possibility of quitting the **Setup** program. When the installation completes clicking **Exit Setup** will simply terminate the **Setup** installation. If you want to rerun **Setup** after having quit the installation before it had been completed, follow the installation steps normally as if you were installing the program for the first time.

The first step in the installation procedure is going through the list of [hardware and software requirements](#).

Hardware and software requirements

To install and run the Ozeki Message Server , you need the following:

Hardware requirements	
Suitable GSM phone or GSM modem*	Every GSM mobile phone, that has a built in modem supporting the GSM 7.05 SMS standard can be used.
Data connection	A phone-to-pc data cable or an Infra Red port (IR)

	or a Bluetooth connection
Intel compatible PC	Pentium III, 500 Mhz or better CPU 40 Mb free hard disk space 512 Mb of RAM
Software requirements	
Operating system	Windows 2000 Windows 2003 Windows Xp
Ozeki Message Server	Ozeki Message Installation Package (CD or downloaded file)

* If you use IP SMS, you don't need a mobile phone

Installation steps

The most common way to send and receive SMS messages from a PC is by attaching a mobile phone to it using a phone-to-PC data cable. The installation steps in this page show you how this can be done. If you plan to use an IP SMS connection, you should skip to step 3: "Installing the application" and you must install an IP SMS driver (e.g.: SMPP, UCP, CIMD2) instead of the GSM modem driver. Concerning the installation there are 8 simple steps:

1. Setting up the GSM phone.
2. Connecting the GSM phone to the computer.
3. Installing the application.
4. Starting the application.
5. Logging in to configure the service.
6. Installing a driver
7. Configuring the installed driver.
8. Starting the connection and sending your first SMS.

Step 1: Setting up the GSM phone

In order for the message server to work properly the following settings should be made on the mobile phone, prior to installation:

- All the PIN codes and security locks should be switched off
- The voice and data calls should be redirected to another phone number
- All SMS messages stored in the phone or the SIM card should be deleted.
- The SMS Service Center should be set

When all these settings are done, you should test your GSM device by sending and receiving SMS messages manually.

Step 2: Connecting the GSM phone to the PC

In order to send or receive SMS messages from your PC, you need to attach your phone to the PC with a phone-to-pc data cable ([Figure 1](#)). The cable is attached to the serial port of your computer. The serial port is located on the back side and has 9 pins.



Figure 1 - Attaching the device to the PC with a phone-to-pc data cable

If you want to use Infrared (IR), you must enable your infrared port on the phone. (Sometimes you have to turn it on in the PC as well). When this is done, you should place the phone within the range of the IR receiver on the PC. When the PC and the phone begin to communicate over IR, you will see an icon in the bottom right hand side of the screen (Figure 2).



Figure 2 - Attaching the device to the PC with IR connection.

Step 3: Installing the application

The Ozeki Message Server installation is similar to any Windows applications. Before you do the installation, you must exit all Windows applications. After having done it, locate the **OzekiMessageServer.exe** file on the installation CD-ROM. If you have downloaded the application from the Internet, **OzekiMessageServer.exe** is in your download directory. Start **OzekiMessageServer.exe** by double-clicking on the icon and follow the instructions:

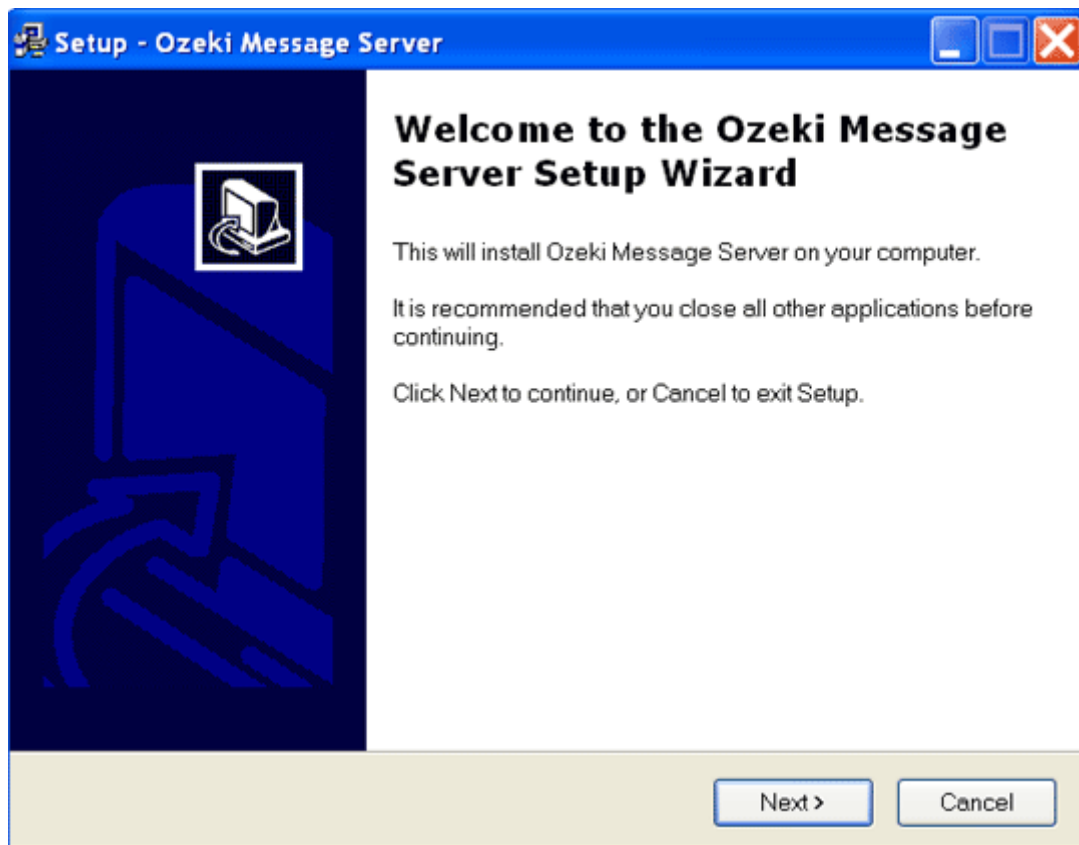


Figure 3 - Starting the installation of Ozeki Message Server

On the Welcome screen click **Next** to start the Installation (Figure 3). This will bring up the End User License agreement of the software. It is important to read this and to accept the agreement if you wish to use the software. If you have accepted the End User License Agreement, you will move to the destination directory selection form. On this form, you have to specify the directory you wish to use. Make sure you have at least 20 Megabytes of hard disk space available on the drive you have chosen. Following this you will be faced with some options about icon creations. After the installation is finished, a shortcut will be available in the Windows **Start** menu and a taskbar icon will appear. This shortcut is a reference to the Ozeki Server Monitor. The monitor application can be used to get information about the state of the server and it helps to open the Ozeki Server Manager configuration screen.

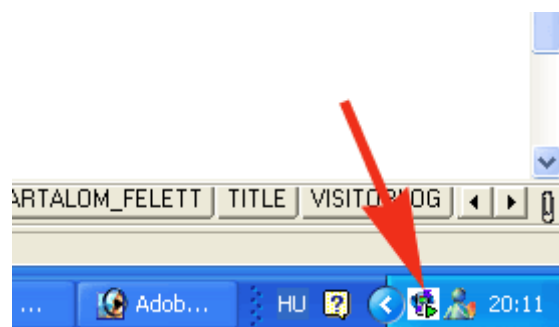


Figure 4 - Starting up Server monitor

The configuration screen can be opened by clicking on the Manage Service menu item (Figure 4).

Step 4 Logging in to configure the service

By clicking on the menu item, a login window comes up. This login window requires a user name and a password. If you didn't specify a login name and a password during the installation, you can use the built in administrator account. Username: admin, Password: abc123. We suggest to change the default password immediately after login!

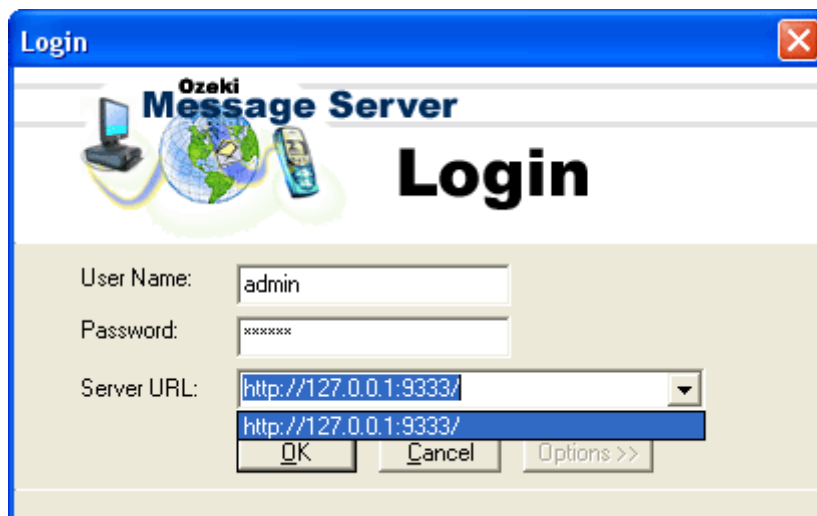


Figure 5 - Login

Login into Ozeki Message Server (Figure 5).

Step 5 Installing drivers

Once you have logged into the Server manager, you can installing one or more drivers. A driver in Ozeki Message Server gives handles the attached GSM phones and GSM modems and the IP SMS services. In this example we install the GSM modem driver, because we want to set up a GSM phone attached to the PC. In case of IP SMS connectivity a different driver needs to be installed in a similar way.

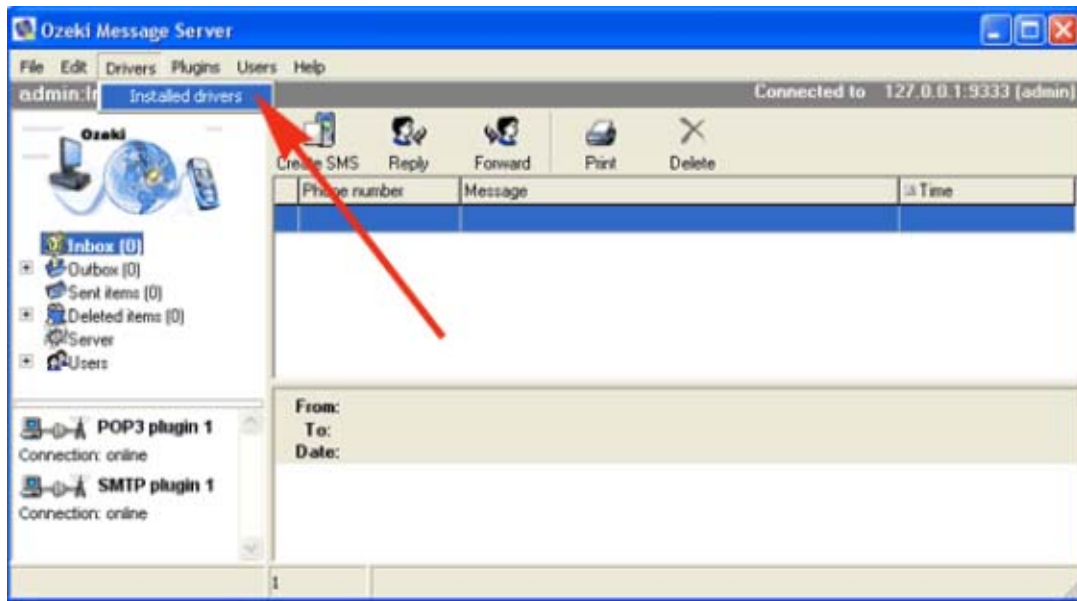


Figure 6 - Installation of Drivers

New drivers can be installed by choosing **Installed drivers** menu item from the **Drivers** menu. This brings up the "Available drivers for installation" page, where a short description of all the available drivers can be found (Figure 7).



Figure 7 - Installation of GSM Modem

In the "Available drivers for installation" window click on the **GSM Modem** [install](#) link and the installation process will start. If you want to set up IP SMS connectivity you should select one of the IP SMS drivers (SMPP, UCP, CIMD2). After you click on the install link, a small animation will appear for a short time signaling, that the driver installation is in progress. Wait until the installation process finishes.

Step 6 Configuring the driver

After the driver installation has completed you'll see the **GSM Modem 1** section appearing on the list of drivers in the "Currently installed drivers" screen. On this page you can start the configuration by clicking on the [Configure](#) link (Figure 8).



Figure 8 . - GSM modem configuration

After clicking on the link, the GSM Modem 1 configuration form comes up. You can set all the necessary settings on this form in order to send and receive messages. As a first step select the phone model you have attached to the computer, then specify the serial port, then click on Autodetect. (If you use IR or Bluetooth or USB, you have to find out the virtual port number and use it instead of the real port.). After selecting the phone model you should specify the name of the operator. The operator name is the name of the GSM company your subscription belongs to. (For example it can be Vodafone, T-mobile, Orange, etc.) When this is done, enter the telephone number of the phone attached to the computer and the SMS Service center address.

Important: phone numbers should be entered in international format e.g.: +36209937723 (Figure 9)!

The Service Center address should also be specified in this for. This address is provided by your GSM service provider. For example, in Hungary you would select one of these numbers . A list of service center addresses can be found in Appendix "C".

T-mobile	+36309888000
Pannon GSM	+36209300099
Vodafone	+36709996500

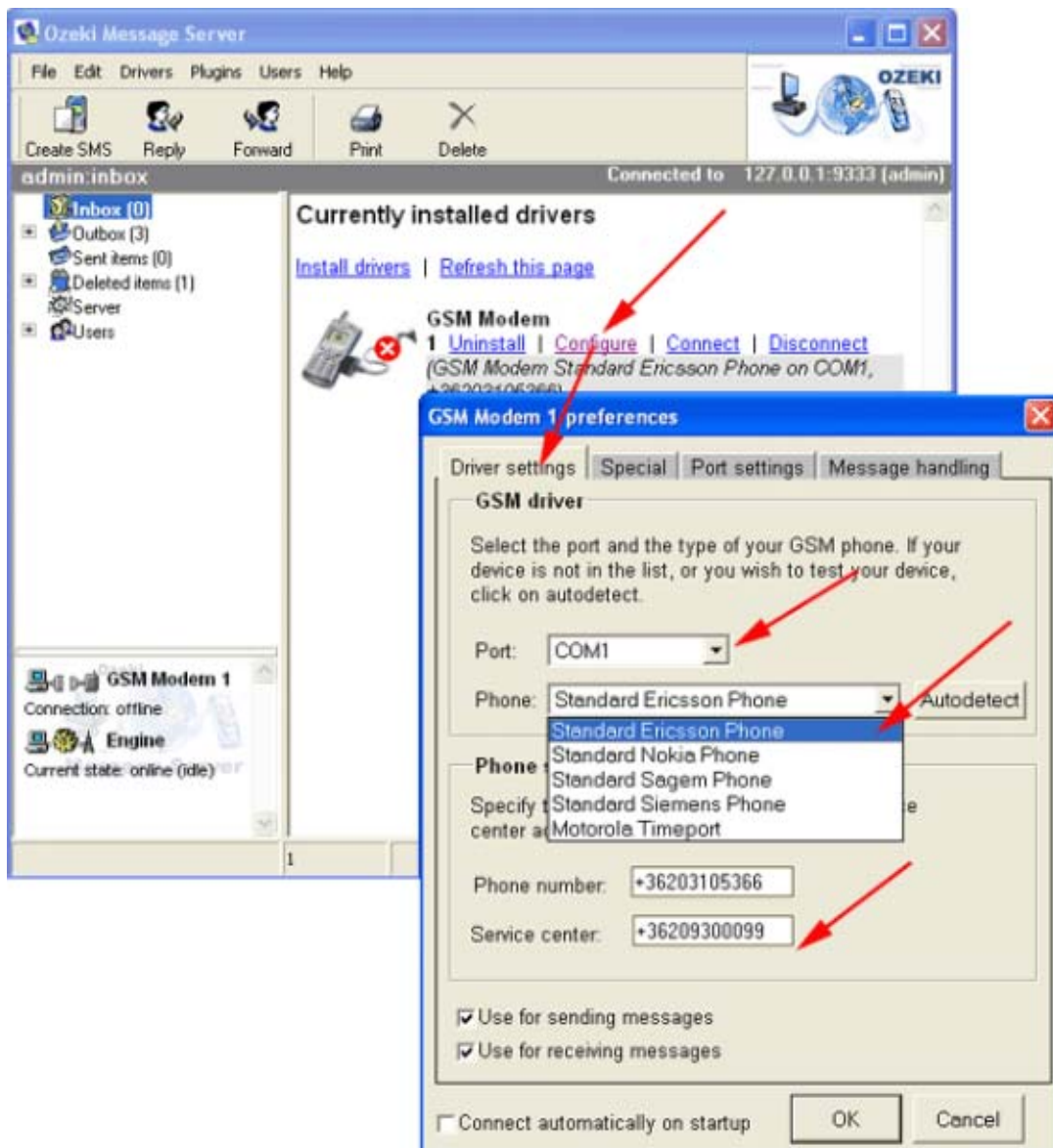


Figure 9. - Configuration of GSM modem

Step 7. - Starting the GSM modem connection

After the driver has been installed, you need to start it. The startup process means that the driver connects to the configured mobile phone. You can start the Ozeki Message Server GSM driver by clicking on "Connect" in the "Currently installed drivers" screen. After you have clicked on the [Connect](#) URL, you may use the [Refresh this page](#) link to check the connection state (Figure 10).

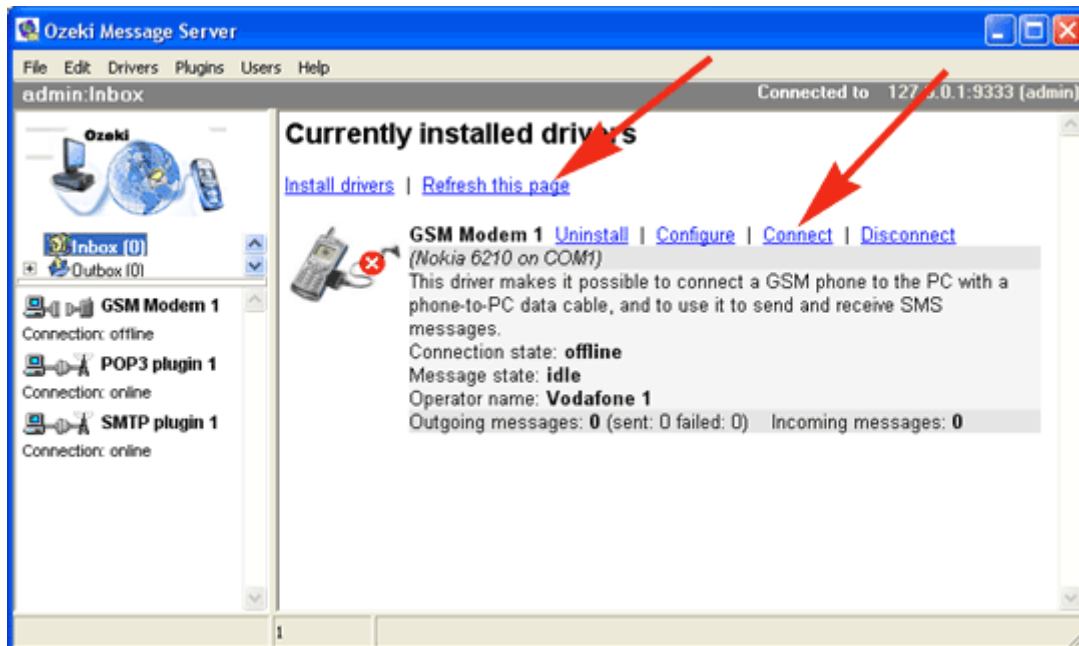


Figure 10. - Connection of GSM modem

Sending your first SMS

Once the application is installed and configured properly, you can send an SMS by clicking on the Create SMS icon. This brings up the SMS composer, where you can specify the phone numbers and the message text (Figure 11).

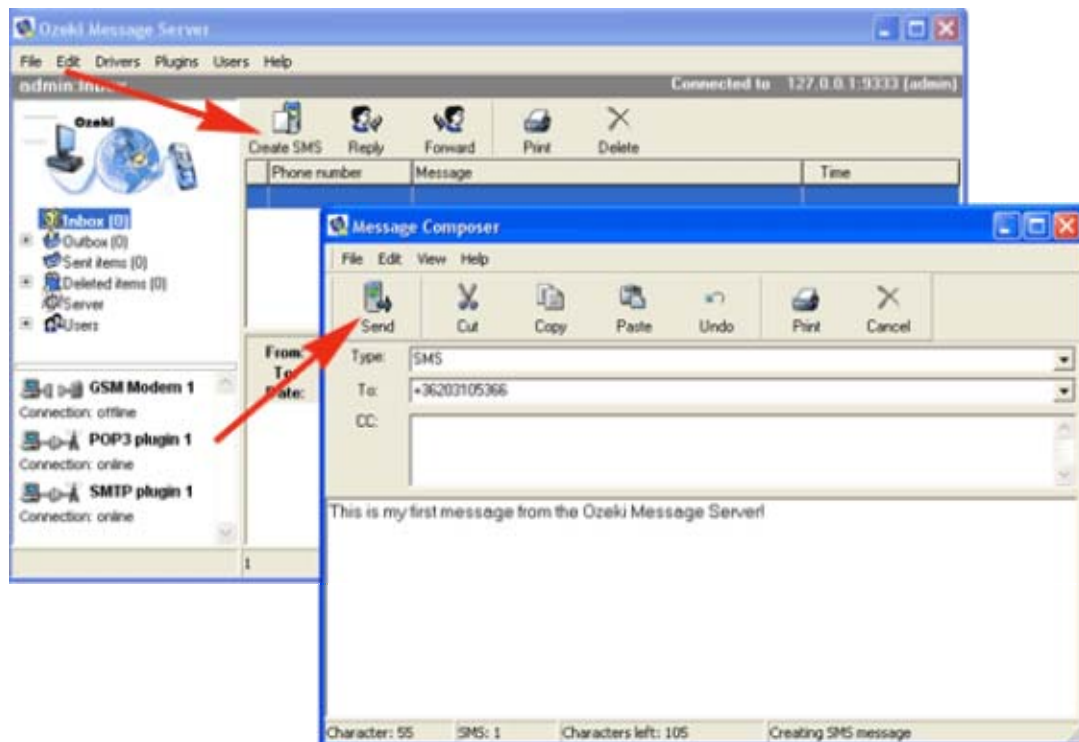


Figure 11. - Sending SMS to the recipient

Product activation

To activate Ozeki Message Server, you need to open the Registration form from the help menu (Figure 1).

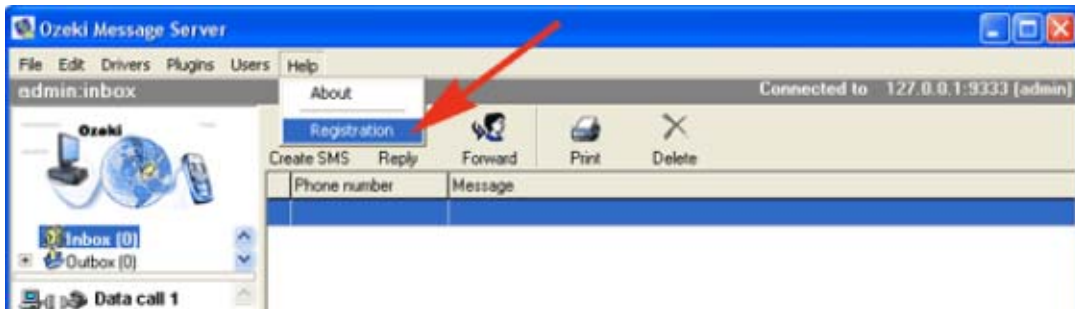


Figure 1. - Opening the product registration form

On the registration form, please enter your serial number into the "Serial No." edit box, select the Internet radio button and click Activate (Figure 2). When you click Activate, the software will contact our product activation server (www.ozeki.hu) over the internet and verify your serial number.

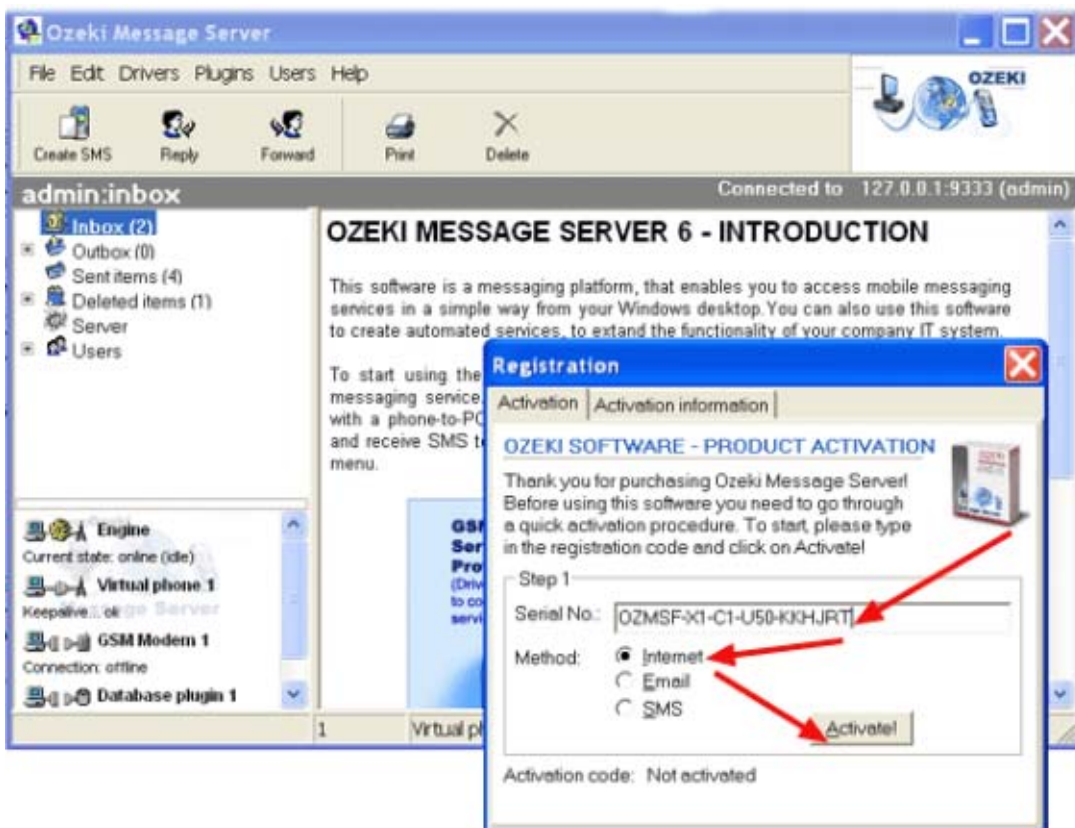


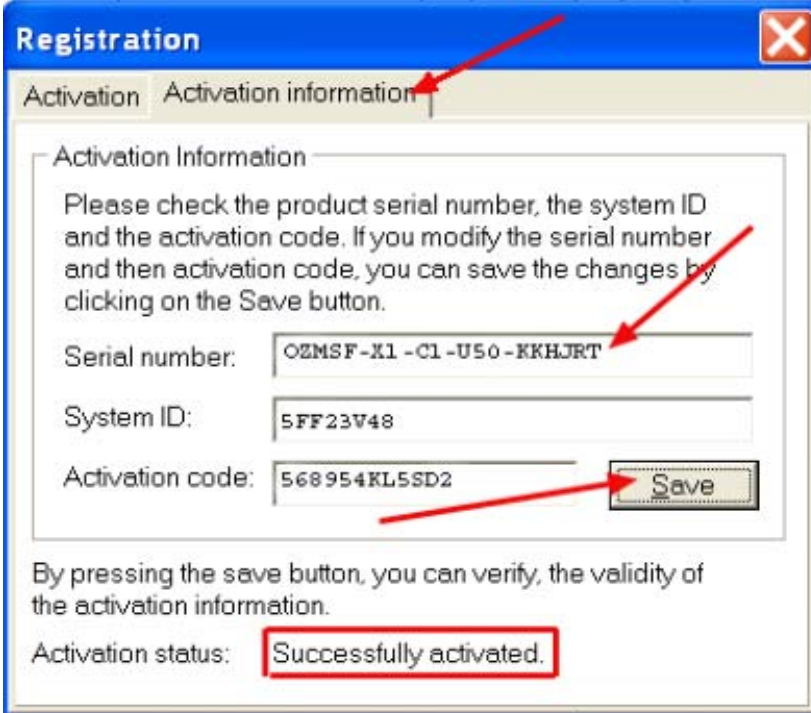
Figure 2. - Activation over the Internet

If your serial number is found in our activation database, the product activation procedure can be finished by clicking on the "OK" button. When the product is activated the service will **be shut down automatically!** . The last step of product activation is the restarting of the service.

Manual product activation

If the automatic activation is not available in your network, because you have a firewall or you are not connected to the Internet, you can use the manual product activation option. To activate Ozeki Message Server manually, you need to open the Registration form from the help menu. In the **Registration** form, select the **Activation Information** tab to figure out your System ID. After you have the System ID, send an e-mail to info@ozeki.hu with your System ID and your Serial number. Within 24 hours we will send you a response e-mail containing your activation code. Once you have your activation code you should enter it in the registration form and click Save (Figure 1).

Note: The system ID is generated automatically, based on the hard disk serial number or some other hardware serial number of your system. If you install the software onto another PC you need to get a new activation code!



The screenshot shows a window titled "Registration" with a blue header and a close button (X) in the top right corner. Below the header are two tabs: "Activation" and "Activation information". The "Activation information" tab is selected. The main content area is titled "Activation Information" and contains the following text: "Please check the product serial number, the system ID and the activation code. If you modify the serial number and then activation code, you can save the changes by clicking on the Save button." Below this text are three input fields: "Serial number:" with the value "OZMSF-X1-C1-U50-KKHJRT", "System ID:" with the value "5FF23V48", and "Activation code:" with the value "568954KL5SD2". To the right of the "Activation code:" field is a "Save" button. Below the input fields is a note: "By pressing the save button, you can verify, the validity of the activation information." At the bottom, the "Activation status:" is shown as "Successfully activated.", which is enclosed in a red rectangular box. Three red arrows point to the "Activation information" tab, the "Serial number:" field, and the "Save" button.

Figure 1. - Activation information

After product activation you need to restart the service.

Restarting Ozeki Message Server 6

To restart Ozeki Message Server 6, you need to click on the task bar icon with the right mouse button. In the popup menu that appears after this event, you need to select the **Stop Service** menu item (Figure 1). The taskbar icon will indicate when the service is stopped. (A red rectangle will replace the green triangle.)

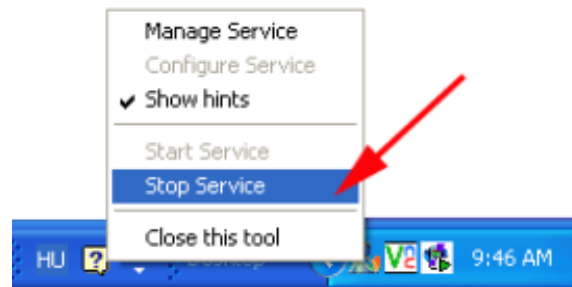


Figure 1. - Stop Service

After the service has been stopped, you can restart it by selecting the Start Service link from the same menu. If the startup is successful a balloon hint will notify you that the configured drivers are connected .

Server Configuration - Server Preferences

After you have installed Ozeki Message Server 6 and have configured a driver, you might want to take a look at the server preferences form. This form offers options to optimize and customize the software according to your needs. The form can be opened from the Edit menu by clicking on the Server preferences menu item (Figure 1).

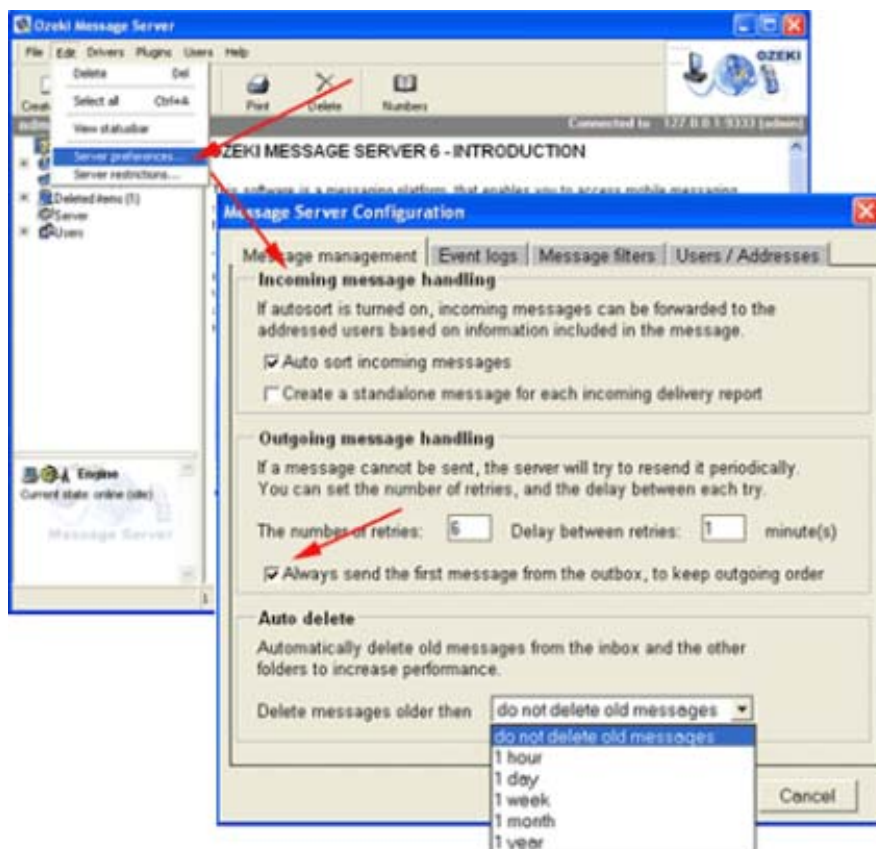


Figure 1. - Server preferences form.

In the server preferences form, on the **Message management** tab you have three sections. The first section is for "**incoming message handling**". If you place a checkmark into the checkbox, is called **Auto sort incoming messages**, that incoming messages will be forwarded to the addressed users, if the first word of the message matches the username. Otherwise the messages are forwarded to the members of the inbox group.

In the "**outgoing message handling**" section, you have the option to set how many times the message server should try to resend those messages, which cannot be sent on the first try, furthermore you can configure delay between each try. There could be several reasons why a message send attempt might fail, the most common reason is that GSM network coverage can go away for a short time.

In the outgoing message handling section, there is another checkbox you can use. If you place a checkmark in the checkbox called **'Always send the first message from the outbox, to keep outgoing order'**, it means, that messages will be sent in the same order they were placed into the outbox. Please note, that this will create lower performance, because in case of send failures, the message server will keep on trying to send the first message, until the maximum number of tries and not move on to sending other messages.

In the final section you can configure the maximum age for messages in the Message Server. You can choose a one of the following possibilities:

- do not delete the messages
- delete messages older then 1 hour
- delete messages older then 1 day
- delete messages older then 1 week
- delete messages older then 1 month
- delete messages older then 1 year

If you are running a service 24/7, it is wise to turn on automatic message deletion, to keep the size of the message queues small, to have better performance. (Not to mention disk space usage).

The second tab on this form is the **Event logs** tab. On this page you can configure the logging of various events. It is very useful to configure logging to be able too keep track of the workflow, and to discover errors .

The third tab is the Message filters tab, that allows you to select filters for incoming and outgoing messages. A message filter can modify the content of the message before processing. In most situations you will not configure any filters.

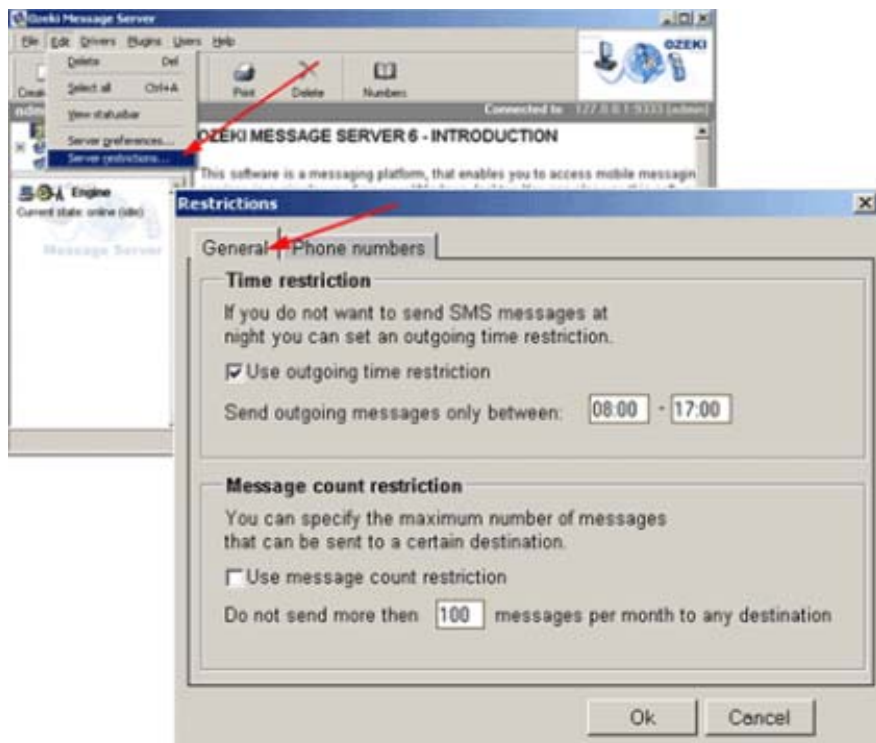
The fourth tab is the **Users/Addresses** tab. On this tab you can select external authentication service providers, which verify the validity of users and passwords. If you place a checkmark into the checkbox is called external authentication service provider, the program will use the selected .dll to perform the authentication. Make sure you configure the provider you wish to use. In addition to this you can also select an addressbook provider. The addressbook .dll-s included in the package make it possible for you to find contact information

in a text file, in an SQL database server or in an LDAP directory. All you have to do is select the appropriate DLL and click on the Configure button.

Server restrictions

In addition to the server preferences form you can perform further configuration using the Server restrictions option. The restriction function is very useful, because you are able to regulate how the outgoing messages are sent. For example you can specify a time interval when outgoing messages are sent to make sure you don't disturb your customers at midnight with automatically generated reports. On this form you can also specify the maximum number of messages the software will send out to a particular telephone number in a month.

The server restriction form has two tabs: The General option tab and the Phone numbers tab. The settings on the general option tab are self explained. In the phone numbers tab you specify a list of telephone numbers and prefixes, you don't want to communicate with using this server software. When you are adding a forbidden telephone numbers to the list, make sure you use international format. Phone number prefixes can be defined using an asterisk, for example: 0690*, The list items should be separated by a semicolon.



Driver Installation

The following section gives information about the configuration of the drivers in Ozeki Messages Server. The primary task of drivers is to make sending and receiving of messages possible.

Available drivers for installation



GSM Modem

This driver makes it possible to connect a GSM phone to the PC with a phone-to-PC data cable, and to use it to send and receive SMS messages.



CIMD2 Connection

CIMD2 (Computer Interface to Message Distribution protocol) can be used to connect your PC directly to the SMSC of the GSM Service Provider. This way you can send and receive SMS messages over the Internet or a private IP network.



SMPP Connection

SMPP (Short Message Peer to Peer) protocol can be used to connect your PC directly to the SMSC of the GSM Service Provider. This way you can send and receive SMS messages over the Internet or a private IP network.



UCP Connection

UCP (Universal Computer Protocol) can be used to connect your PC directly to the SMSC of the GSM Service Provider. This way you can send and receive SMS messages over the Internet or a private IP network.



Colour SMS

The Colour SMS driver makes it possible to connect to the Premium Rated SMS service provided by T-Mobile Hungary. This driver accepts incoming SMS messages over HTTP and generates a reply message in the HTTP response.



Eurotel M2M

The Eurotel M2M driver makes it possible to connect to the Premium Rated SMS service provided by Eurotel (Czech Republic).



HTTP PRS

The HTTP Premium Rate Service (PRS) driver makes it possible to attach to premium rate SMS services. These services use an HTTP request for each incoming SMS message and they accept a reply message in the HTTP response.



Pannon GSM SSM

The Pannon GSM SSM driver makes it possible to connect to the Premium Rated SMS service provided by Pannon GSM Hungary. This driver accepts incoming SMS messages over HTTP and generates a reply message in the HTTP response.



Data call

This driver makes it possible to connect a modem to the PC and use it to send and receive data in a standard phone call.



Virtual phone

The virtual phone can be used by developers to test their applications without the need of sending real SMS messages. If the developer phone is installed, the Internet Explorer webbrowser can be used to connect of the SMS server to simulate SMS sending and receiving.

Installing the GSM Modem Driver

This text gives detailed information on the installation of the GSM Modem driver. The GSM modem driver can be used to send and receive SMS messages with a GSM phone attached to the PC. The installation of this driver can be started from the server manager by clicking on the **Installed drivers** menu item from the **Drivers** menu. In the "Available drivers for installation" window the **GSM Modem** [install](#) link should be clicked and the installation process will start. After then installation process has finished, you'll see the **GSM Modem 1** appearing in the list of drivers in the "Currently installed drivers" screen. On this page you can start the configuration by clicking on the [Configure](#) link (Figure 1).

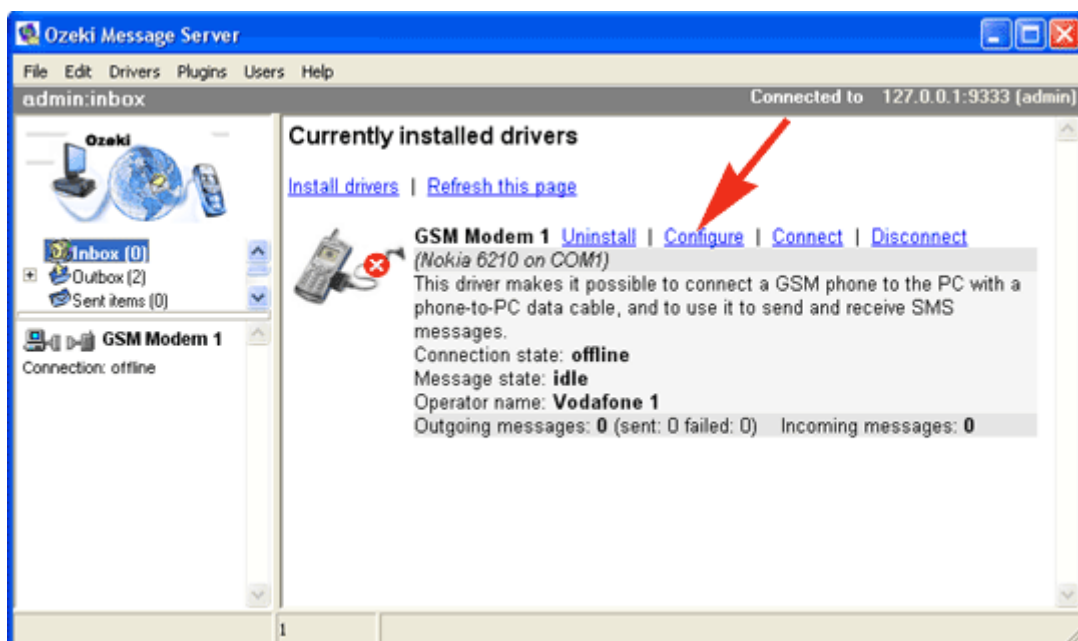


Figure 1.- Starting GSM driver configuration

The Configure link opens the configuration form. On this form select the phone model you have attached to the computer, then specify the serial port. If use use an USB port you must install a USB driver, that will create a virtual COM port for your device. If you use Infra Red (IR), you might have to install IROCom support for your Windows PC, to get a virtual COM port you can use to communicate with your device.

After selecting the serial (COM) port and the phone model you should click on the Autodetect button. During the autodetection phase, Ozeki Message Server will read the capabilities of your phone. If your phone supports SMS messages you will see the model and the brand displayed in the Phone model edit box.

The next step in the configuration is creating a unique name for your device. This name is called the operator name and it is used if more then one devices are attached to your PC. The operator name is typically the name of the GSM company your subscription belongs to. (For example it can be Vodafone, T-

mobile, Orange, etc.) When this is done, enter the telephone number of the phone attached to the computer and the SMS Service center address (Figure 2). **Important: phone numbers should be entered in international format e.g.: +36209937723 !**

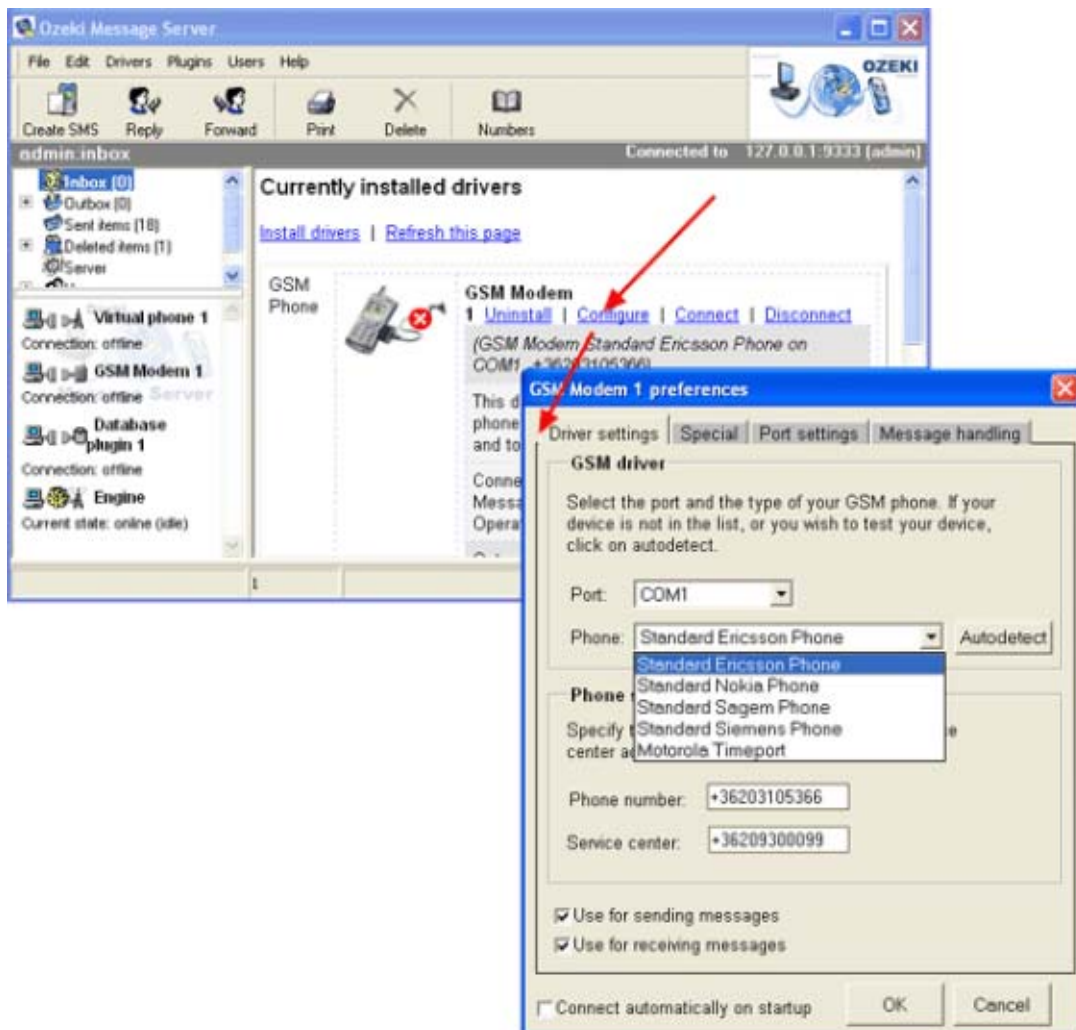


Figure 2 . - Configuring the details

The Service Center address is provided by your GSM service provider . Some SMSC addresses are listed in this document. If you don't know the Service Center address of the GSM service provider you should check [Appendix "C"](#).

For example, in Hungary you would select one of these numbers:

T-mobile	+36309888000
Pannon GSM	+36209300099
Vodafone	+36709996500

Starting the GSM modem connection

After the driver has been installed, you need to start it. The startup process means that the driver connects to the mobile phone. You can start the Ozeki Message Server GSM modem driver by clicking on "Connect" in the "Currently installed drivers" screen (Figure 3)

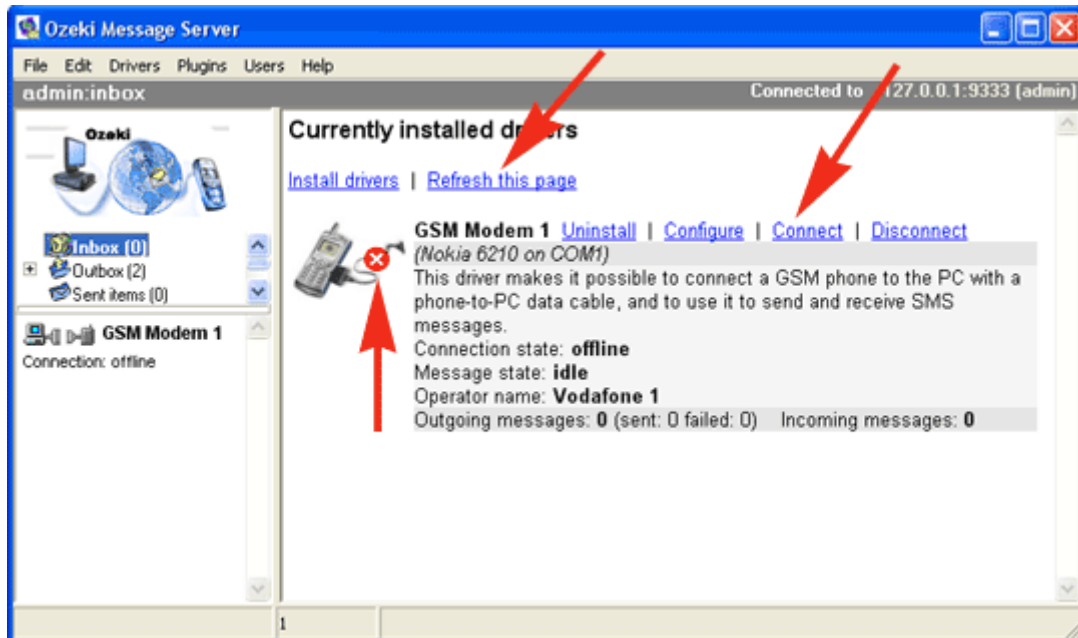


Figure 3 . Connection of GSM driver

After you have clicked on the [Connect](#) URL, you may use the [Refresh this page](#) link to check the connection state. Once the application is installed and configured properly, you can send an SMS by clicking on the Create SMS icon to verify your setup.

Configuring delivery reports

If you use a GSM modem, you can benefit from a strong feature of the GSM network. The GSM network can be asked to send a report SMS message to the sender party when an SMS message is delivered to the destination phone. This SMS message is called the delivery report SMS.

If you wish to receive delivery reports, you should place a checkmark into the "Request delivery report SMS" checkbox on the Message Handling tab of the GSM Modem Configuration form (Figure 1).

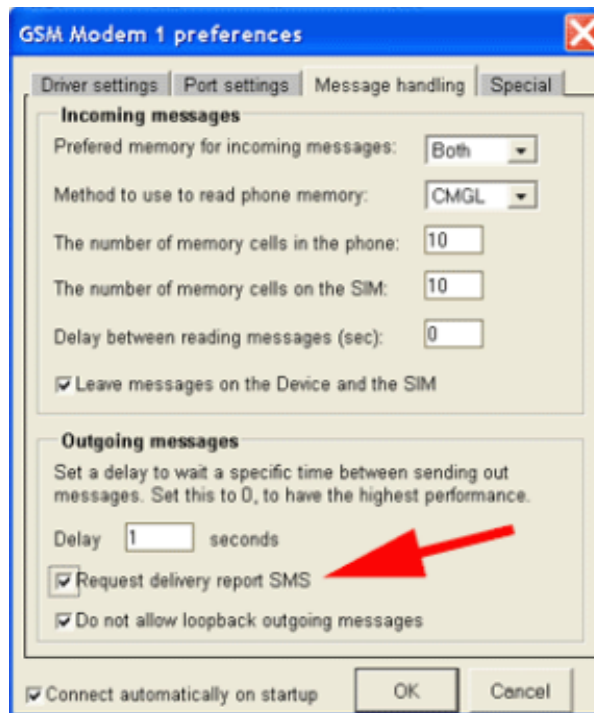


Figure 1 - GSM Modem Configuration

How to Install more then one device

Ozeki Message Server can handle several phones simultaneously. This is very useful if you wish to increase the capacity or to create least cost routing. If several devices are attached, the server automatically performs load balancing. To get multiple devices support, you need to understand some basic concepts: In Ozeki Message Server every device is handled by an instance of a device driver. If you use more then one device you have to install more then one instance of the same driver. As you already know, the driver installation can be initiated by clicking on the **Installed drivers** menu item in the Drivers menu.

Installing the first device

To install the first device you have to select the GSM Modem (or the appropriate IP SMS) driver and click on the **install** link. After the driver is installed the "Currently installed drivers" page will appear in the server manager. On this page, locate your newly installed driver and click on the **configure** link to perform the configuration (Figure 1). To have multiple driver support, on the configuration form, you must assign a unique **operator name** to the installed device. Operator names are used to identify the device. You can specify the operator name by selecting the **Special** tab, and filling in the Operator editbox. Change the text to something unique, for example "Vodafone1" (Figure 1).

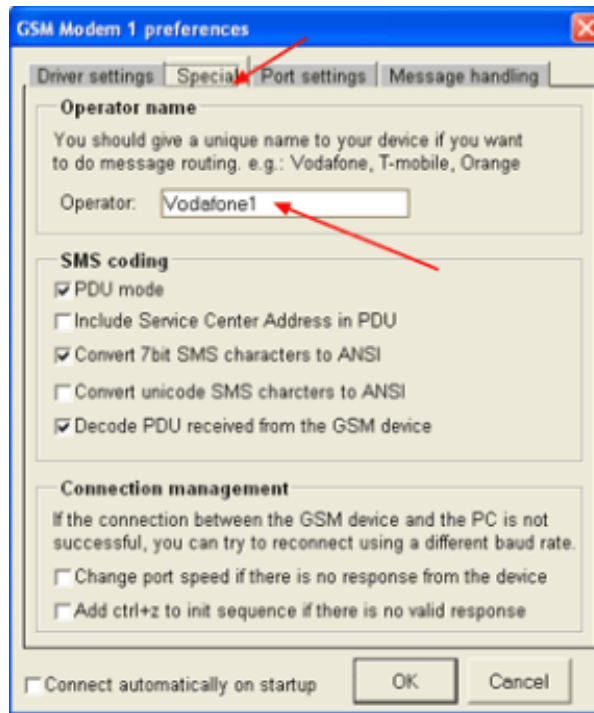


Figure 1 - Settings on the **Special** tab

Click OK to finish the configuration.

Installing the second device

To install a second device, click on the **Install drivers** link in the "Currently installed drivers" page. On this page select the **GSM Modem** driver **again**. This will install a second instance of the driver allowing you to configure the second device(Figure 2).

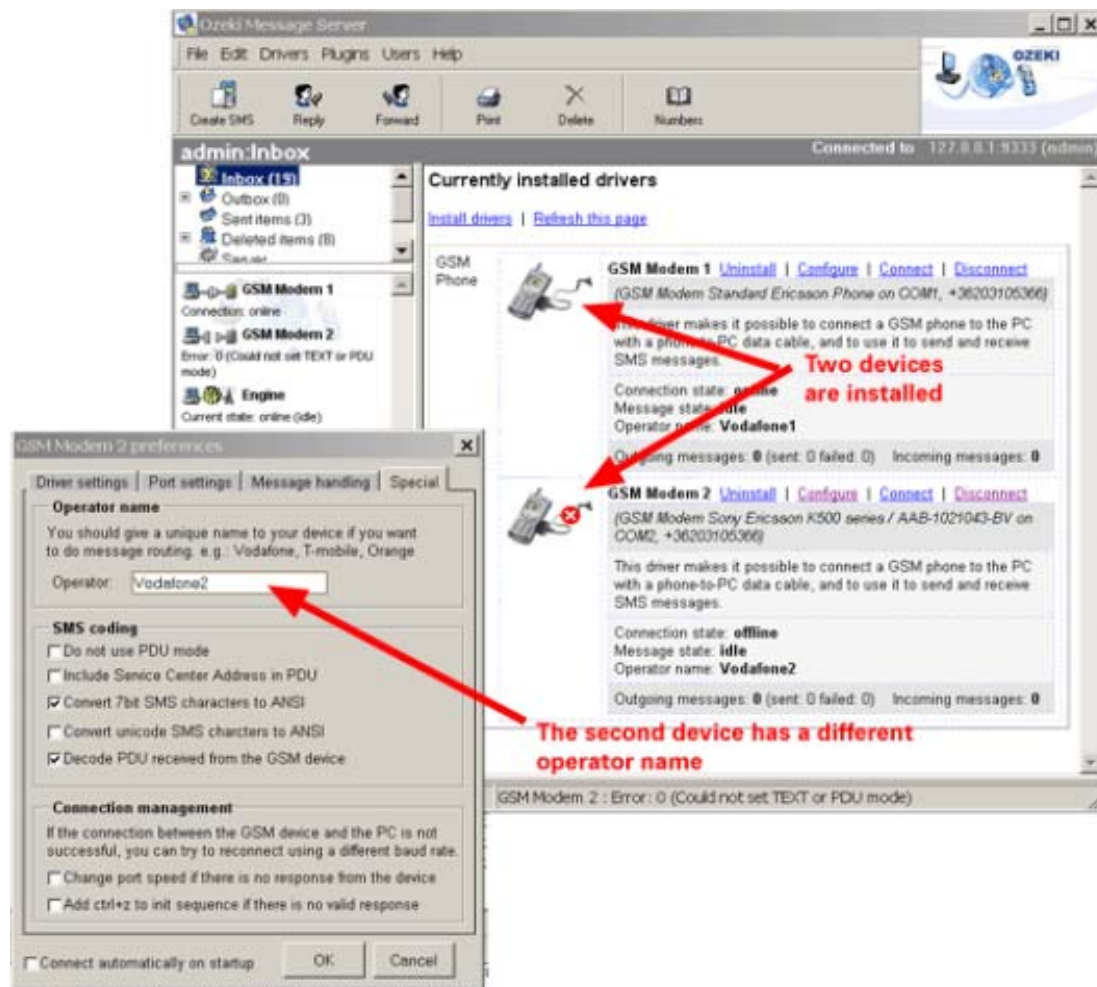


Figure 2 - The second device

On the configuration form of the second device, you must assign an **operator name** that is different from the operator name you have assigned to the first device . For example specify "Vodafone2".

Logging the port communication

If you would like to log the communication between your GSM phone/modem and your computer, you should turn on the "Log Port Communication" feature on the Port settings tab of the GSM Modem 1 preferences form (Figure 1).

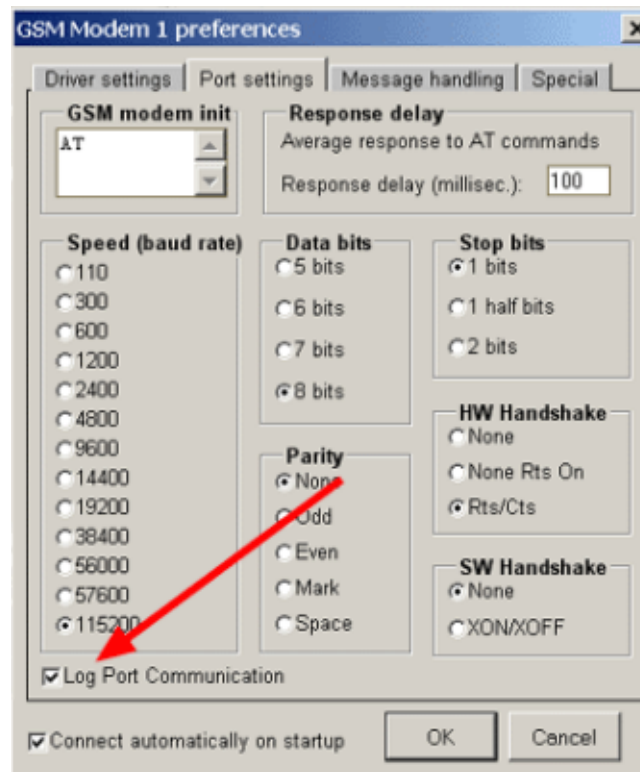


Figure 1 - GSM Modem preferences form

After this feature is activated, a log file will be created in the C:\Program Files\Ozeki\Ozeki Message Server\Logs directory. The name of the log file is "GSM Modem1.txt"

This is the log file that holds the contents of the communication (Figure 2).

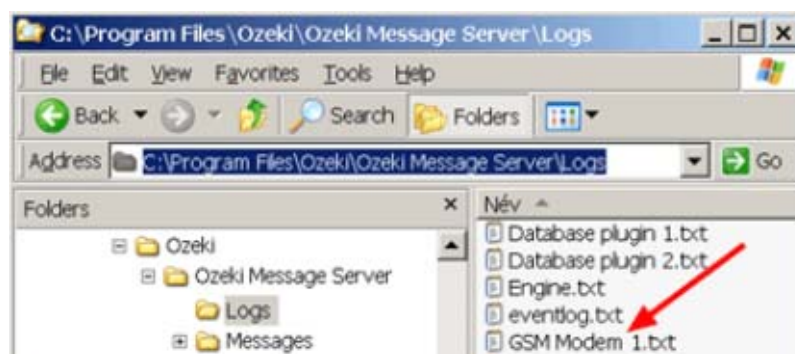


Figure 2 - Log file

Using hyper terminal to test the GSM Modem

In some situations Ozeki Message Server will not detect your GSM phone. In this case you can test whether you phone responds using Hyper Terminal. Hyper Terminal is built into windows. It is available in the Start menu. To start it, selecting Start->All programs->Accessories->Communications->HyperTerminal. When Hyperterminal start, you must provide a name for the

connection. Type any name you like to get to the "Connect To" configuration form (Figure 1). In the "Connect To" configuration form select the COM port your device is attached to. If you use a USB cable, a PCMCIA modem, a BlueTooth connection or an Infrared port, you should select the virtual port created by the driver of your connection here .

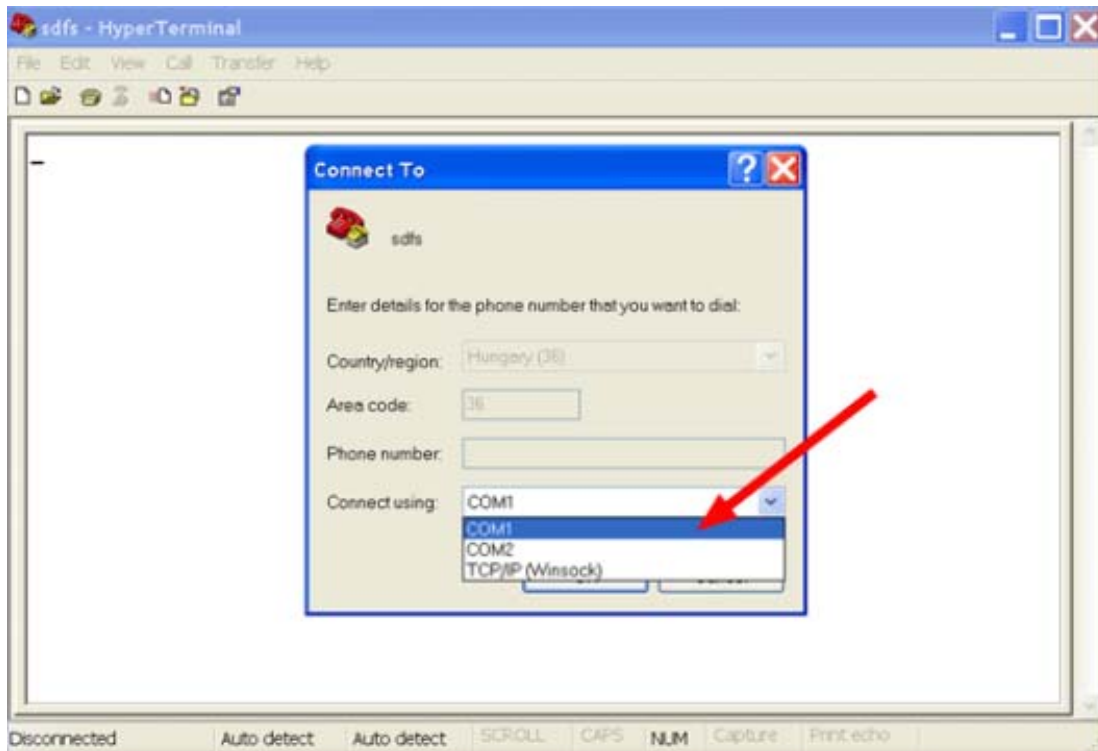


Figure 1 - select the COM port

When the port is selected hit Ok, to get to the Port configuration form. On this form specify the port settings. The most common port speeds for GSM phones and modems are 9600, 19200, 57600 and 115200. The Data bits should be set to 8. The Parity to none the Stop bits to 1 and the flow control should be hardware. Finally the terminal window will open. You can use this window to test the communication.

The communication can be tested by typing AT into the terminal window. If your GSM phone or modem is working the AT will appear and an OK response will be returned. If your GSM phone or modem is not working then you will not see the AT command after typing it in (Figure 1).

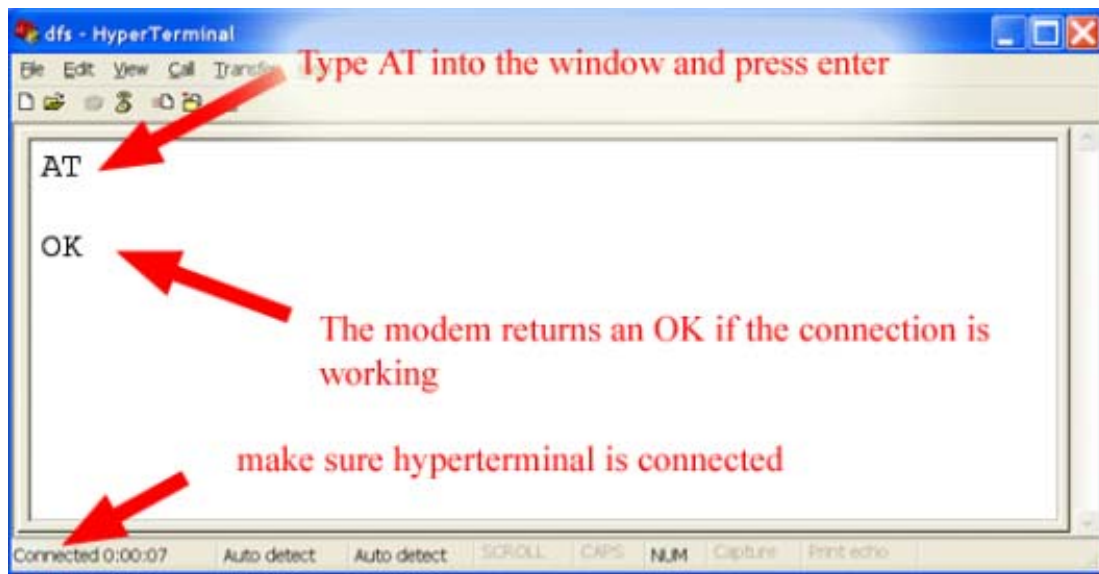


Figure 2 - testing the connection

If the test was successful, which means you have seen the OK response from your GSM phone or modem please close HyperTerminal and try the "Autodetect" option in the GSM Modem 1 Configuration form in Ozeki Message Server 6.

Installing the CIMD2 driver

The CIMD2 protocol is used for IP SMS connectivity. The installation of the CIMD2 driver is similar to the installation of the GSM modem driver. In this chapter we give the instructions on how to configure this connection. Before you start the installation, make sure you have the necessary login information from your IP SMS provider to connect to their service. You will need the IP address of their server, the Port, a username and a password plus a telephone number that was assigned to the connection.

The CIMD2 configuration preferences form (Figure 1.) can be accessed in a similar way to the GSM Modem preferences form. The difference, is that this form contains different configuration options.

To configure a CIMD2 connection you should provide the necessary login information to connect to the CIMD2 server . This information is provided by the SMS service provider, that offers IP SMS connectivity. The connection parameters are the host name (sometimes only an IP address is given), the port, a user name and a password. Often service providers use the phone number of the service or the port number as the user name.

The operator name can be any unique text, that will be used if more then one connection is installed in Ozeki Message Server. This is a name for the connection. Most often we give the name of the service provider, this driver connects to, to identify the connection.

The phone number refers to the telephone number assigned to the application. This phone number in many cases can be freely chosen when sending outgoing messages. You can even specify a text.eg.: OZEKI in this field.

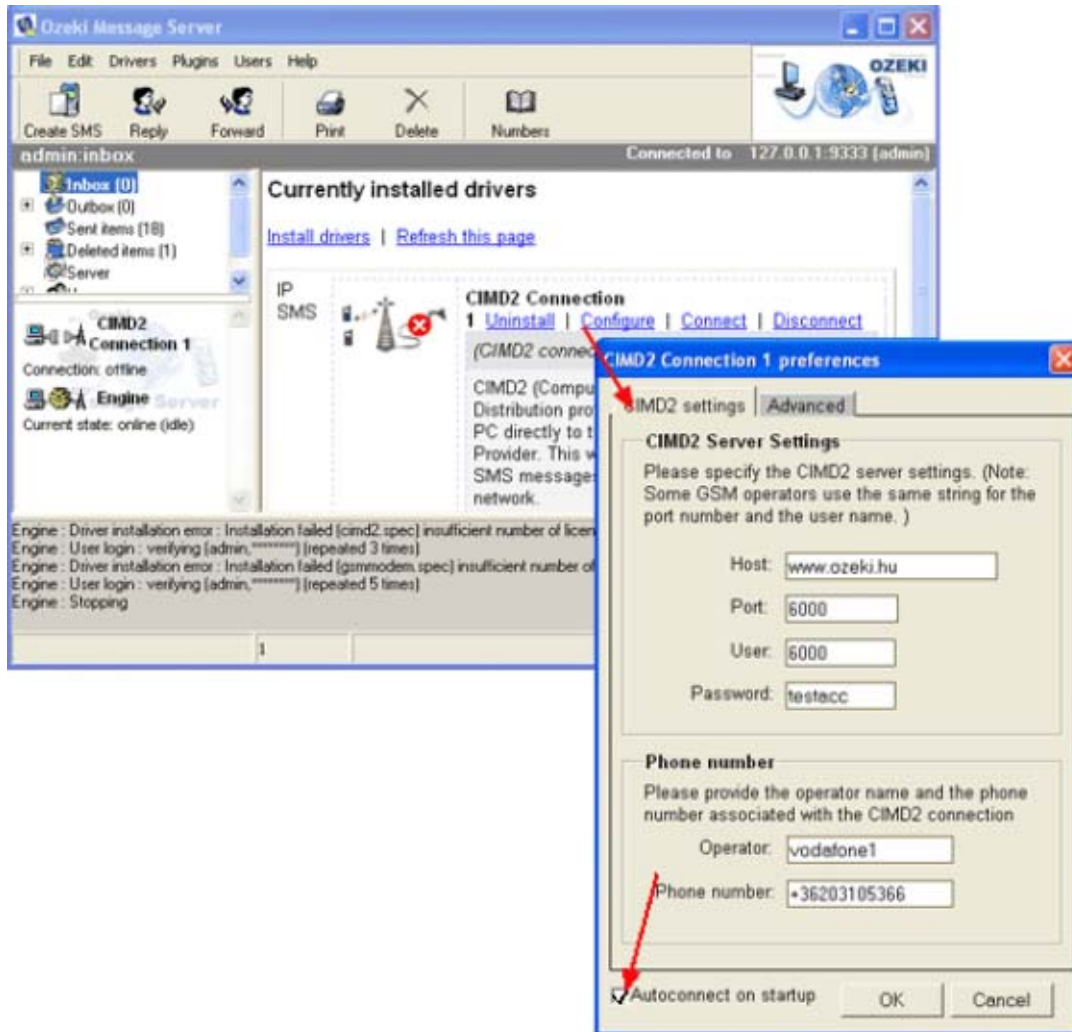


Figure 1. - CIMD2 settings

In the Advanced tab of the CIMD2 Connection 1 preferences form, we have to leave the editbox of the source IP and Source port blank, unless the operator inquires us to specify it. We have a choice to set the time period for **Keepalive** requests (Figure 2.). The checkboxes below the keepalive field should be left untouched in most cases.

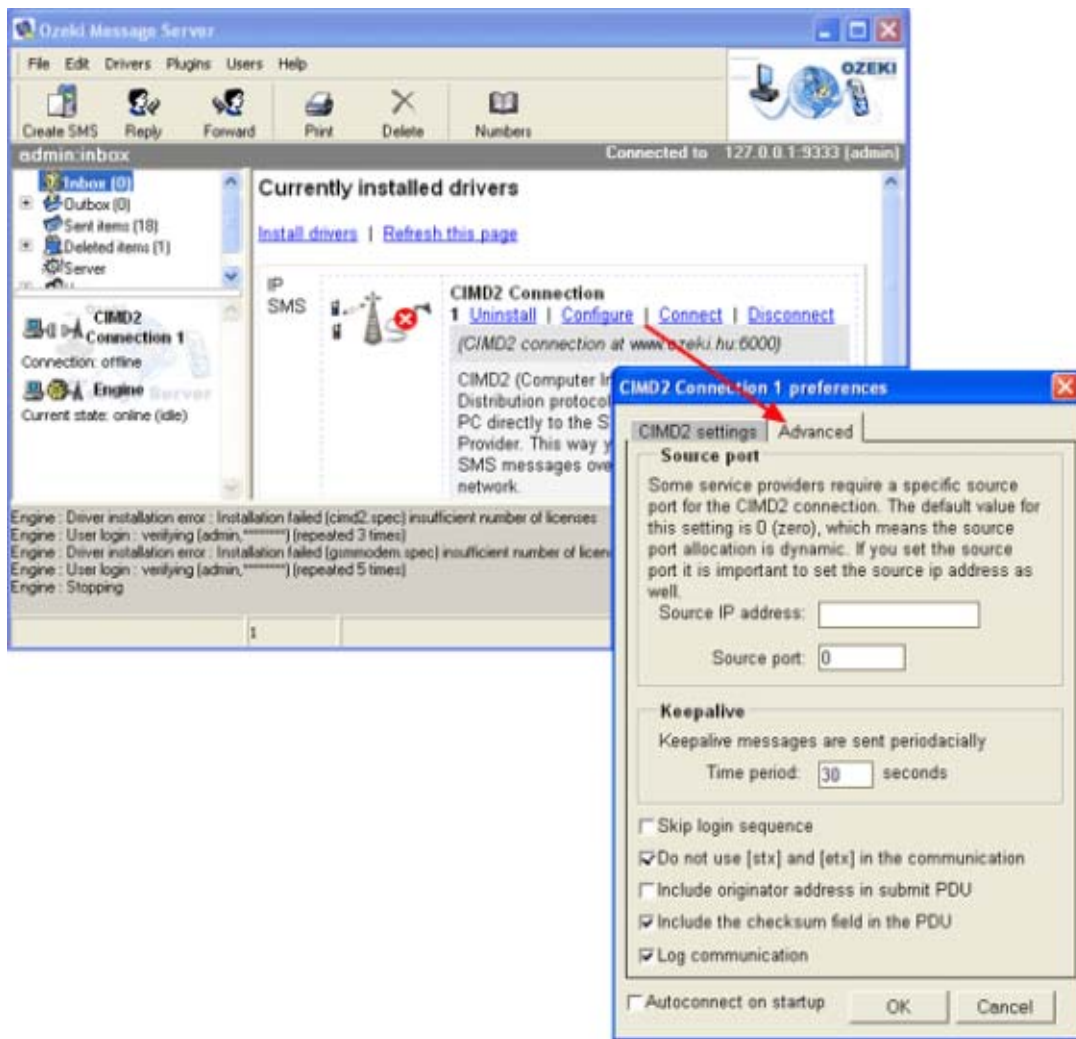


Figure 2. - Advanced

Configuring dynamic phone number support

Every IP SMS protocol (CIMD2, SMPP, UCP) allows you to specify a custom sender telephone number. You can use this option to specify the sender address individually for each outgoing SMS message.

To achieve this you must replace the sender phone number to a special keyword called `_DYNAMIC_`. The sender phone number can be configured on the configuration form of the IP SMS driver. Figure 1 shows it on the UCP configuration form.

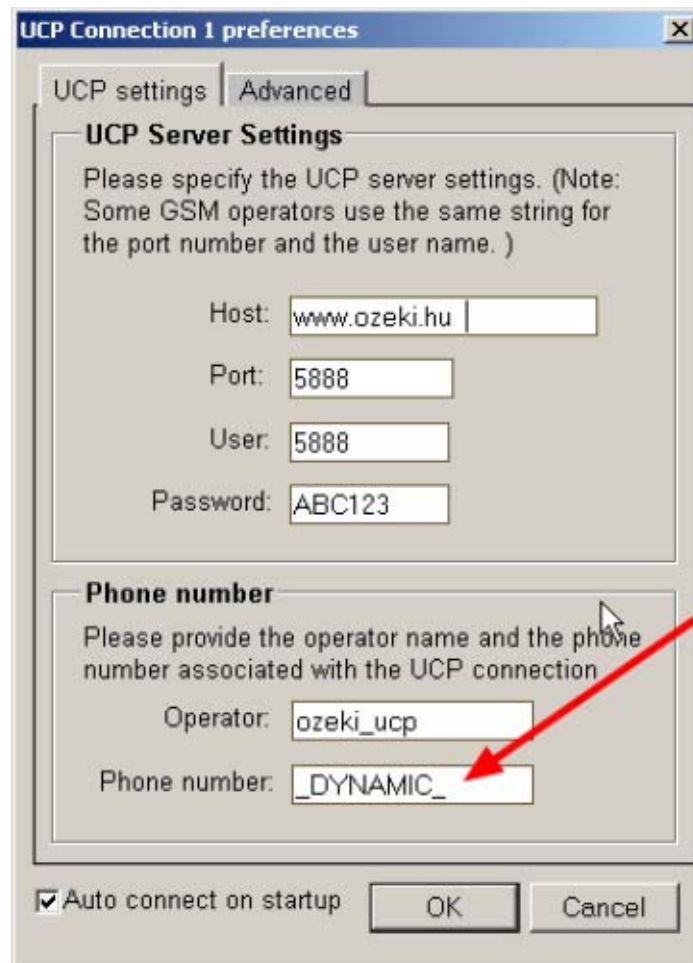


Figure 1 - Configuring dynamic sender phone number support

If you are using the SQL plugin, and you have configured dynamic phone number support in the UCP driver, make sure you always specify the sender address in your SQL table.

For example you can use the following query:

```
insert into ozekimessageout (sender,receiver,msg,status) values ('+36205222245','+36303596242','dynamic sender test','send');
```

Note: this feature currently only works from plugins, because you cannot access the sender field in the Ozeki GUI.

Installing the SMPP driver

The SMPP protocol is used for IP SMS connectivity. The installation of the SMPP driver is similar to the installation of the GSM modem driver. In this chapter we give the instructions on how to configure this connection. Before you start the installation, make sure you have the necessary login information from your IP SMS provider to connect to their service.

You will need the IP address of their server, the Port, a username and a password plus a telephone number that was assigned to the connection.

The SMPP configuration preferences form (Figure 1.) can be accessed in a similar way to the GSM Modem preferences form. The difference, is that this form contains different configuration options.

To configure a SMPP connection you should provide the necessary login information to connect to the SMPP server. This information is provided by the SMS service provider, that offers IP SMS connectivity. The connection parameters are the host name (sometimes only an IP address is given), the port, a user name and a password. Often service providers use the phone number of the service or the port number as the user name.

The operator name can be any unique text, that will be used if more then one connection is installed in Ozeki Message Server. This is a name for the connection. Most often we give the name of the service provider, this driver connects to, to identify the connection.

The phone number refers to the telephone number assigned to the application. This phone number in many cases can be freely chosen when sending outgoing messages. You can even specify a text.eg.: OZEKI in this field.

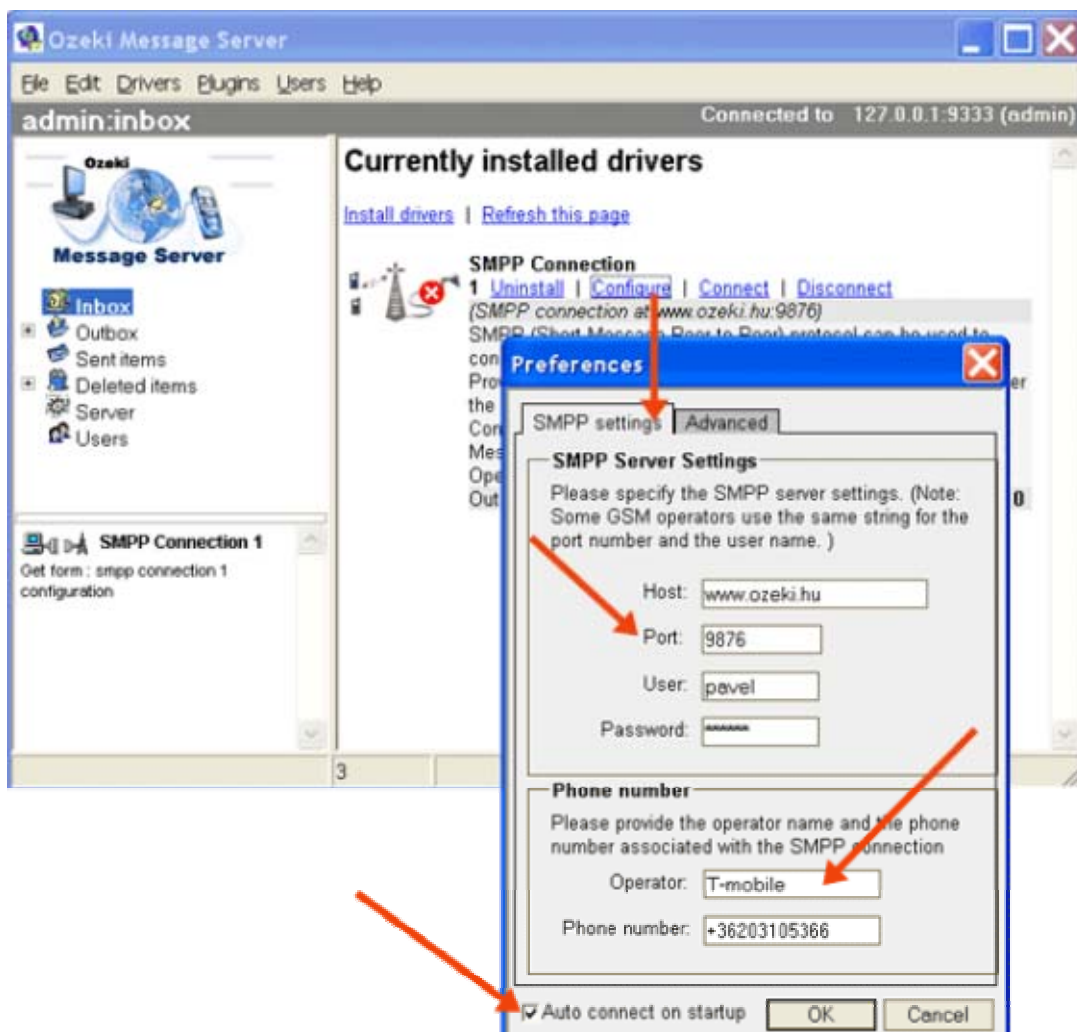


Figure 1 . - SMPP settings

You can also put a special word in the phone number field: `_DYNAMIC_`. This means, that when a message is being sent out, for example from the database plugin, the sender phone number is changed to the sender phone number stored in the message. *Note: this feature currently only works from plugins, because you cannot access the sender field in the Ozeki GUI.*

In the Advanced tab of the SMPP Connection 1 preferences form, we have to leave the editbox of the source IP and Source port blank, unless the operator inquires us to specify it. (Figure 2.)

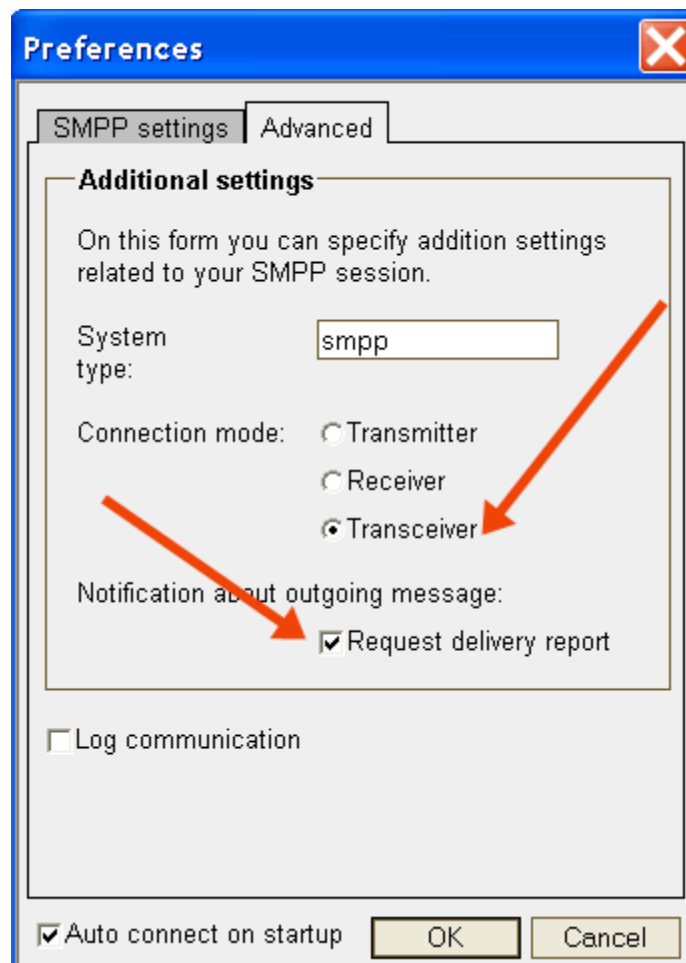


Figure 2. - SMPP advanced

How to turn on the SMPP communication logging

If there is a problem in the communication between the SMPP server and the Ozeki Software, the best way to resolve the problem is to check the SMPP communication logs. In this section we give guidelines on how you can turn on the logging.

The first step is to click on the Installed drivers menu item in the drivers menu. On the driver listing page, you need to click on the SMPP driver configuration link to open the SMPP driver configuration form. The SMPP preferences form has two tabs, the SMPP settings tab and the Advanced tab. On the advanced tab, place a checkmark in the Log communication checkbox (Figure 1).

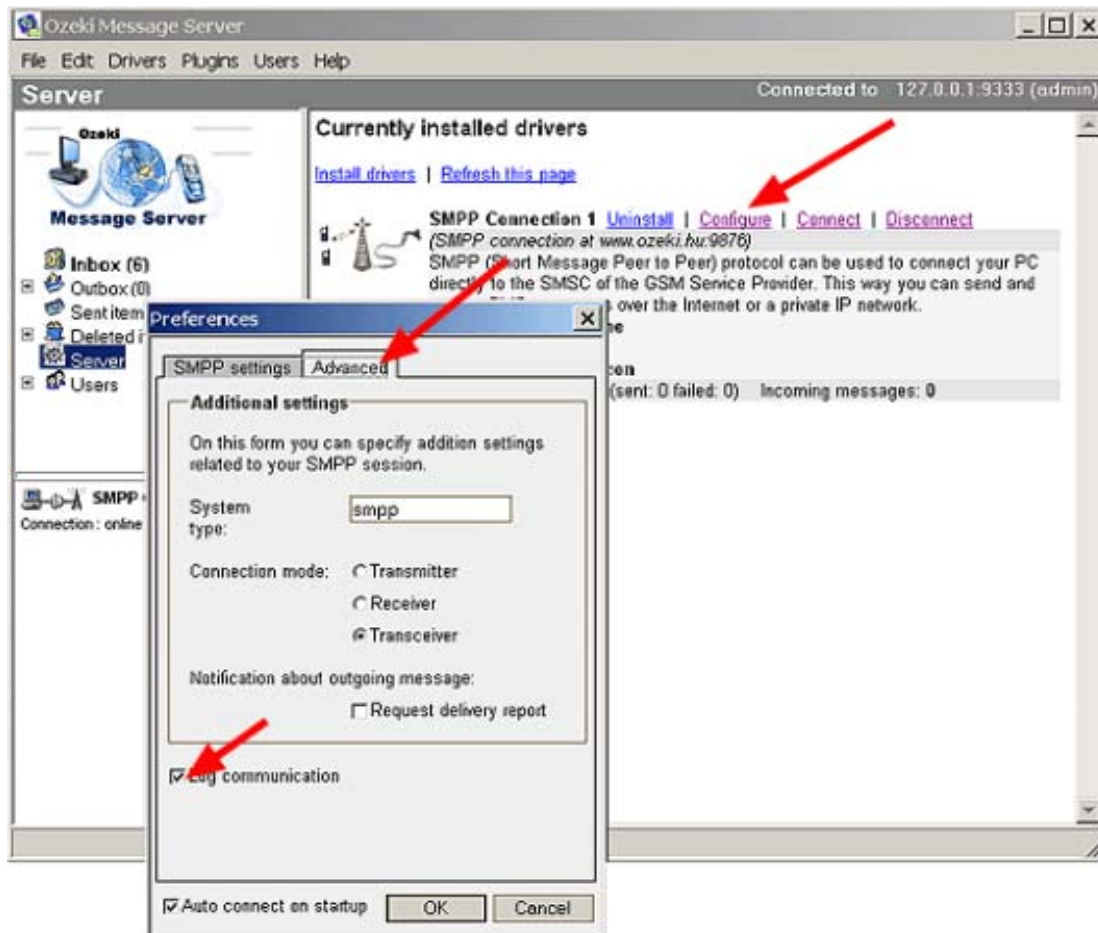


Figure 1 - SMPP configuration

Following this, configure the server to accept communication log messages from the SMPP driver. This can be done by opening the Server Preferences Form. On this form you can click on the Event logs tab, where you should check the Driver communication checkbox .

After the logging has been configured, and the actions that resulted in log events have been taken, please send us the log file. The SMPP log file can be found in the following directory: C:\Program Files\Ozeki\Ozeki Message Server\Logs (Figure 2).



Figure 2 - Logs

Installing the UCP driver

The UCP protocol is used for IP SMS connectivity. The installation of the UCP driver is similar to the installation of the GSM modem driver. In this chapter we give the instructions on how to configure this connection. Before you start the installation, make sure you have the necessary login information from your IP SMS provider to connect to their service. You will need the IP address of their server, the Port, a username and a password plus a telephone number that was assigned to the connection.

The UCP configuration preferences form (Figure 1.) can be accessed in a similar way to the GSM Modem preferences form. The difference, is that this form contains different configuration options.

To configure a UCP connection you should provide the necessary login information to connect to the UCP server. This information is provided by the SMS service provider, that offers IP SMS connectivity. The connection parameters are the host name (sometimes only an IP address is given), the port, a user name and a password. Often service providers use the phone number of the service or the port number as the user name.

The operator name can be any unique text, that will be used if more than one connection is installed in Ozeki Message Server. This is a name for the connection. Most often we give the name of the service provider, this driver connects to, to identify the connection.

The phone number refers to the telephone number assigned to the application. This phone number in many cases can be freely chosen when sending outgoing messages. You can even specify a text.eg.: OZEKI in this field.

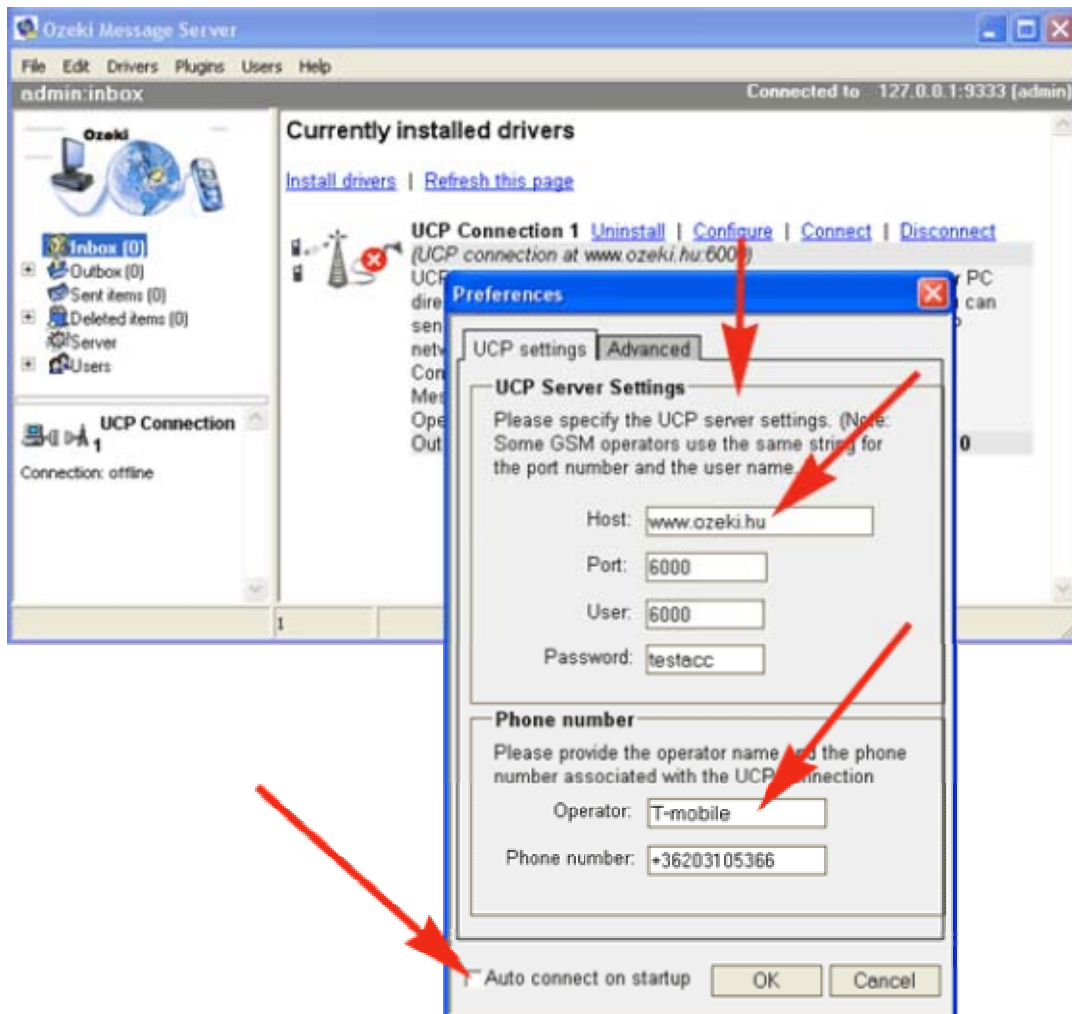


Figure 1. - Configuration of Driver details

In the Advanced tab of the UCP Connection 1 preferences form, we have to leave the editbox of the source IP and Source port blank, unless the operator inquires us to specify it.

Installing the virtual phone driver

The virtual phone is a tool developers can use to build applications. It simulates the presence of a GSM network (Figure 1). It can be used to imitate sending and receiving messages with Ozeki Message Server. This driver was introduced, because it is hard to develop and test applications if you have to send/receive SMS messages from a GSM handset and costs can be also reduced this way.

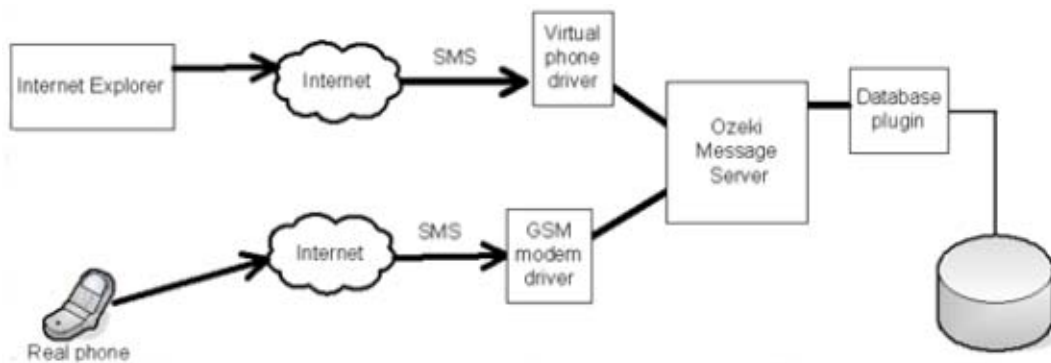


Figure 1. - Virtual phone for developers

The Virtual phone driver can be installed like any other driver in the Server Manager application. It provides a user interface accessible through a web browser like Internet Explorer. The user interface becomes available after the driver is connected by opening the following URL: <http://127.0.0.1:9335>. The 127.0.0.1 IP address should be replaced by the server IP address and the 9335 port should match the port specified in the virtual phone driver preferences form (Figure 2). Note, that this port number should be different from the default port (9333), Ozeki Message Server is working on. 9335 is a good option.

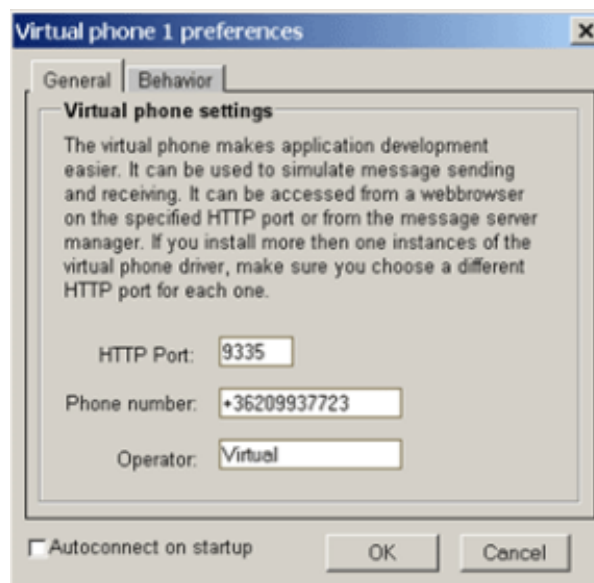


Figure 2 . - Virtual phone settings

To provide good simulation, the virtual phone driver makes it possible to set the average delay period between sending out messages. If you place a checkmark in the "**Simulate message transmission failures**" checkbox, it means that sending errors will be simulated for outgoing messages. Incoming messages can be simulated by opening the Drivers/ Virtual phone 1 / Testphone menu item. This option opens a form where the originator phone number and the message text can be entered (Figure 3).

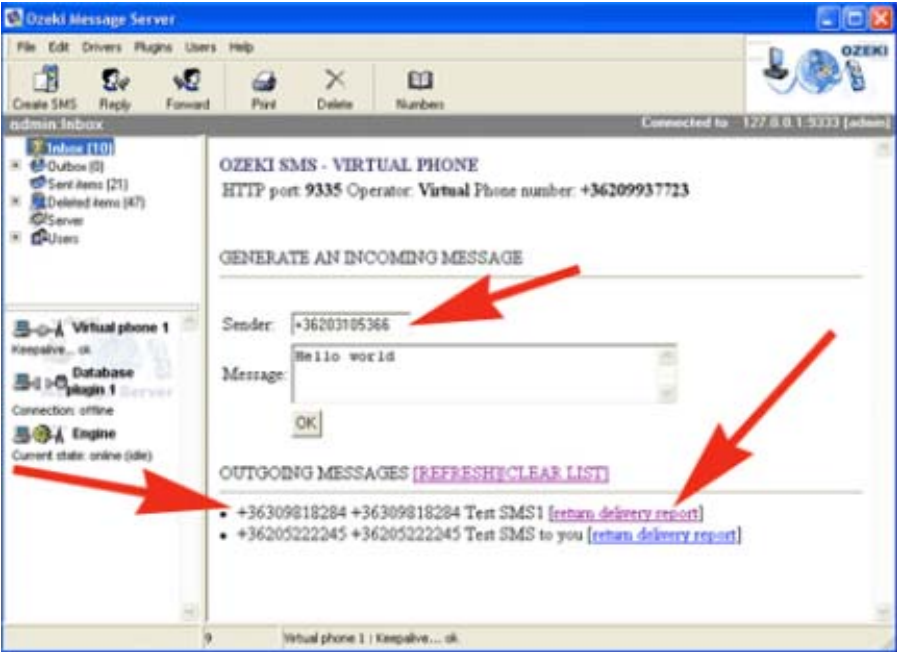













Figure 3 . - Delivery report

Plugins

Plugins make it possible to attach external applications to Ozeki Message Server. Each plugin offers an interface, such as a database server that can be used to pass outgoing messages to the message engine. Of course these interfaces also make receiving of messages possible. Currently Ozeki Message Server contains the following plugins:

	<p>Colour SMS plugin</p> <p>The colour SMS protocol was introduced by the Hungarian GSM service provider T-Mobile to pass incoming SMS messages to 3rd party content providers in HTTP requests. The applications operated by the content providers return a response SMS message in each HTTP request. This plugin implements this functionality.</p>
	<p>Database plugin</p> <p>With the help of this plugin you can use a database server to send and receive SMS messages. The plugin connects to the database and checks a database table periodically for outgoing messages. If an incoming message arrives it inserts it into another table.</p>
	<p>File plugin</p> <p>The FILE PLUGIN plugin makes it easy to send and receive messages using text files. When an incoming message arrives the plugin creates a text file for the message. If you would like to send a message, all you have to do is place a text file into the outgoing directory.</p>
	<p>HTTP Client plugin</p> <p>The HTTP CLIENT plugin takes incoming messages and passes them in an HTTP GET or an HTTP POST request to an external webserver. The HTML page returned by the webserver can contain SMS messages, that will be sent by the server.</p>
	<p>PHP plugin</p> <p>The PHP plugin makes it easy to build a script that can process incoming messages and create response messages. In this plugin you have all the functionality PHP offers.</p>
	<p>POP3 plugin</p> <p>With the help of this plugin you can use a standard e-mail client, such as MS Outlook Express to download your incoming SMS messages.</p>
	<p>SMTP plugin</p> <p>With the help of this plugin you can send SMS messages from a standard e-mail client, such as MS Outlook Express. This plugin is often installed along with the POP3 plugin which makes receiving of SMS messages possible.</p>
	<p>TCP plugin</p> <p>With the help of this plugin you can connect TCP/IP clients to the Message Server to send and receive SMS messages.</p>
	<p>Appstarter plugin</p> <p>The appstarter plugin can start applications and run scripts on your computer when an incoming message arrives.</p>
	<p>Autoreply plugin</p> <p>The autoreply plugin makes it possible to reply to incoming messages and to forward these messages to various phone numbers based on the sender</p>

	address and the message content.
	Tester plugin This plugin generates random outgoing SMS messages. Good for developing and testing services.

System maintenance

In this section some information is provided to help system administrators keep the system healthy and achieve optimal performance.

Keep the system healthy by deleting old messages

The Ozeki Message server software works with message queues. The software provides better performance if the number of messages in the queues are smaller. It is advised to keep the number of message in a queue below 5 000. To reach this we suggest turning on the automatic message deletion feature. This can be done, by opening the Server Preferences for in the Edit menu. On the Server preferences form, you can select a time period, such as 1 hour, to tell the server, that messages older then this period should be deleted. Note, that if you have plugins, such as the database plugin installed, the messages will not be deleted from the database. They will only be deleted from the internal queues of the Ozeki Message Server Applications.

Ozeki Performance Monitor

The Ozeki Performance Monitor is an optional tool for Ozeki Message Server. It's main job is to monitor the activity of the Message Server and to take action if something goes wrong. This performance monitor service periodically downloads the login page of the message server through HTTP. If the message server does not respond for any reason for 4 consecutive times, it terminates the appropriate server process and restarts the message server service.

You can download and install this tool from the following link: [OzekiPerformanceMonitor.exe](#)

Manual uninstallation


To uninstall Ozeki Message Server 6 from your computer manually, you have to do two things: delete the files and delete the registry keys from your computer.

Deleting the files

You must delete the following directory: C:\Program Files\Ozeki\Ozeki Message Server

Deleting the registry keys

To You can delete the registry keys using the Registry Editor (Regedit.exe).
Warning: If you use Registry Editor incorrectly, you can cause serious problems that may require you to reinstall your operating system!

To delete the registry kes, click on the Windows Start button , select run and type **regedit** into the input field. (Figure 1)

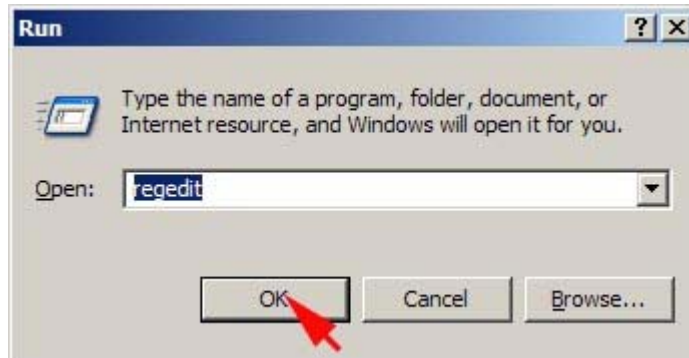


Figure 1 - Running regedit

In the registry editor navigate to the **HKEY_LOCAL_MACHINE / SOFTWARE / OZEKI** key . Hit the delete keyboard key and the **OzekiMessageServer** entry will be removed (Figure 2)!

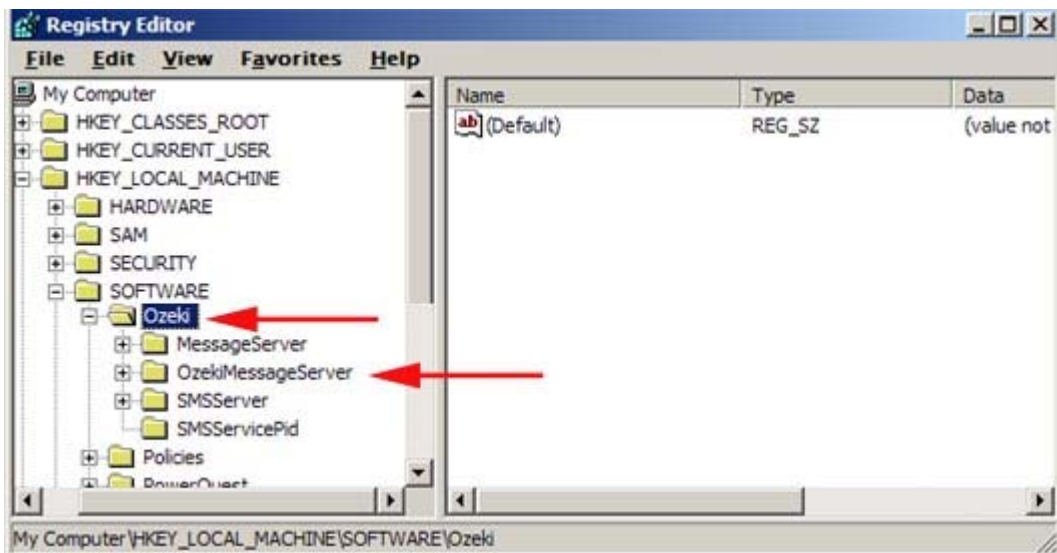


Figure 2 - Registry editor

Developers guide

Ozeki Message Server makes it possible to send and receive SMS messages. One of the strongest features of this Server software, is that it makes it possible to pass the incoming messages to other applications. These applications can send reply messages or they can send outgoing message any other time.

The most common question we receive, is how can we pass the incoming messages to the outside application and how do we deal with outgoing messages. The answer to this question is that we have developed several plugins, that act as a gateway between the application and Ozeki Message Server. Please take a look at some of the available plugins:



Most populpar plugins used by software developers:

<input type="checkbox"/> SQL
<input type="checkbox"/> Delphi
<input type="checkbox"/> Files
<input type="checkbox"/> Http client
<input type="checkbox"/> JAVA
<input type="checkbox"/> PHP
<input type="checkbox"/> ASP
<input type="checkbox"/> Access

General information

This section contains some information that every SMS developer should consider when creating automated SMS services.

Autoresponding SMS applications

If you plan to create an autoresponding SMS service, you must give a thought to the following scenario. Autoresponding SMS services threaten each other, and can cause serious problems for the operators and the service providers. If a user sends an SMS message from an SMS service, that responds to incoming messages automatically, to another service that behaves the same way, the two services can get in an endless loop causing infinite costs for the service providers.

Figure 1 illustrates this issue in more detail. As you can see on the figure the process is initiated by a human user sending an SMS message from a service called "Other autoresponding SMS application". The SMS arrives to Ozeki Messages Server, that responds automatically. The response generated by Ozeki

Message Server is then received by the "Other autoresponding SMS application" that may not interpret it and may generate a response SMS, creating an endless loop.

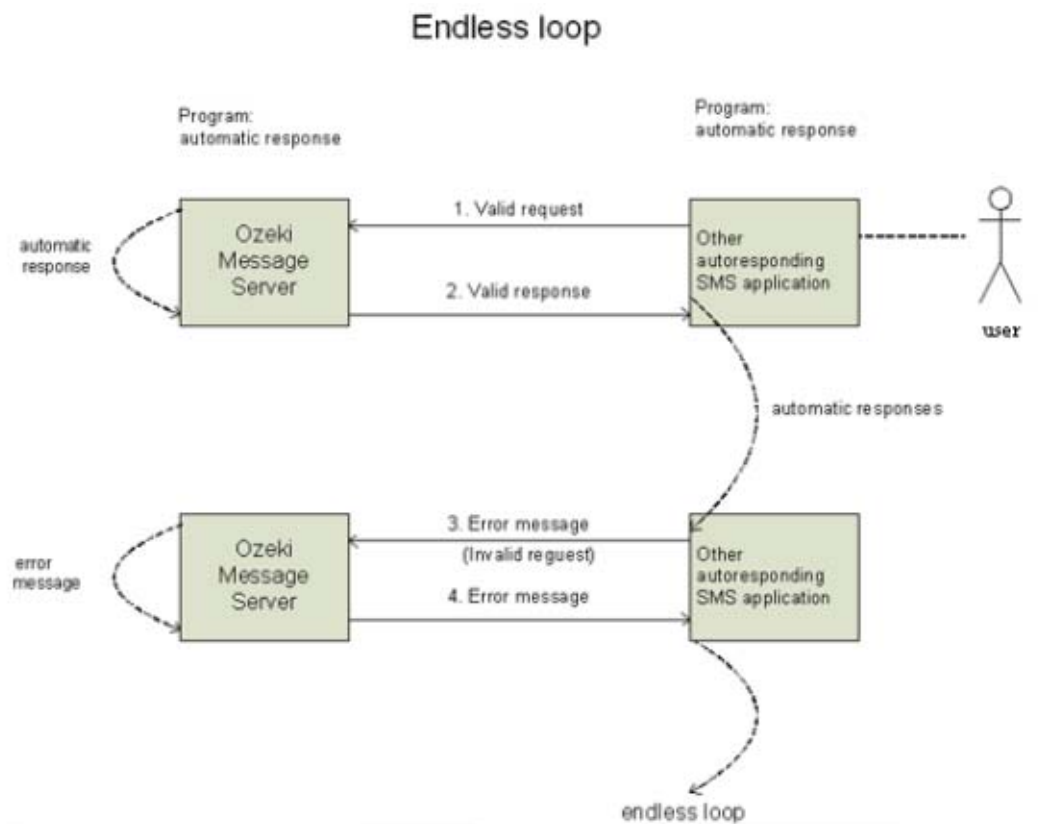


Figure 1 . - Two SMS applications in an endless leep

Application Mode

Developers creating SMS services can increase their efficiency in some situations if they run a software in application mode instead of running it as a system service. This can be very useful if problems need to be tracked down. Service applications cannot print messages on the screen and cannot be terminated and started as easily as a standard application. Ozeki message server 6 gives you an option to run the software in application mode.

If you want to start the Message Server Software in application mode, please start a Windows command prompt, by using Start/Run and entering cmd (Figure 1).

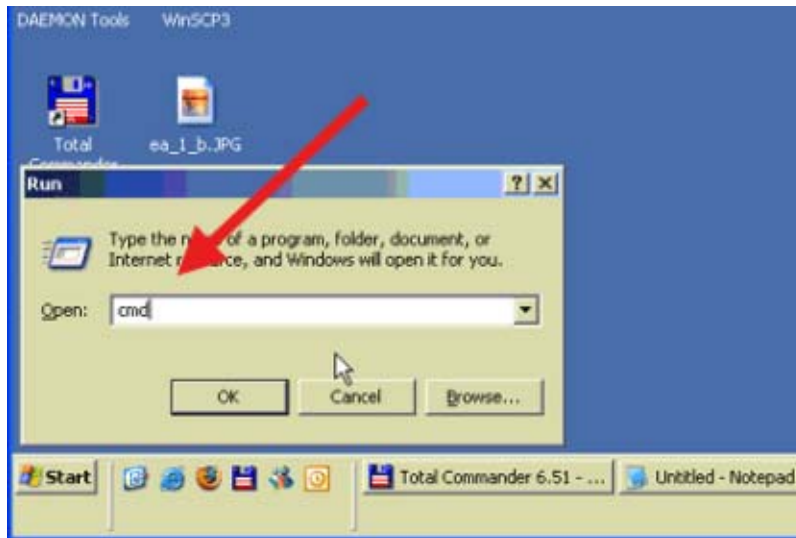


Figure 2 - Starting the command prompt

To start the Ozeki Message Server Software in application mode, you must switch to the Ozeki Message Server 6 directory using the following command:

```
cd C:\Program Files\Ozeki\Ozeki Message Server\
```

and you must issue the following command:

```
MessageServer.exe /run
```

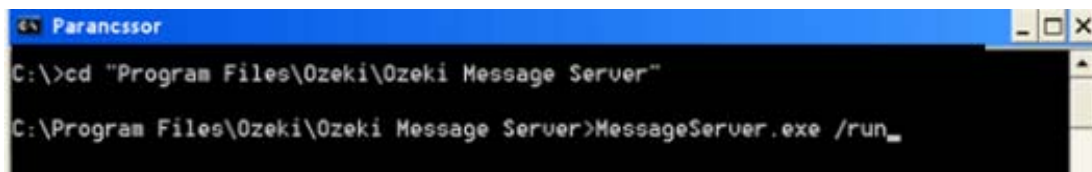


Figure 3 - Starting the server in application mode

After the server has been started in application mode, you can will see the server window. Please make sure, you do not close this window, before you start the Server Manager. If it is closed it means the server is not runing and you will not be able to log into it:

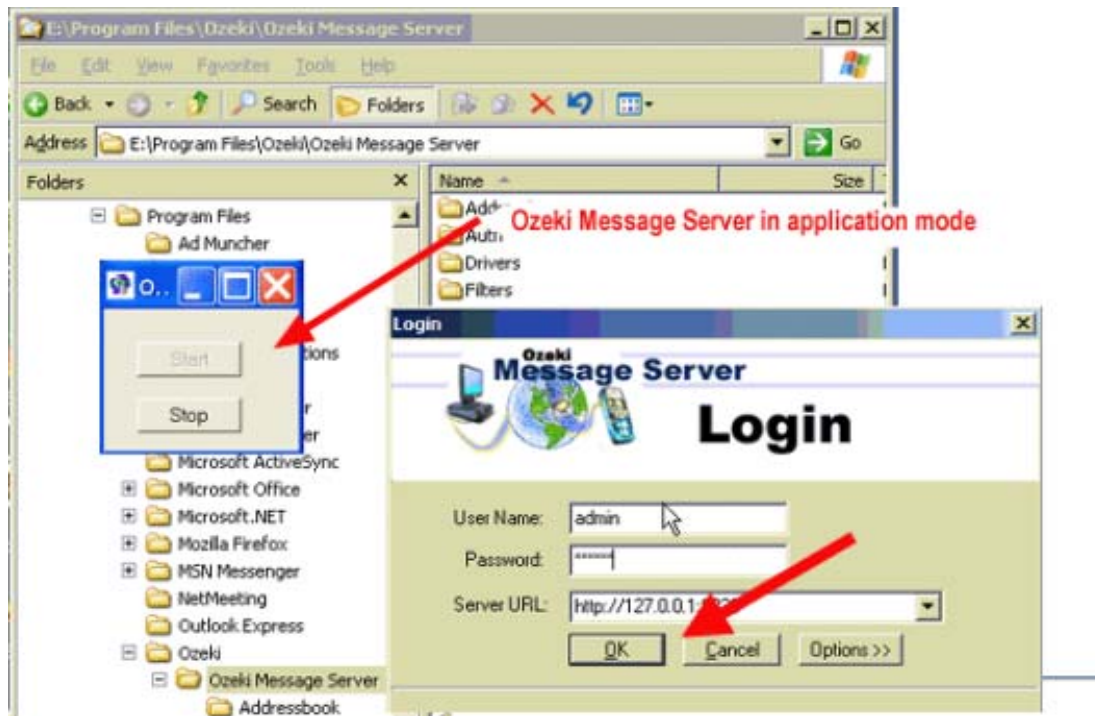


Figure 4 - log into the server.

Programming APIs and languages

This section contains example code and all the necessary information to use the services Ozeki Message Server 6 offers from various programming languages. You can find examples in C++, SQL, HTTP, Java, PHP, ASP, Delphi, Access and you will find some information if you want to simplify SMS messaging to creating and reading text files in your file system.

SMS API for C++ Developers

C++ developers can use a COM dll to connect to Ozeki Message Server 6. This DLL implements the communication between the C++ client application and the Ozeki Message Server. The DLL can be freely distributed along with your application. On this page we give the steps on how to use the C++ API.

STEP 1. - Download the C++ API with source

The file can be downloaded from the following URL:
http://www.ozeki.hu/index.phtml?ow_page_number=503 The name of the file is: cppClient.zip

STEP 2. - Unzip cppClient.zip

When you unzip the file you will see the source code in the cppClient directory. The cppClient directory also contains a compiled .exe version. The name of it this file is cppClient.exe

STEP 3. - Register the COM dll in your system

Open a windows command prompt (Start -> Run -> cmd.exe). In the command prompt change to the cppClient directory and issue the following command:

```
regsvr32 ozCOMClient.dll
```

STEP 4. - Compile the source

Open the project "cppClient.sln" in Visual Studio.NET 2003 and press F5 to compile it.

SMS API for Database developers (SQL)

If you use the [OZEKI Message Server](#) you can send and receive SMS messages using a database server with the help of SQL queries.

In order to use this option, you need to have a database server (such as *Oracle, Access, MySql, MS SQL, Postgres, Sybes, etc*) installed, and there should be two database tables created: [ozekimessageout](#) and [ozekimessagein](#). One of these will be used for sending and the other for receiving SMS messages.

Ozeki Message Server will connect to the database through a standard ADO or ODBC connection (Figure 1). Using this connection it will periodically query the database table [ozekimessageout](#) for outgoing messages using a SELECT statement and will INSERT incoming messages into the database table used for incoming messages ([ozekimessagein](#)).



Figure 1. - Ozeki Message Server connecting to a database server

If you want to send a message you can issue the following SQL statement:

```
sql> INSERT INTO ozekimessageout (receiver,msg,id,status)
VALUES ('+36203105366','test message',23,'send');
```

It is important to mention that the SMS Server identifies the messages by their ids. You must make sure that you specify a unique id for each message or you should use auto_increment columns. This ID is used by Ozeki Message Server to update the status attribute of the message. For example if the message is accepted by the GSM service provider for delivery it's status field is updated to 'transmitted' to notify the database programmer what happened to the message.

The incoming messages are placed into the **ozekimessagein** table. You can see the create definition for this table at the url above as well. To view the incoming messages you can use the select statement.

```
sql> SELECT sender,msg,receivedtime FROM ozekimessagein;
+-----+-----+-----+
| sender      | msg          | receivedtime |
+-----+-----+-----+
| +36203105366 | test message | 01/11/03 08:43:32 |
+-----+-----+-----+
1 row in set (0.00 sec)
```

It is interesting that often, without any external application, you can create powerful SMS enabled solutions. You can do this with the help of database triggers or stored procedures. Here is an example:

Autoreply function with a Microsoft SQL Server:

```
CREATE TRIGGER autoreply ON ozekimessagein
FOR INSERT AS
DECLARE @tel VARCHAR(30)
SELECT TOP 1 @tel=sender FROM ozekimessagein ORDER BY ID DESC
INSERT INTO ozekimessageout (receiver,msg,status) VALUES (@tel,'Thank you for the
message','send')
GO
```

How to set up an SQL - SMS gateway

This section gives information about the major steps in setting up an SQL-SMS gateway. The gateway functionality uses the following system architecture: (Figure 1) An Ozeki Message Server 6 SMS Gateway software is installed on a windows server. It uses a GSM modem to send and receive the SMS messages. (IP SMS connectivity can also be used.) The Ozeki Message Server 6 SMS Gateway software connects to a database over the network using an ADO connection. The database can be a MySQL, Oracle, MSSql, SyBase, Access, PostgreSQL or any other database that has an ADO or an ODBC driver. The software uses two database tables in this database, one for outgoing and one for incoming messages. The SMS Gateway queries the outgoing database table periodically using an SQL select statement. If it finds messages it downloads them and sends them. During and after sending, it updates a status field in the database to notify the application about what happened to the message. If an incoming message is received, it is saved into the incoming database table.

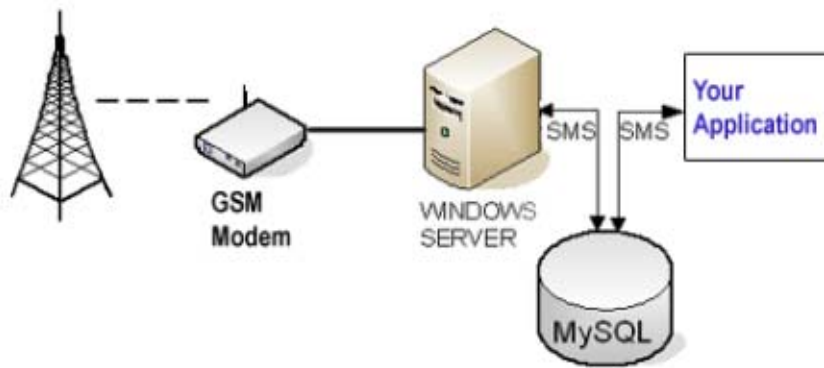


Figure 1 - SMS / SQL gateway

To create this functionality you need to go through the following steps. Most of these steps require custom action depending on you database server, but if you can manage your database server they shouldn't cause any problem.

I. Work on the database server

Step 1.) Create the database tables

First you have to create the database tables: `ozekimessageout` and `ozekimessagein`. The layout of these tables can be found on the following webpage: http://www.ozeki.hu/index.phtml?ow_page_number=422. You can put these tables in any existing database or you can create a new database that would only have these two tables. For example in a MySQL environment, you can create a text file called `createdb.sql`, copy the SQL table definitions into this text file and could execute the following commands:

```
C:\mysql> echo "create database ozekidb" | mysql -u root
C:\mysql> type createdbmysql.sql | mysql ozekidb -u root
```

Step 2.) Create a user account that Ozeki Message Server 6 can use to access these tables. Make sure this user account has `SELECT`, `INSERT`, `UPDATE` permissions for both of the newly created database tables: `ozekimessageout` and `ozekimessagein`. On MySQL you would use the following command to create the user account:

```
C:\mysql> mysql mysql -u root
mysql> insert into user values ('%', 'winduser', password('winpass'), 'Y', ...
C:\mysql> mysqladmin flush-privileges -u root
```

Step 3.) Set up the firewall. If the database server is located on a different computer, you might need to configure your firewall to allow Ozeki Message Server 6 to connect to your database server remotely. For example if your MySQL database server is located on a Linux box, you would use the following command to allow connection on port 3306 to your MySQL server. You can do this by issuing the following command:


```
[root@linux root]# iptables -I INPUT -p tcp --dport 3306 -j ACCEPT
```

II. Work on the SMS gateway functionality

Step 1.) [Install Ozeki Message Server 6 on your Windows Server](#)

In your SQL - SMS gateway, the SMS sending and receiving will be handled by Ozeki Message Server 6. This means that you need to install it onto your Windows computer. You can have Ozeki Message Server 6 and your database server installed on different computers but for higher performance it's better to have them on the same PC. The installation procedure for Ozeki message Server 6 is explained in the [Ozeki Message Server 6 Installation guide](#), the latest version of this guide can be found on the following URL: http://www.ozeki.hu/index.php?ow_page_number=185

Step 2.) [Configure the GSM Modem](#)

If you have installed Ozeki Message Server 6, you need to configure a driver to have SMS connectivity. You can set up one or more GSM modems or IP SMS connections.

Step 3.) [Optionally install an ODBC driver](#)

Microsoft Windows has built in drivers for many database servers (MS SQL, Oracle, Access, etc), but some database servers require you to install a database client that will make ODBC connectivity or ADO connectivity possible. For example MySQL requires you to have MyODBC installed.

Step 4.) [Install the Database plugin](#)

Once you have the database client or appropriate ODBC driver installed, you can start the installation and the configuration of the Database plugin in Ozeki Message Server 6. The database plugin installation guide (latest version is available at http://www.ozeki.hu/index.php?ow_page_number=535) gives you all the possible options you can use to set up the database plugin. The most important things to configure in the database configuration form is the database connection string and the SQL command templates. The database connection string contains the connection parameters for your database server. It can be entered manually or can be created with the "Build database connection" button. The http://www.ozeki.hu/index.php?ow_page_number=37 webpage and the next chapter gives information about the possible connection strings. The SQL command templates describe the SQL statements that are used to work with the database. You can see how the SQL templates can be configured in the following chapters.

Step 5.) [Modify the service user](#)

The last step to get the SMS - SQL gateway working is the changing of the service user. By default windows services are executed by the SYSTEM account on Windows computers. Since Ozeki Message Server 6 runs in the background as a system service it is also executed by the SYSTEM account. Unfortunately on most systems the SYSTEM user does not have access to the database settings configured using the user account you use when you log onto your Windows system. This can be worked around if you change the service Log On parameter for the Ozeki Message Server 6 service. The following webpage and the next

chapter gives you information on how this can be achieved: http://www.ozeki.hu/index.php?ow_page_number=536

Sql create table scripts

You can use any database with that can be accessed using an ADO or ODBC driver. All you have to do is create two database tables: **ozekimessagesin** and **ozekimessageout**. (The tables can have additional columns.) Ozeki Message Server 6 will insert the incoming messages to the **ozekimessagesin** table. If you want to send a message, you should insert a record into the **ozekimessageout** table. The SMS Server checks this table periodically for messages to send.

Microsoft SQL Server

```
CREATE TABLE ozekimessagein ( id int
IDENTITY(1,1),
sender varchar(30),
receiver varchar(30),
msg varchar(160),
senttime varchar(100),
receivedtime varchar(100),
operator varchar(100),
msgtype varchar(160),
reference varchar(100)
);
```

```
CREATE TABLE ozekimessageout ( id int
IDENTITY(1,1),
sender varchar(30),
receiver varchar(30),
msg varchar(160),
senttime varchar(100),
receivedtime varchar(100),
reference varchar(100),
status varchar(20),
operator varchar(100),
msgtype varchar(160)
);
```

MySQL

```
CREATE TABLE ozekimessagein (
id int(11) NOT NULL auto_increment,
sender varchar(30) default NULL,
receiver varchar(30) default NULL,
msg varchar(160) default NULL,
senttime varchar(100) default NULL,
receivedtime varchar(100) default NULL,
```

```
operator varchar(100),
msgtype varchar(160) default NULL,
reference varchar(100) default NULL,
PRIMARY KEY (id)
);

CREATE TABLE ozekimessageout (
id int(11) NOT NULL auto_increment,
sender varchar(30) default NULL,
receiver varchar(30) default NULL,
msg varchar(160) default NULL,
senttime varchar(100) default NULL,
receivedtime varchar(100) default NULL,
reference varchar(100) default NULL,
status varchar(20) default NULL,
msgtype varchar(160) default NULL,
operator varchar(100),
PRIMARY KEY (id)
);
```

Interbase

Installation instructions for Interbase:

http://www.ozeki.hu/index.phtml?ow_page_number=177

```
CREATE TABLE ozekimessageout (
id int NOT NULL,
sender varchar(30),
receiver varchar(30),
msg varchar(160),
senttime varchar(100),
receivedtime varchar(100),
reference varchar(100),
operator varchar(120) default NULL,
msgtype varchar(160) default NULL,
status varchar(20),
PRIMARY KEY (id)
);

CREATE GENERATOR sms_id;

CREATE TRIGGER set_id FOR ozekimessageout
BEFORE INSERT AS
BEGIN
New.id = gen_id(sms_id,1);
END;

CREATE TABLE ozekimessagein (
```

```
id int NOT NULL,  
sender varchar(30),  
receiver varchar(30),  
msg varchar(160),  
operator varchar(120) default NULL,  
msgtype varchar(160) default NULL,  
senttime varchar(100),  
receivedtime varchar(100),  
PRIMARY KEY (id)  
);  
  
CREATE GENERATOR sms_id_  
  
CREATE TRIGGER set_id_ FOR ozekimessagein  
BEFORE INSERT AS  
BEGIN  
New.id = gen_id(sms_id_,1);  
END;
```

Oracle

```
CREATE TABLE ozekimessagein (  
id int,  
sender varchar(30) default NULL,  
receiver varchar(30) default NULL,  
msg varchar(160) default NULL,  
operator varchar(120) default NULL,  
msgtype varchar(160) default NULL,  
senttime varchar(100) default NULL,  
receivedtime varchar(100) default NULL,  
reference varchar(100) default NULL  
);  
  
CREATE SEQUENCE X;  
  
CREATE TRIGGER ozekimessagein_auto  
BEFORE INSERT on ozekimessagein  
for each row  
when (new.id is null)  
begin  
SELECT x.nextval INTO :new.id FROM DUAL;  
end;  
/  
CREATE TABLE ozekimessageout (  
id int,  
sender varchar(30) default NULL,  
receiver varchar(30) default NULL,  
msg varchar(160) default NULL,  
senttime varchar(100) default NULL,
```

```
receivedtime varchar(100) default NULL,  
operator varchar(120) default NULL,  
msgtype varchar(160) default NULL,  
reference varchar(100) default NULL,  
status varchar(20) default NULL  
);  
  
CREATE SEQUENCE Y;  
  
CREATE TRIGGER ozekimessageout_auto  
BEFORE INSERT on ozekimessageout  
for each row  
when (new.id is null)  
begin  
    SELECT y.nextval INTO :new.id FROM DUAL;  
end;  
/
```

PostgreSQL

```
CREATE TABLE ozekimessagein (  
id serial,  
sender varchar(30),  
receiver varchar(30),  
msg varchar(160),  
senttime varchar(100),  
receivedtime varchar(100),  
operator varchar(100),  
msgtype varchar(160),  
reference varchar(100)  
);  
  
CREATE TABLE ozekimessageout (  
id serial,  
sender varchar(30),  
receiver varchar(30),  
msg varchar(160),  
senttime varchar(100),  
receivedtime varchar(100),  
reference varchar(100),  
status varchar(20),  
operator varchar(100),  
msgtype varchar(160)  
);
```

SQL Templates

Ozeki Message Server uses SQL templates, to communicate with the Database Server. When an SQL action needs to be taken, the appropriate SQL template is selected and the keywords in the template are replaced to the corresponding elements in the message. The SQL templates can be modified if the tables structure is different from the default.

The SQL templates can be changed on the SQL templates tab of the Database plugin configuration form (Figure 1).

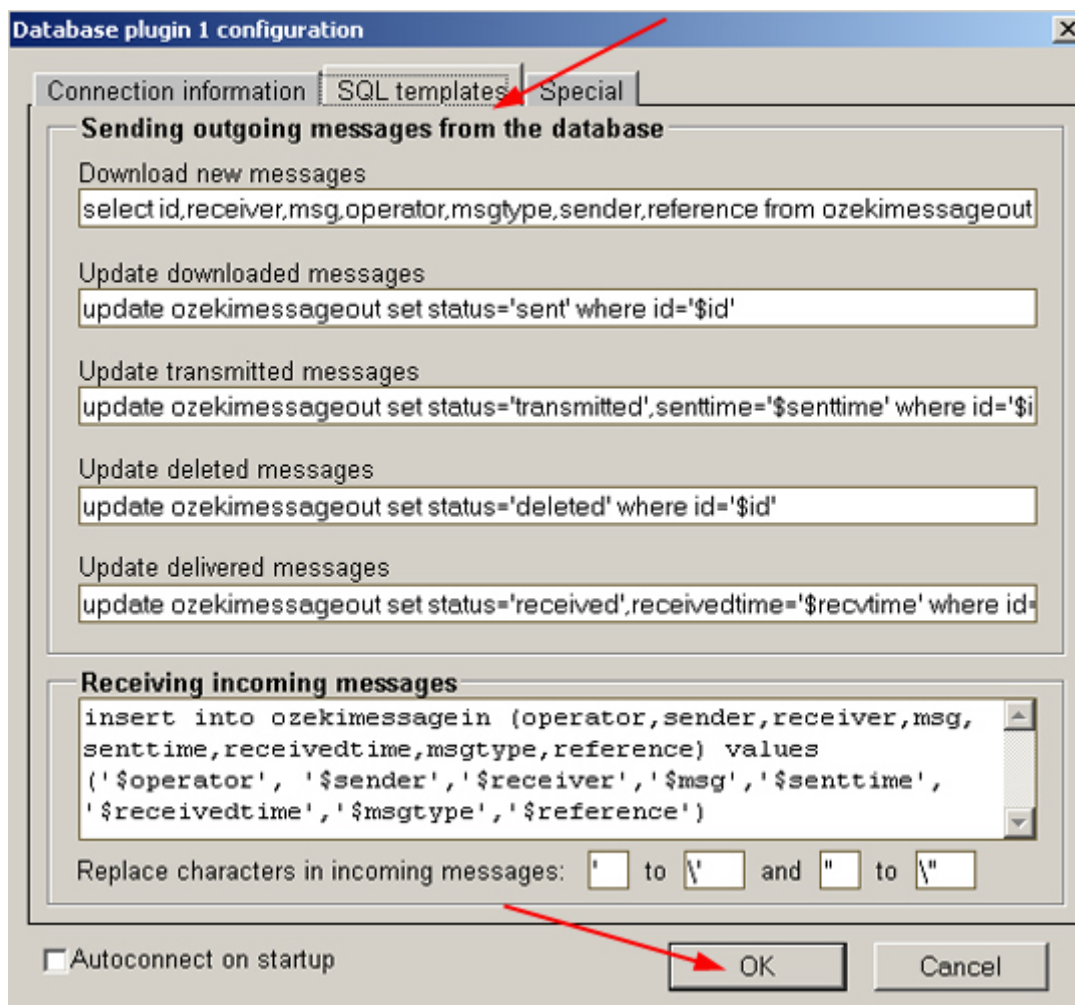


Figure 1. - SQL Templates

On the bottom of the SQL template tab of the database configuration form you can define character replacements. This is necessary because if an incoming message contains an apostrophe (') or a quotation mark ("), the SQL statement becomes invalid when the message text is inserted into the template.

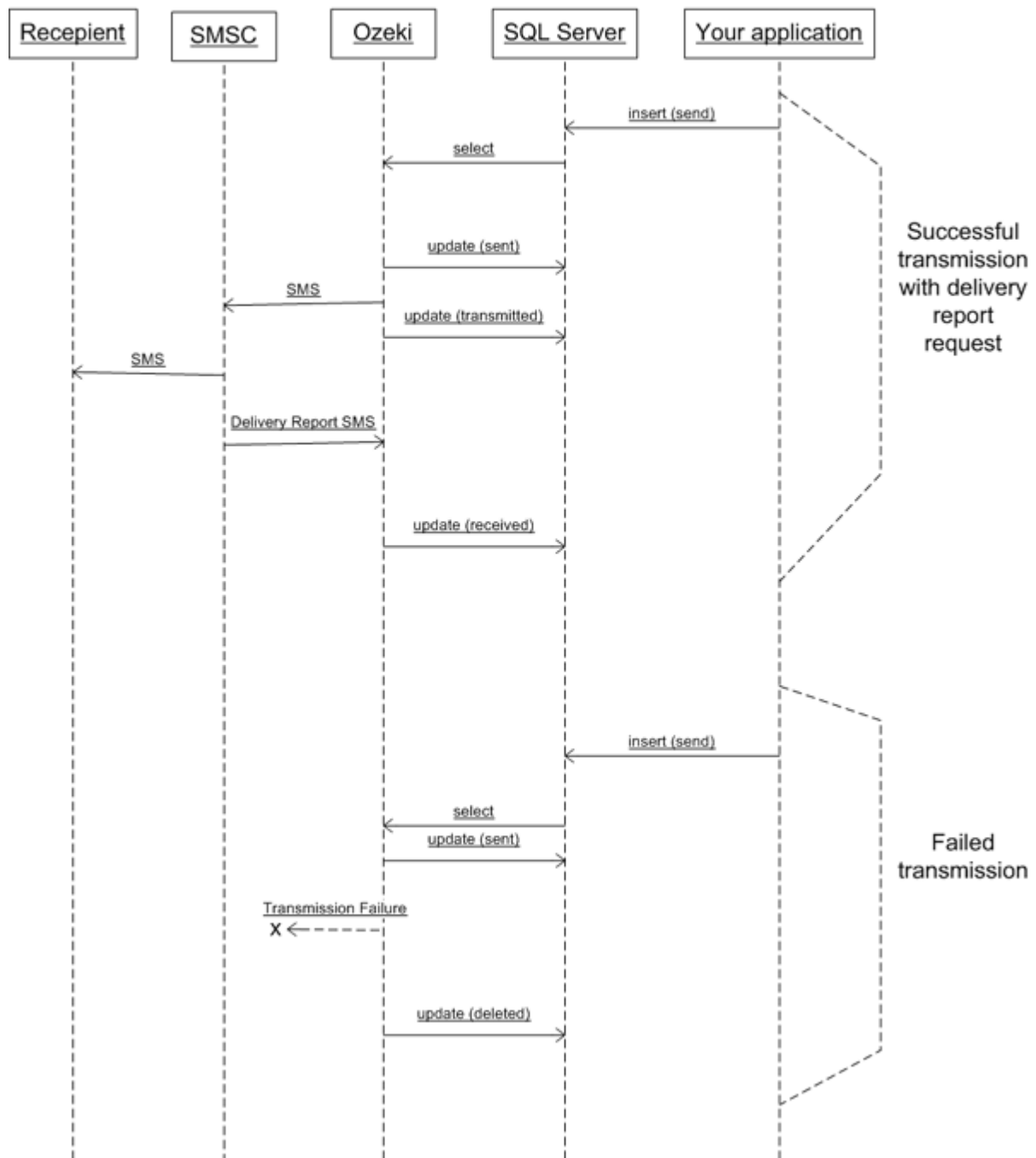
Ozeki message server 6 provides a mechanism to replace the apostrophe (') and the quotation mark (") in the received message before it is inserted into the SQL template. On the Database settings form you can specify what character sequence should be used instead of these characters.

Note: On Sybase, MSSql and on MySql, you should substitute the apostrophe (') with backslash and apostrophe (\') and you should do the same for quotation marks.

Message state updates

The ozekimessageout table has a field called "status". This field contains information about the state of the outgoing message. When a new message needs to be sent, the status field should be set to **"send"**. This value indicates that the SMS should be downloaded from the outgoing message table for delivery by Ozeki Message Server 6. After this download happens Ozeki will update the status to **"sent"**, to make sure the next time this message is not downloaded again. If the message is in the "sent" state it means it is waiting in the outgoing message queue of Ozeki, but it is not yet sent to the provider. If the GSM service provider accepts the message it's state is updated to **"transmitted"**. If the message is not accepted by the provider for delivery, for example due to an invalid telephone number, the state is changed to **"deleted"**. In some situations the provider can send a delivery report back to the system if the message is successfully delivered to the destination phone. In this case the message state is changed to **"received"** from **"transmitted"**.

**Ozeki Message Server
SQL Status updates**



Service owner, resolving connection problems

In some situations the actions taken by the Ozeki Message Server requires user privileges. For example a database connection might not be available or an application cannot be started if a user is not logged on. In this case you have to modify the properties of the registered SMS service in order to grant the necessary privileges to the software.

For example if you cannot connect to your database server, the most probable reason is that the "database connection" you wish to use is only available for a logged on user. By default the Ozeki Messages Server runs as a System service and it does not have permission to access the "database connections" available for the user. This problem can be solved using this workaround.

1. Properties of Ozeki Message Server in Windows

By default the ozeki sms service is running using the SYSTEM user account. In order to run the service as administrator, please following these steps :

As the first open the services manager window of operating system. This can be done by clicking on Control Panel/Administration tools/Services (Figure 1.).

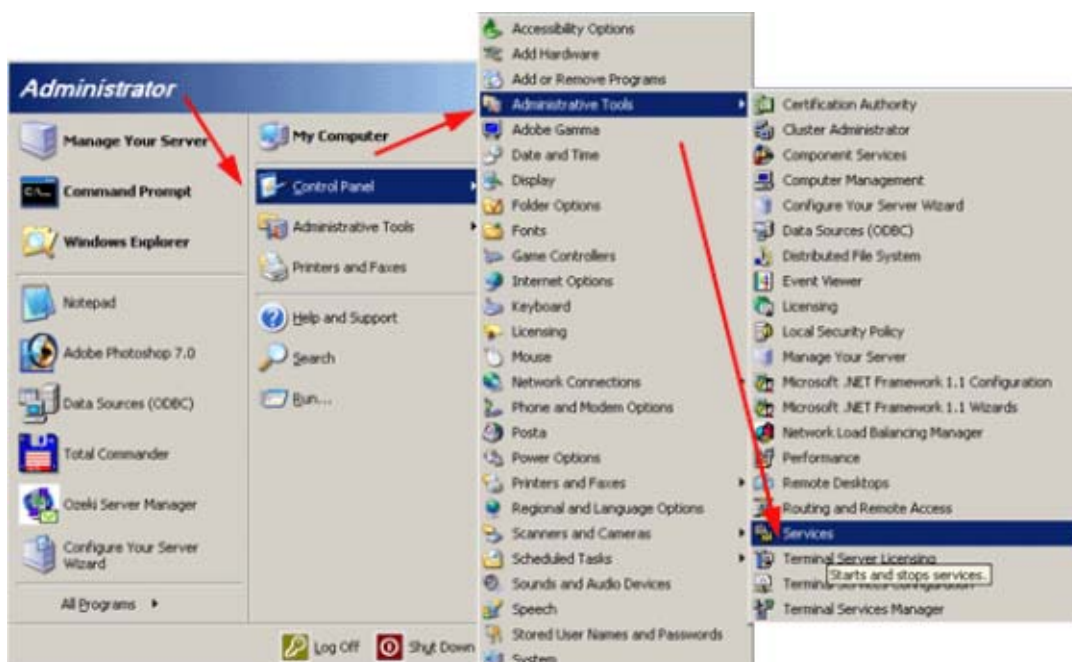


Figure 1 - Open the service manager window

In the service manager select the properties of the Ozeki Message Server item (Figure 2.).

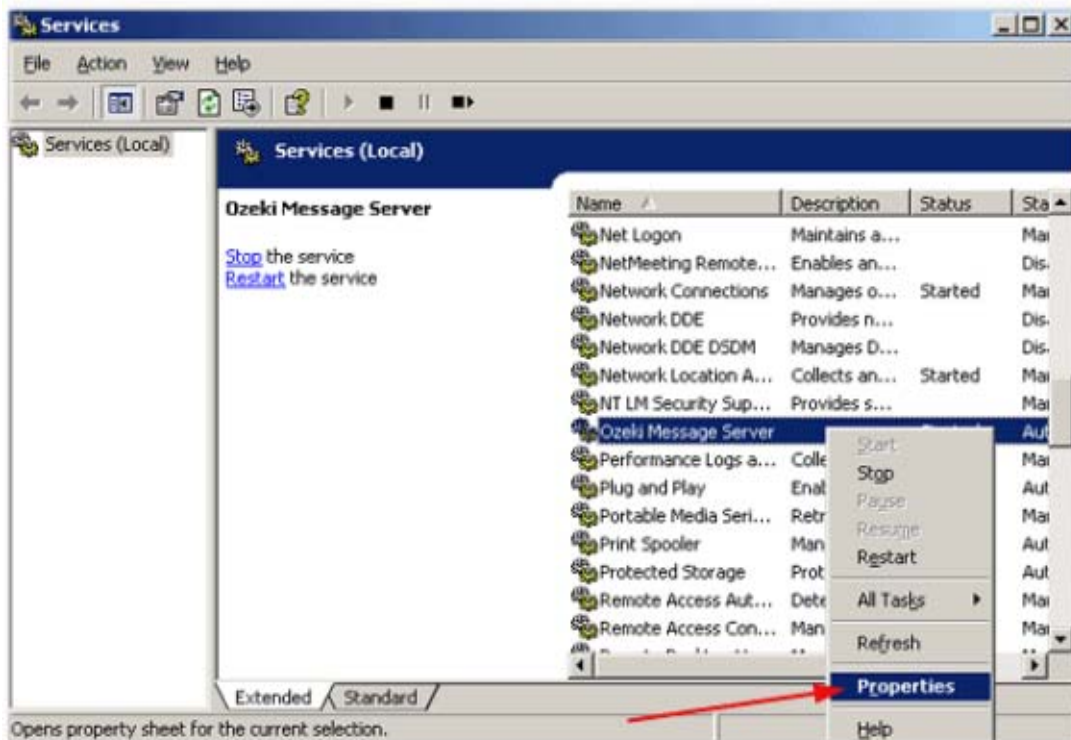


Figure 2 - Ozeki Message Server Properties

After this the Ozeki Message Server Properties form comes up. Click on the **Log on** tab. In this form the local system account is set by default. Please select the "This account" option and specify the user name and password of the administrator account (Figure 3).

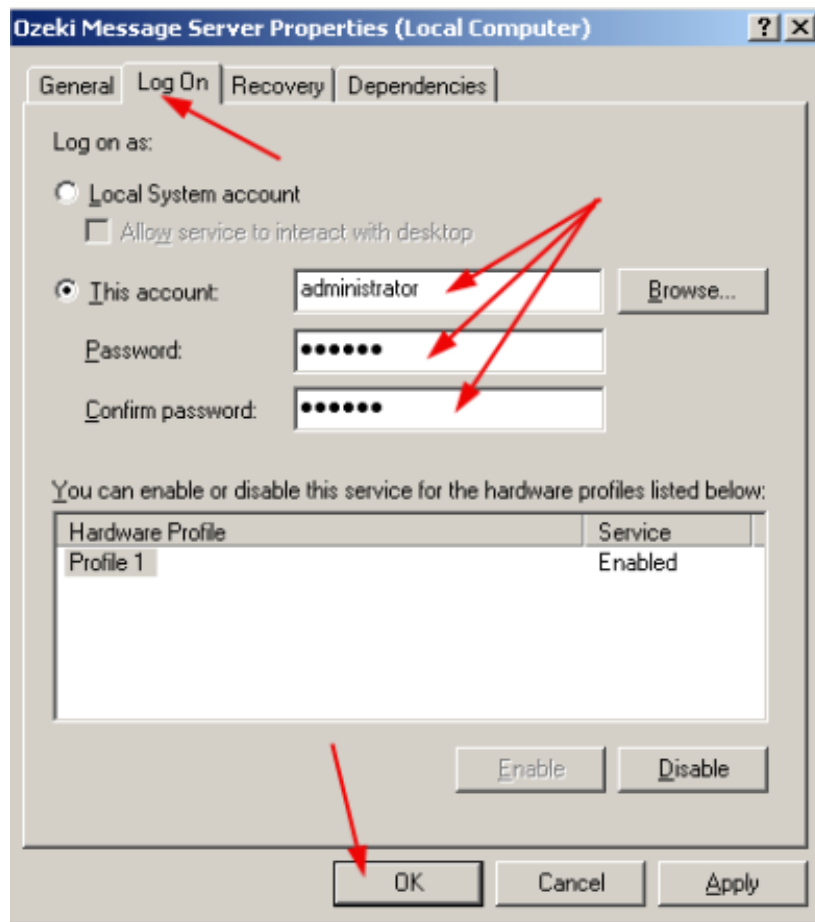
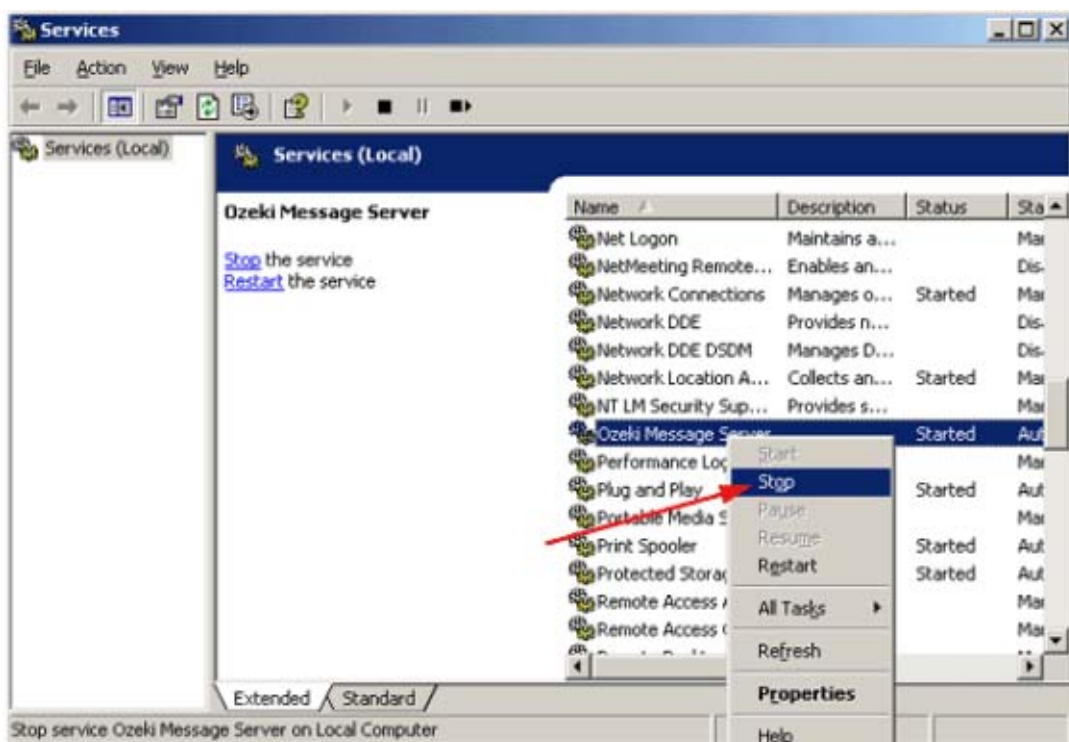


Figure 3 - selecting the administrator account

After this you have to stop and re start the Ozeki Message 6 service (Figure 4).



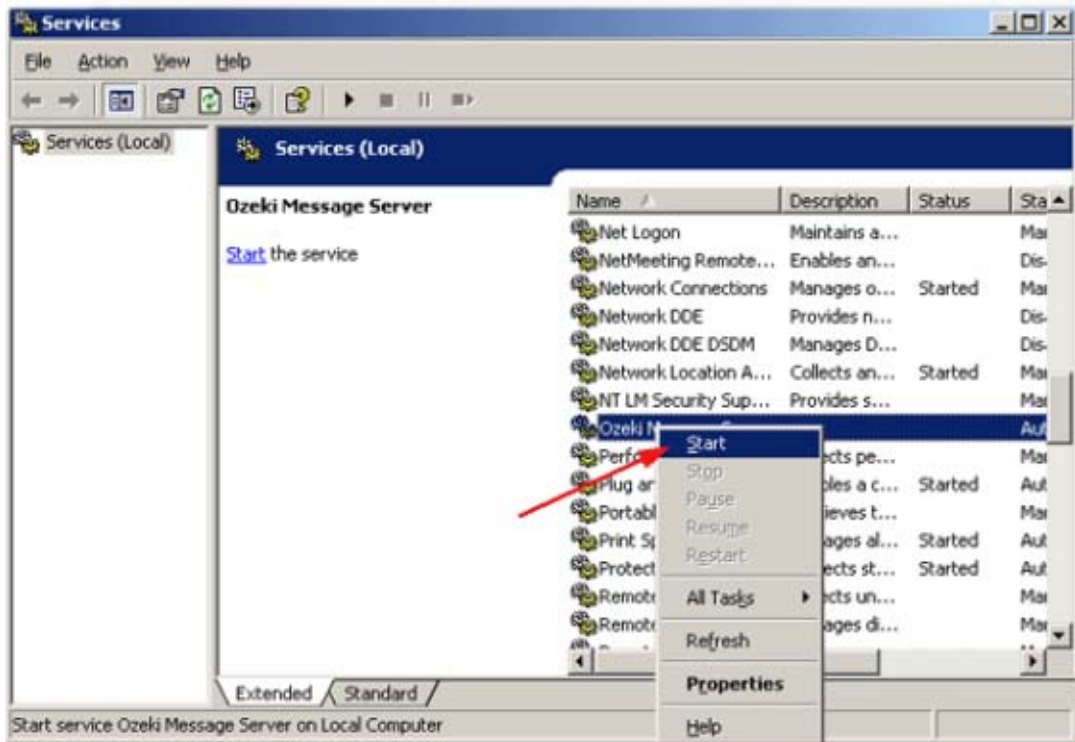


Figure 4 - Restarting the service

Problems with incoming messages

This section gives some information about what to check if the incoming messages are not inserted into the database table `ozekimessagein`. It is assumed that you can see the messages in the inbox of the admin user in the Ozeki Message Server Manager.

As a first step to resolve this issue, open the database plugin 1 configuration form from the Plugins menu. Make sure the "Use database for receiving SMS messages" checkbox is checked.

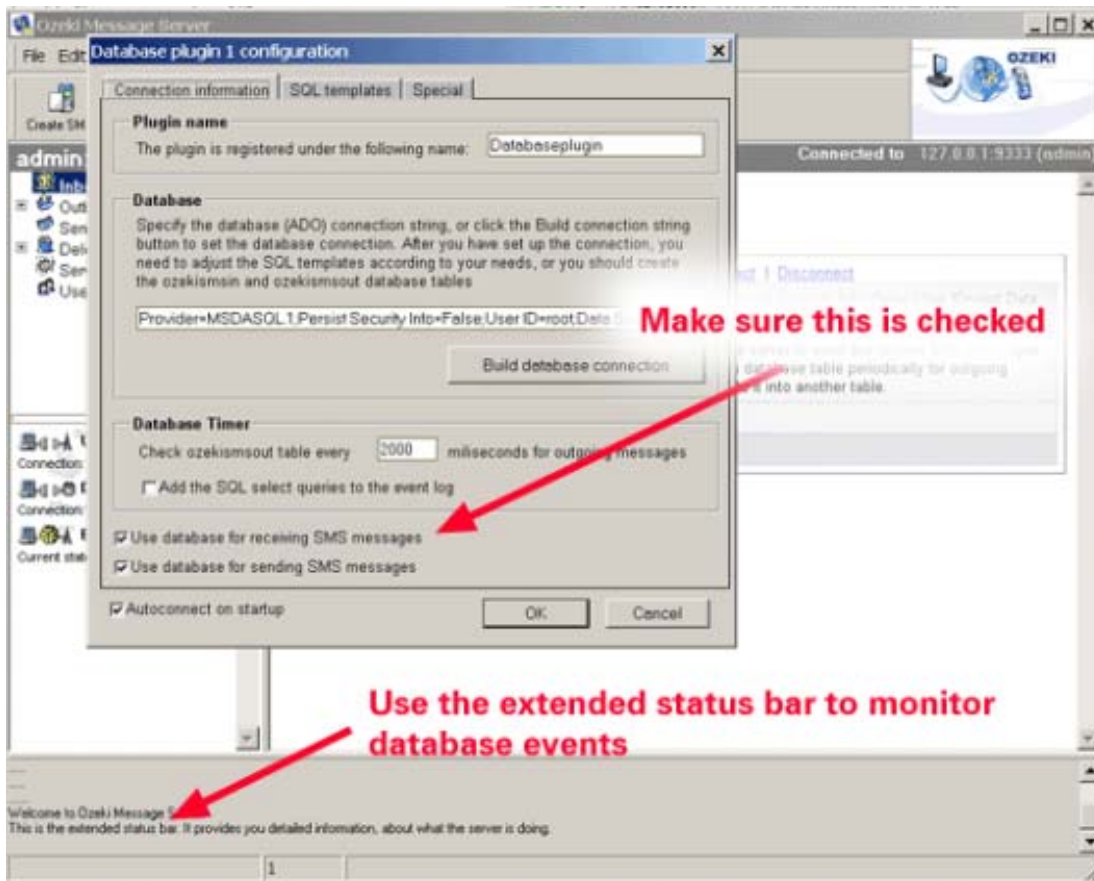


Figure 1 - Database plugin configuration

In the database plugin 1 configuration form select the SQL templates tab. On this tab you can verify the SQL template used for inserting messages into your database.

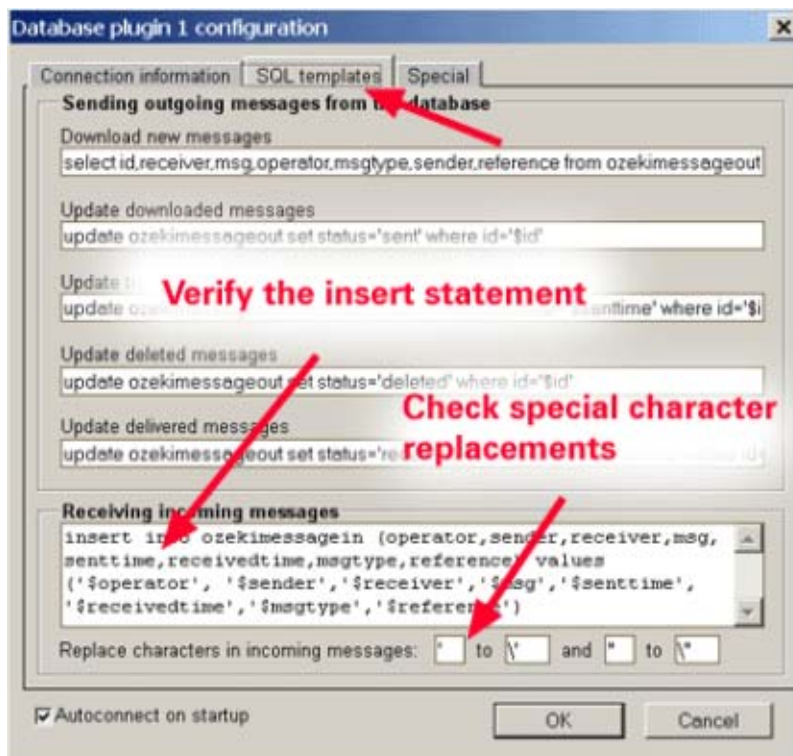


Figure 2 - SQL templates

The extended statusbar displays database events. You can open it by clicking on the standard statusbar. In most cases the you can figure out what causes the problem if you read the SQL error messages in the extended statusbar (Figure 1).

Monitoring database activity

Ozeki Message Server 6 logs every step it takes to help you keep track of your messages. You can use the Ozeki Message Server 6 Server Manager GUI to watch what is happening in real time or you can read the log files in the C:\Program Files\Ozeki\Ozeki Message Server 6\Logs directory. In the GUI, you would probably want to see the outgoing messages sent by the database. These can be found in the Users/Database Plugin 1 folder. The real time database events can be tracked in the extended status bar (Figure 1).

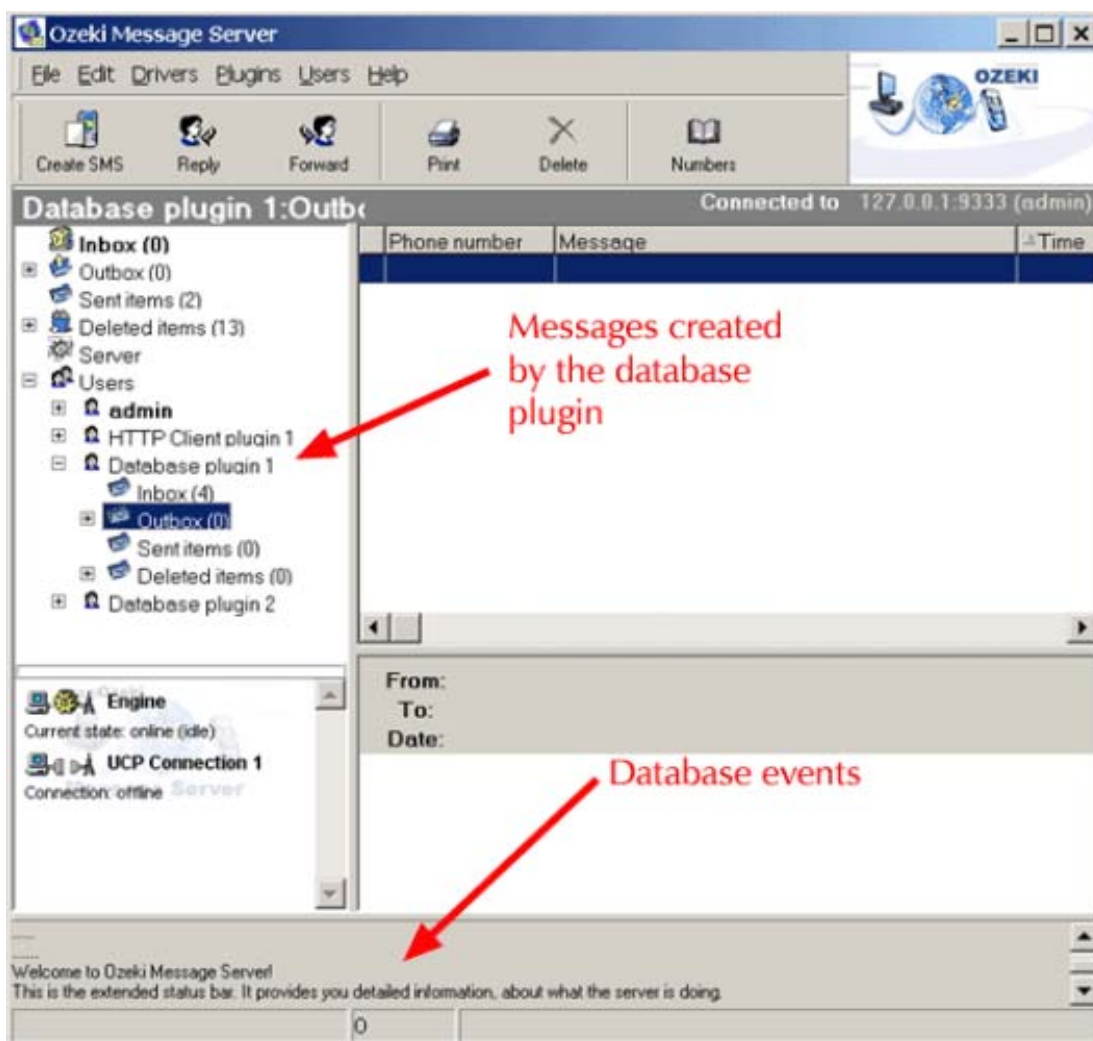


Figure 1 - Monitoring database activity

SMS API for Webdevelopers (HTTP)

Webdevelopers can use the HTTP protocol to send and receive SMS messages. Ozeki message server can be configured to act as an HTTP Client and/or as an HTTP server. After the configuration the webdeveloper can send and receive SMS messages using HTTP queries.

SMS API for webdevelopers (HTTP Client plugin)

Web applications communicate with the user with HTML forms. When a user wants to send information to the server, he fills out a form. The HTTP-SMS client plugin does the same. It fills out a form automatically every time an SMS message is received. The receiving web application can return a response SMS message after processing the received data.

THE INSTALLATION

The installation of the HTTP Client plugin is similar to the installation of the other plugins. In this chapter we give the instructions on how to configure this plugin.

PLUGIN CONFIGURATION

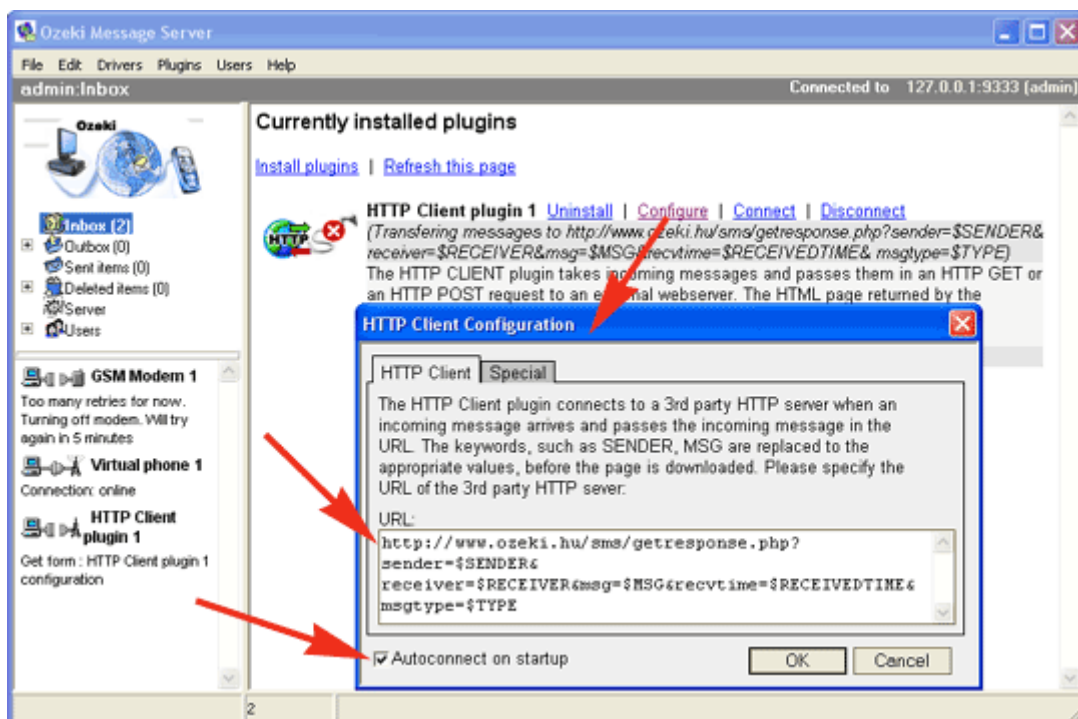


Figure 5 - Automatic startup

To start the configuration, click on the [Configure](#) URL. If you would like the HTTP server of the plugin to start automatically when the Ozeki Message Server software starts, place a checkmark into the "**Autoconnect on startup**" checkbox and press **Ok**. (Figure 5.) The URL shows the 3rd party HTTP server.

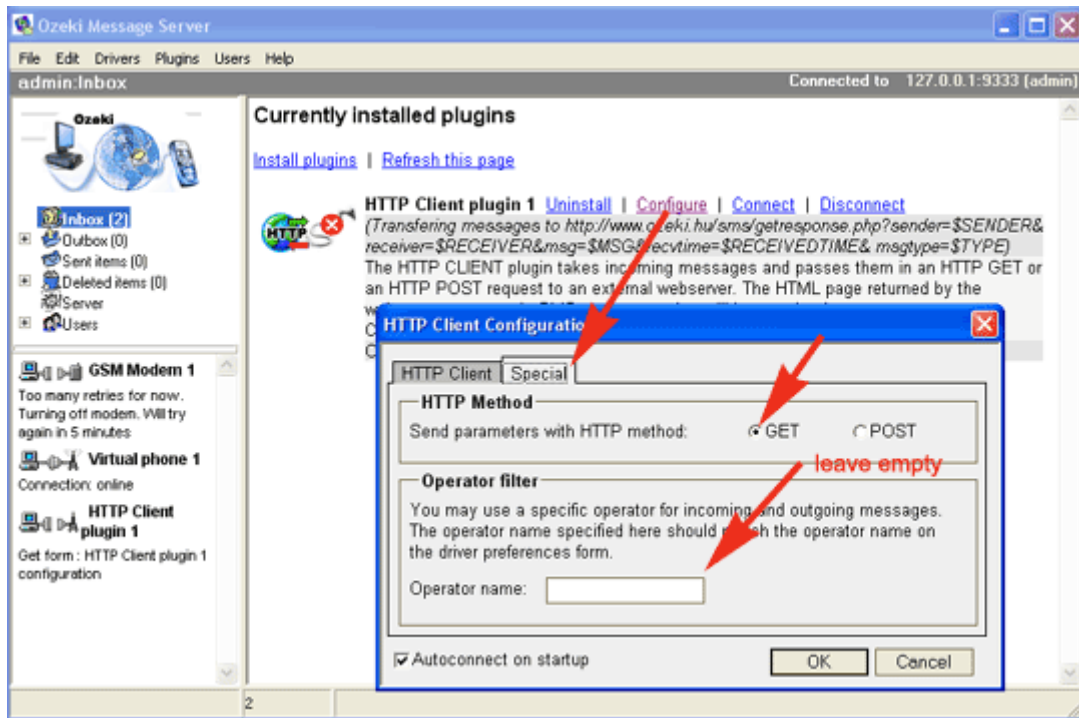


Figure 6 - HTTP method

You can select the **HTTP method** with the radio buttons. Leave the *Operator name:* text box blank.

ROUTE THE INCOMING MESSAGES TO THE PLUGIN

To receive the messages route the incoming SMS messages to the plugin.

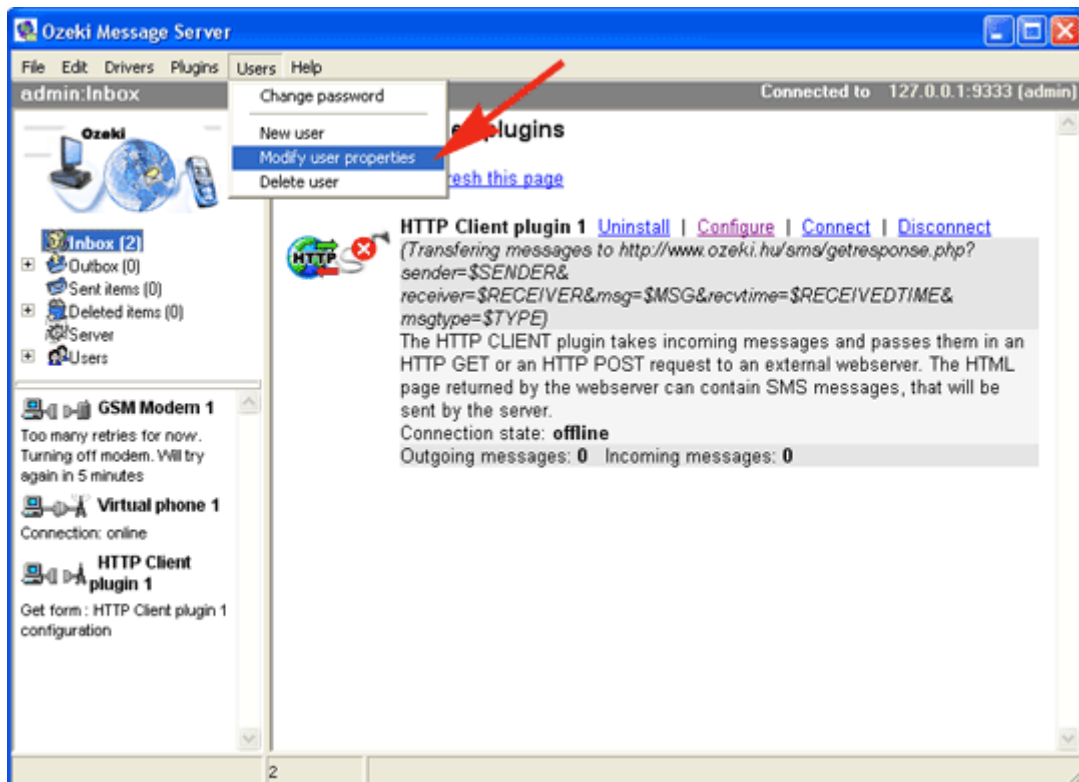


Figure 7 - Modify the user properties of the HTTP plugin

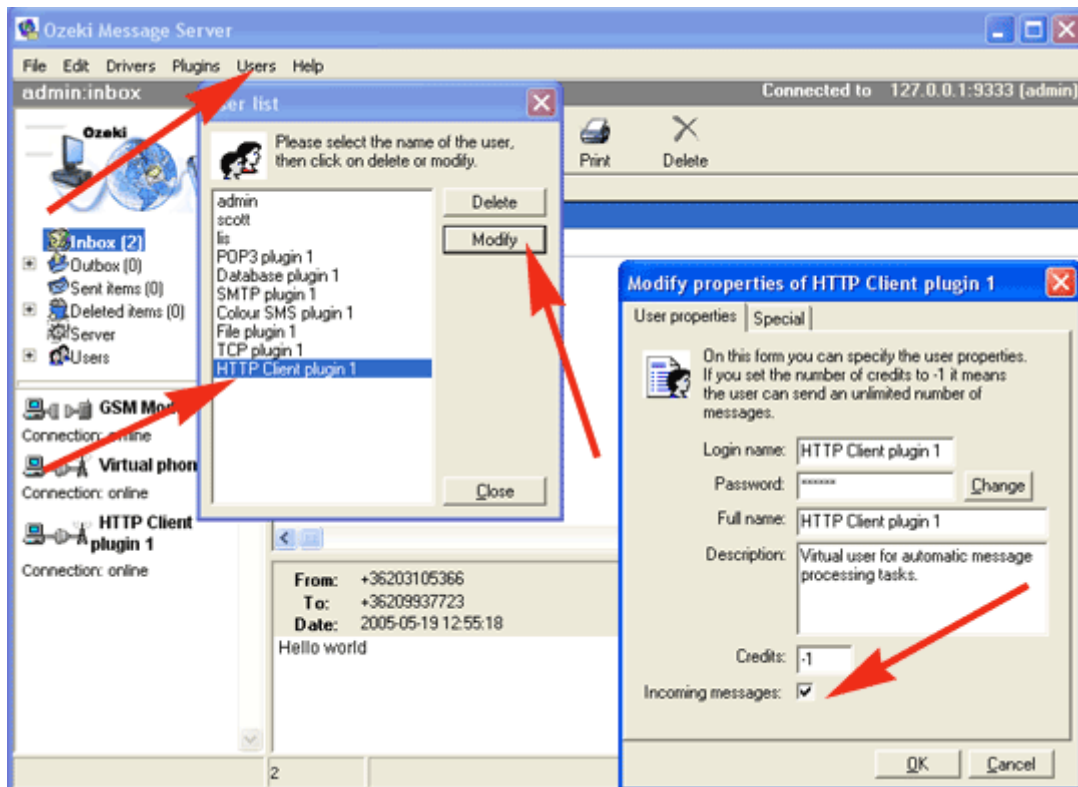


Figure 8 - Incoming messages check in

Select the *HTTP Client plugin 1* from the list and click on the **Modify** button. In the new windows check in the *Incoming messages:* checkbox. (Figure 8.)

TURN ON LOGGING

You can create the HTTP log file if you turn it on (Figure 9.) .

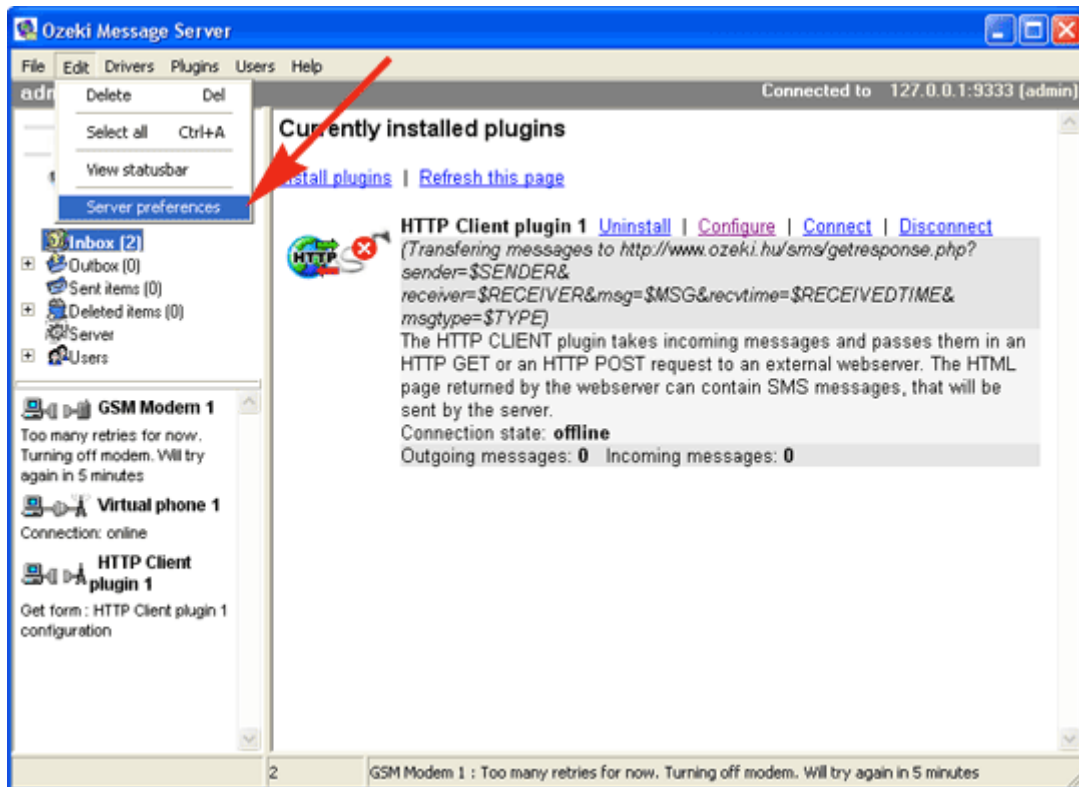


Figure 9 - Server preferences

Select the **Edit/Server preferences** menu.

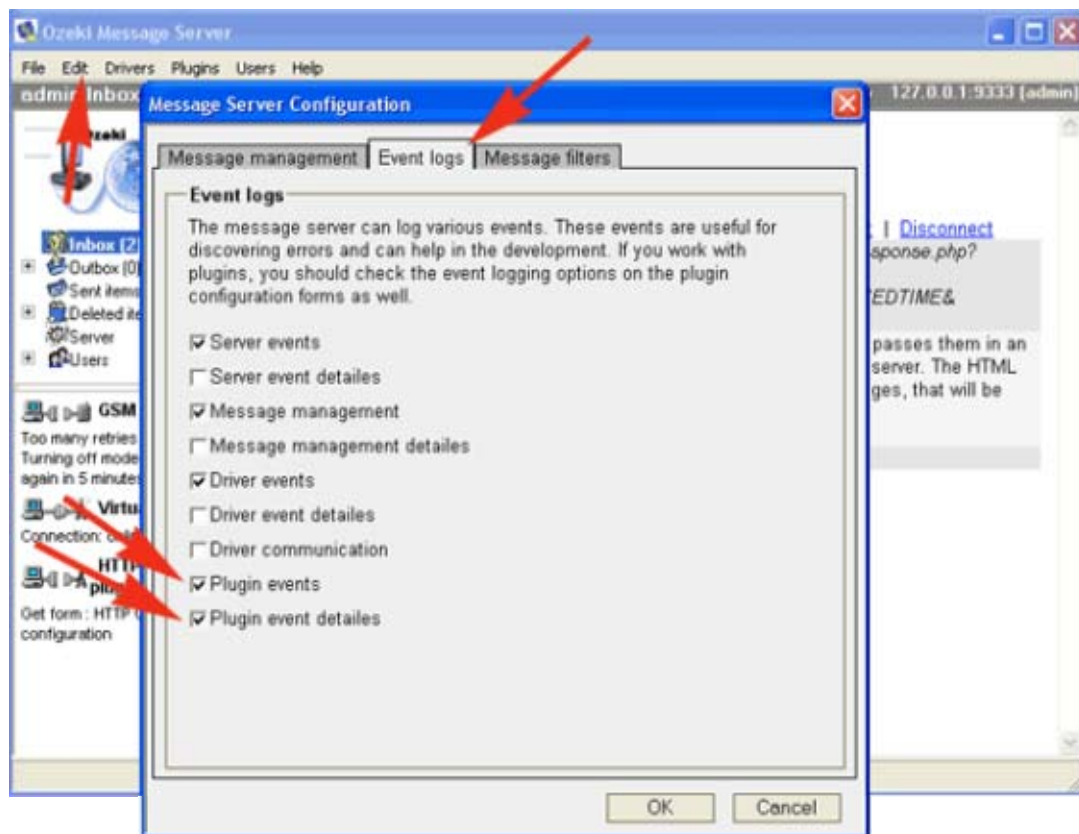


Figure 10 - Event logs

Click on *Event logs* tab and check in the **Plugin events** and **Plugin event details** checkboxes.

STARTING SENDING AND RECEIVING SMS

Send an SMS to the phone attached to the PC. This will trigger the HTTP client request. The http client request means that a webpage from a specified webserver is downloaded. If this webpage returns plain text content containing one more lines with this syntax, response SMS messages can be generated.

```
{GSMSMS}{ }{ +3620555224}{ +3620111245}{Thank you for the answer}
```

In this format the +3620555224 number is the sender number. It is optional. The +3620111245 recipient number and the message text is mandatory.

VIEW THE LOG

Check the plugin functionality

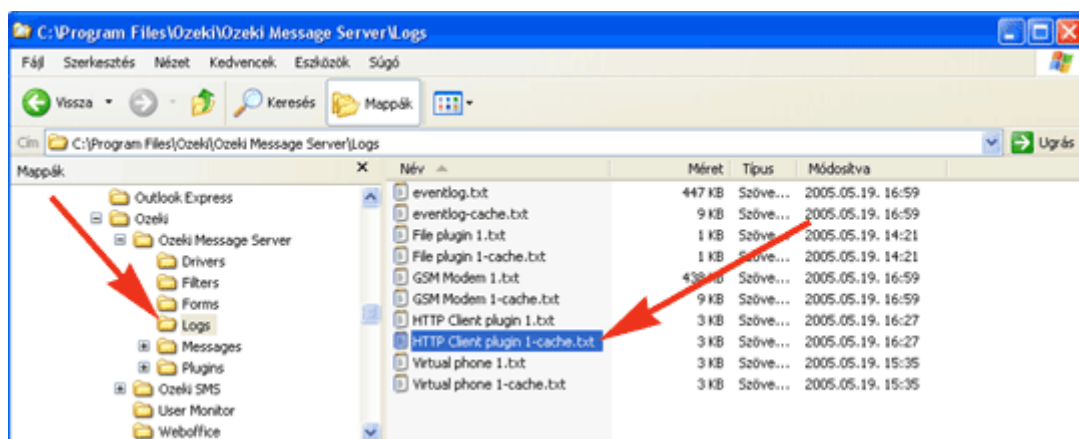


Figure 12 - The login file path

Open the log file:

C:\Program Files\Ozeki\Ozeki Message Server\Logs\HTTP Client plugin 1-cache.txt

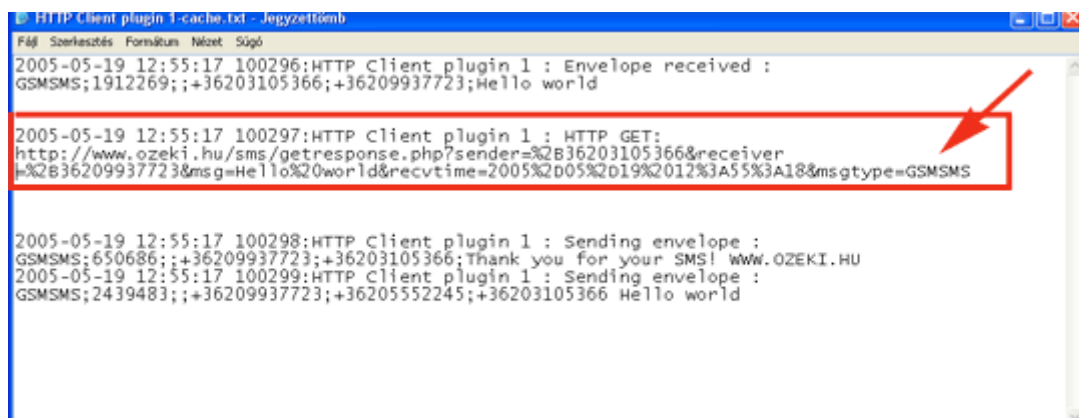


Figure 13 - In the login file

SMS API for webdevelopers (HTTP Server)

Ozeki Message Server 6 has a built in HTTP server. This means, that simply using HTTP client requests you can send / receive SMS messages and you can control the SMS server. This guide explains the basic HTTP interface elements.

I. How to check the number of credits

HTTP Request:

<http://127.0.0.1:9333/ozeki?action=getCredits&ozmsUserInfo=admin:abc123>

or

<http://127.0.0.1:9333/ozeki?action=getCredits&ozmsUserInfo=admin:e99a18c428cb38d5>

HTTP Response:

```
<Response><Action>getCredits</Action><Data>-1</Data></Response>
```

Note:

- The number of credits determine how many SMS messages can be sent by the user. If this number is -1 it means unlimited.
- The **action** parameter specifies the command you are executing. In this case the command is `getCredits`
- The **ozmsUserInfo** parameter specifies the user. You can put the user information in plain text, or you can use the [md5 checksum](#) of the password, which would give you a little more protection.

II. How to send an SMS message

HTTP Request:

<http://127.0.0.1:9333/ozeki?action=sendMessage&ozmsUserInfo=admin:abc123&recepier>

HTTP Response:

```
<Response>
  <Action>sendMessage</Action>
  <Data>[ACCEPTREPORT]</Data>
</Response>
```

ACCEPTREPORT:

```
<AcceptReport>
  <StatusCode>0</StatusCode>
  <StatusText></StatusText>
  <MessageID>1737016</MessageID>
```



```
</AcceptReport >
```

Note:

- When you send a message you must specify the **recipient** and the **messageData** fields. The values must be urlencoded.
- The **action** parameter specifies the command you are executing. In this case the command is `sendMessage`
- The **ozmsUserInfo** parameter specifies the user information.
- If the message is accepted for delivery you receive an AcceptReport containing a **StatusCode** of zero (0) and a **message id** string
- If the message is not accepted for delivery the **StatusCode** is greater than zero and the **StatusText** contains the error message
- If this user name and password is invalid, the server returns an HTML page containing a login prompt.

Send SMS messages using text files

OZEKI Message Server empowers you to send and receive SMS from a personal computer through a GSM device attached to the computer or by using an IP SMS connection. The sending task could be performed by placing txt files into a certain directory. Receiving can be done by reading txt files from another directory in the file system.

If you want to send and receive SMS this way, you should configure the OZEKI SMS Server to place the incoming messages into the selected directory and to check another directory periodically for messages to send. The directory we use for receiving messages is called the incoming directory, the one we use for sending messages is called the outgoing directory.

The installation and configuration instructions for this functionality can be found on the following page: [File Plugin Installation](#)

When the directory settings are configured, an appropriate txt file format should be selected. Ozeki Message Server supports three kinds of file formats. You should select the one that fits your needs the most. You can read about the file formats on the [file format information page](#).

Sending an SMS

To perform sending 3 directories are used:

- 1.) C:/smsout/ - the outgoing directory
- 2.) C:/smssent/ - the directory that holds the messages that were successfully sent
- 3.) C:/smsfailed/ - the directory that holds the messages that could not be sent.

To send an SMS first you should create a "lock" file in the outgoing directory. A lock file is used to make sure Ozeki Message Server will not start to read the real file containing the messages while the file is still being written. The lock file is an empty file, that has a name that matches the real file. The lock file name has

a special extension: .lock. After the lock file is created you can edit the real file containing the outgoing message text. When you are finished simply delete the lock file.

Step 1.)

Create: C:/smsout/mysms.sml.lock

Step 2.)

Create: C:/smsout/mysms.sml

Write : C:/smsout/mysms.sml

Step 3.)

Delete: C:/smsout/mysms.sml.lock

After you have created your file, you need to wait. Ozeki Message Server will delete your file, and will try to send the messages. When the sending is done a text file will be placed into the smssent or the smsfailed directory based on the sending results using the same locking steps.

Please take a look at the following diagram:

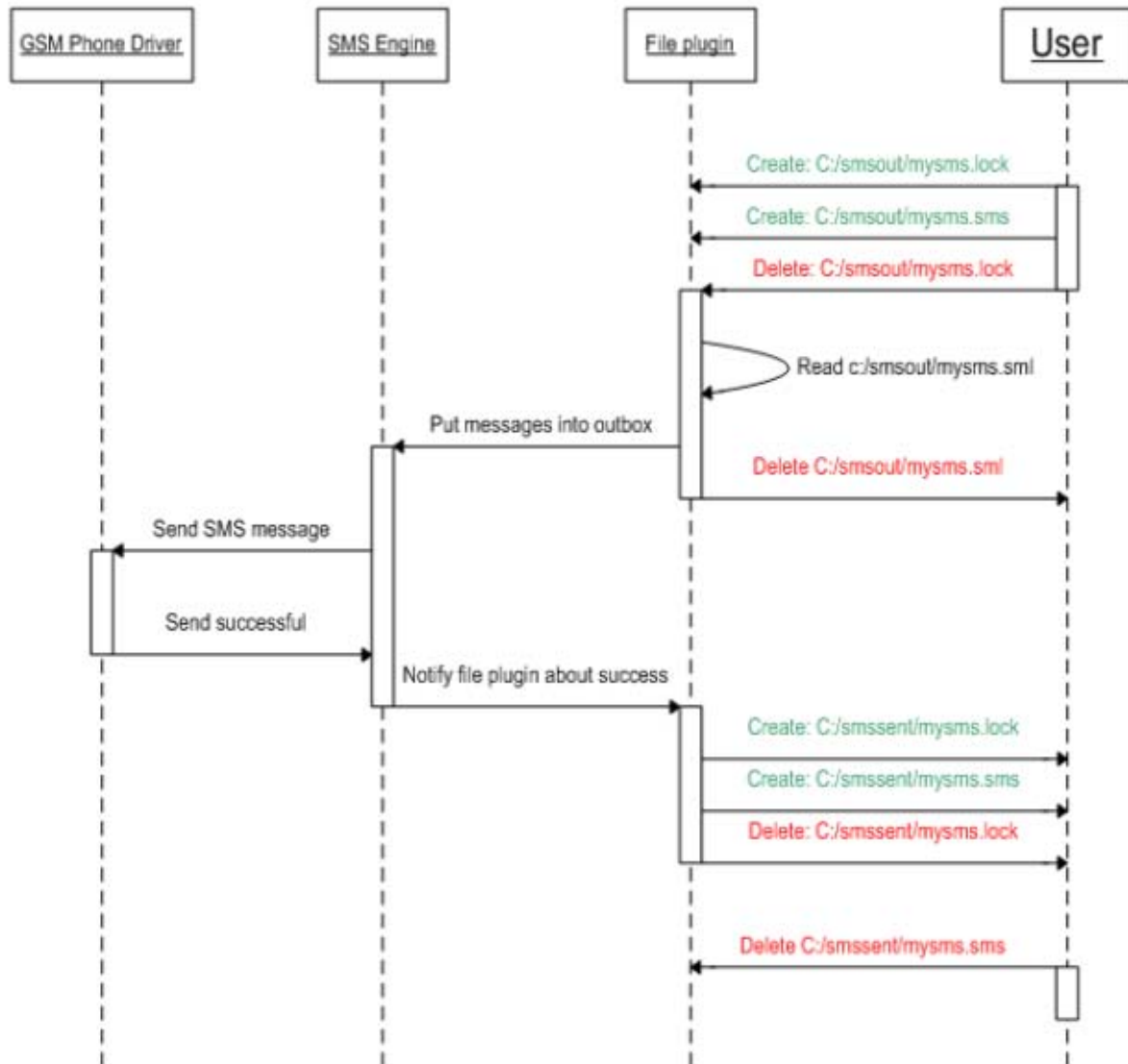


Figure 1 - Sengind an SMS using the file plugin

File plugin - file formats

This section presents the file formats you can select on the file plugin configuration page for file-to-sms gateway functionality.

THE SIMPLE FILE FORMAT

When you select the simple file format, each incoming message is saved in a separate file. The name of the file is the **phone number** and the first row of the file contains the message. (Figure 1.)

c:\smsout\+36203105366

Hello world

If you want to send messages using the simple file format, you should create a separate textfile for each outgoing message and place these files into the smsout directory.

THE VERBOSE FILE FORMAT

The verbose file format is similar to the simple file format in a way that each message resides in a separate file. The difference is that it contains more detailed information about the SMS messages. The main advantage of this file format is that more than one message could be placed into the same directory with the same destination.

If you want to send messages using this format, make sure that the extension of the file name that is placed into the outgoing directory is .sms and the file format is the following:

The first row of the file is the sender's phone number, the second row contains receiver phone number and the message text follows these in the next row. Take a look at the example:

```
c:\smsout\valamilyennev.sms
c:\smsout\anyname.sms
```

```
+36203105366
+36209937723
The message text
```

THE LIST FILE FORMAT

The list file format is a convenient way to send and receive a large number of sms messages. In this case more than one message is stored in the same file. The format of the file can be seen below.

For outgoing messages you should create a text file, where each line contains a message. The messages consist of a phone number in international format and the message text. The phone number and the message text is separated by a space character.

When an outgoing file is ready to be sent it should be placed into the smsout directory. The name of the file should have the .sml extension. (sml stands for sms list)

```
c:\smsout\lista.sml
c:\smsout\list.sml
```

```
+36205555366 a message
+36209937723 the second message
+36307789723 using Ozeki Message Server
```

SMS API for JAVA developers

The following example gives you detailed instructions on how you can send a WAP Push SMS messages from Java. The example uses HTTP Client functionality to pass the outgoing messages to the built in HTTP server of Ozeki Message Server. In the example we assume, that Ozeki Message Server is already installed and configured. A WAP push message contains a description of a WAP page and an URL that points to it.

Ozeki Message server can also be used to send text messages and various binary SMSes such as vCard and vCalendar messages, Operator Logos and Ringtones. All you have to do is change the messageType field and the message content in this example. A list of supported message types can be found on the following URL: <http://www.ozeki.hu/index.php?owpn=488>.

To send a WAP push message we use the HTTP client library of Apache.org. First you need to download this library:

Step 1. - download the apache HTTP client library

Go to the following webpage:

http://jakarta.apache.org/site/downloads/downloads_commons-httpclient.cgi, and download version 2.0.2.

After you have downloaded the file (commons-httpclient-2.0.2-src.zip) unzip it! The unzip will create the following jar file, that contains the HTTP client utilities: commons-httpclient-2.0.2.jar

Setp 2 - set the CLASSPATH to have access to the functions stored in this jar.

```
SET CLASSPATH=C:/tmp/commons-httpclient-2.0.2/commons-httpclient-2.0.2.jar
```

Step 3 - create a .java source file, that will connect to Ozeki Message Server and will post a message.

PostExample.java

```
import org.apache.commons.httpclient.*;
import org.apache.commons.httpclient.methods.*;

public class PostExample {
    public static void main( String[] args ) {
        String url = "http://127.0.0.1:9333/ozeki";
        String wapPush= " <si>"+
            "    <indication href=\"http://wap.yahoo.com/\"
action=\"signal-high\">"+
            "        A WAP Push message, that sends the browser to
wap.yahoo.com."+
            "    </indication>"+
            "</si>";
        try {
```

```
HttpClient client = new HttpClient();
PostMethod method = new PostMethod( url );

// Configure the form parameters
method.addParameter( "login", "admin" );
method.addParameter( "password", "abc123" );
// method.addParameter( "operator", "Vodafone" );
method.addParameter( "action", "sendMessage" );
method.addParameter( "messageType",
"SMS: WAPPUSH: application/vnd.wap.sic" );
method.addParameter( "recepient", "+36205222245" );
method.addParameter( "messageData", wapPush );

// Execute the POST method
int statusCode = client.executeMethod( method );
if( statusCode != -1 ) {
    String contents = method.getResponseBodyAsString();
    method.releaseConnection();
    System.out.println( contents );
}
}
catch( Exception e ) {
    e.printStackTrace();
}
}
```

Step 4 - compile the file

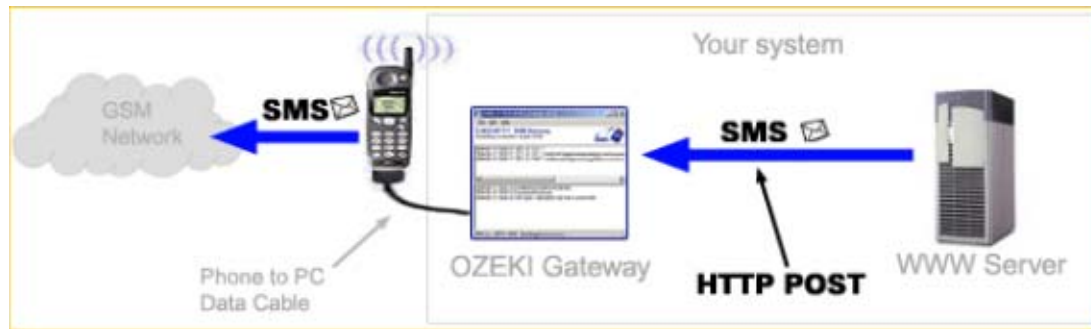
```
javac PostExample.java
```

Step 5 - send the WAP push message

```
java PostExample.java
```

SMS API for PHP developers

Ozeki Message Server can be used to send out SMS messages from any application. In order to send a message, the application must perform an HTTP request. The built in webserver of the Ozeki Message Server receives the request and adds the posted SMS message to the outgoing message queue. After some time the message will be sent. (The status of the outgoing message can be queried by subsequent HTTP requests)



To send an SMS message from PHP, you may use the following code. This example will send a text message containing the text Hello World to the telephone number +36205222245.

PHP EXAMPLE

`http://localhost/smssend.php`

```
<?php
$gatewayURL = 'http://localhost:9333/ozeki?';
$request = 'login=admin';
$request .= '&password=abc123';
$request .= '&action=sendMessage';
$request .= '&messageType=GSM SMS';
$request .= '&receptient='.urlencode('+36205222245');
$request .= '&messageData='.urlencode("Hello World");

$url = $gatewayURL . $request;

//Open the URL to send the message
file($url);
?>
```

Binary message types such as WAP PUSH messages, Operator Logos and Ringtones can also be sent using this technique. All you have to do is change the messageType field and the message content. A list of supported message types can be found on the following URL: <http://www.ozeki.hu/index.php?owpn=488>.

Creating a PHP quiz game

The Ozeki Message Server contains an HTTP Client Plugin, that can be used to develop autoresponding SMS services, such as a Quiz game. To understand how this service works, take a look at (Figure 1.). In this setup the Ozeki message server, has an [HTTP Client plugin configured](#), that acts as an HTTP - SMS gateway.

Ozeki Message Server runs as service on a Windows server. It communicates with the phone attached to this server with a phone-to-PC data cable or directly with a GSM service provider's SMSC using an IP SMS connection. Ozeki Message Server collects the SMS messages that arrive on the connection and passes them to your web application using an HTTP request. Your web application gets the

sender phone number and the message text in the parameters of the HTTP request, and can generate the HTTP response, containing response SMS messages.

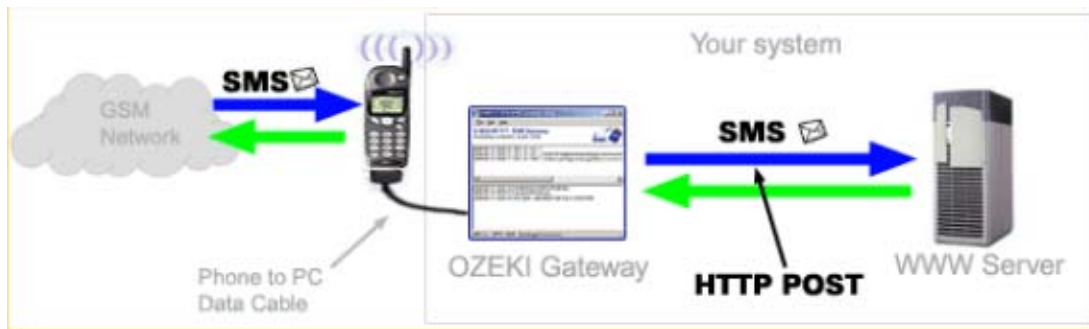


Figure 1. - Processing incoming SMS messages with a WWW server.

RECEIVING SMS MESSAGES AND SENDING A REPLY

If you are a web developer, you know, that web applications communicate with the user, using HTML forms. When a user wants to send information to the server, he fills out a form. In this case Ozeki Message Server does the same. It fills out a form automatically every time an SMS message is received. In other words it generates an HTTP GET or an HTTP POST request whenever an incoming message arrives. To pass this incoming message to your application the following URL is invoked, if the [HTTP Client Plugin](#) is configured:

<http://www.yourserver.com/smsproc.php?sender=0620522245&receiver=06203105366&message=hello+world>

The receiving web application (smsproc.php) can return a response SMS message by creating the following output.

```
{GSM SMS} {+36205552245}{Thank you for the message}
```

Note, that this output cannot contain any line-breaks!

Using this technology we can create a simple Quiz game that works the following way: If the user sends in "September", the game responds in an SMS by saying "Correct", if any other text comes in it responds: "The answer is incorrect, please try again.". Here is the source code in PHP language of this application:

PHP EXAMPLE

<http://www.yourserver.com/smsproc.php>

```
<?php
$sender      = $_REQUEST['sender'] ;
$receiver    = $_REQUEST['receiver'];
$message     = $_REQUEST['message'];
$receivedtime = $_REQUEST['receivedtime'];

/* Write the log */
```



```
<%
  Dim strMsg
  Dim strReceiver
  Dim strUrl

  ' Retrieve the posted items from the HTTP-SMS gateway
  strUrl = "http://localhost:9333/ozeki?"
  strRequest = "login=admin";
  strRequest = strRequest+"&password=abc123";
  strRequest = strRequest+"&action=sendMessage";
  strRequest = strRequest+"&messageType=GSMSMS";
  strRequest =
strRequest+"&recepient="+Server.URLEncode("+36205222245");
  strRequest = strRequest+"&messageData="+Server.URLEncode("Hello
World");

  strUrl = strUrl+strRequest;

  'Create InternetExplorer
  Dim WebBrowser: Set WebBrowser =
CreateObject("InternetExplorer.Application")

  WebBrowser.Navigate strUrl

  Do While WebBrowser.busy
  Loop
  WebBrowser.Quit

  'Display message info to the user
  Response.Write("The message has been sent . " + chr(13))
%>
```

Binary message types such as WAP PUSH messages, Operator Logos and Ringtones can also be sent using this technique. All you have to do is change the messageType field and the messageData field to the appropriate message content. A list of supported message types can be found on the following URL: <http://www.ozeki.hu/index.php?owpn=488>.

SMS API for Delphi developers

This downloadable component makes it possible to write a software in Delphi that can use the Ozeki message server to send and receive SMS messages. This component can attach to Ozeki Message Server over the network or on the localhost. It supports sending Unicode messages (eg.: Arabic)

Step 1: Download the ozclient.dll file:

[ozclient.dll](#)

Step 2: Copy the ozclient.dll file into your C:\Windows directory

Step 3: Download and unzip the following delphi component:

[DelphiSMS.zip](#)

Step 4: Install the OzekSMS.dpk file (Figure 1):

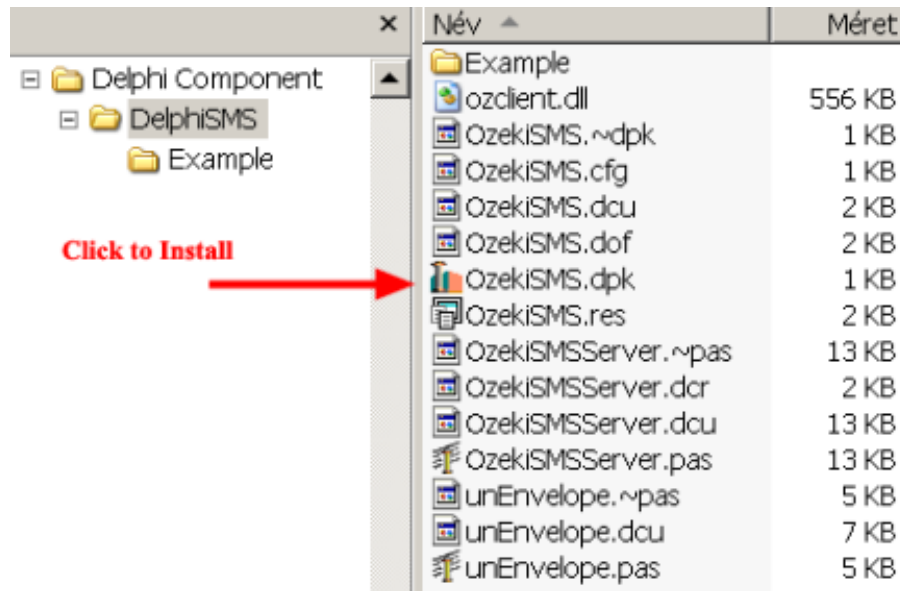


Figure 1 - DelphiSMS

Plugin development

Step 1:

Download and unzip the example source code [ExamplePlugin.zip](http://www.ozeki.hu/attachments/616/ExamplePlugin.zip) from <http://www.ozeki.hu/attachments/616/ExamplePlugin.zip>

Step 2:

Copy the **ExamplePlugin.spec** file into the following directory:
C:\Program Files\Ozeki\Ozeki Message Server\Plugins

Step 3:

Open Delphi and load the **examplePlugin.dpr** project file. The source code you have to edit can be found in the unPlugin.pas file.

Step 4:

Configure the output directory in Delphi using the Project/Options menu item (Figure 1):

Make sure it points to **C:\Program Files\Ozeki\Ozeki Message Server\Plugins**

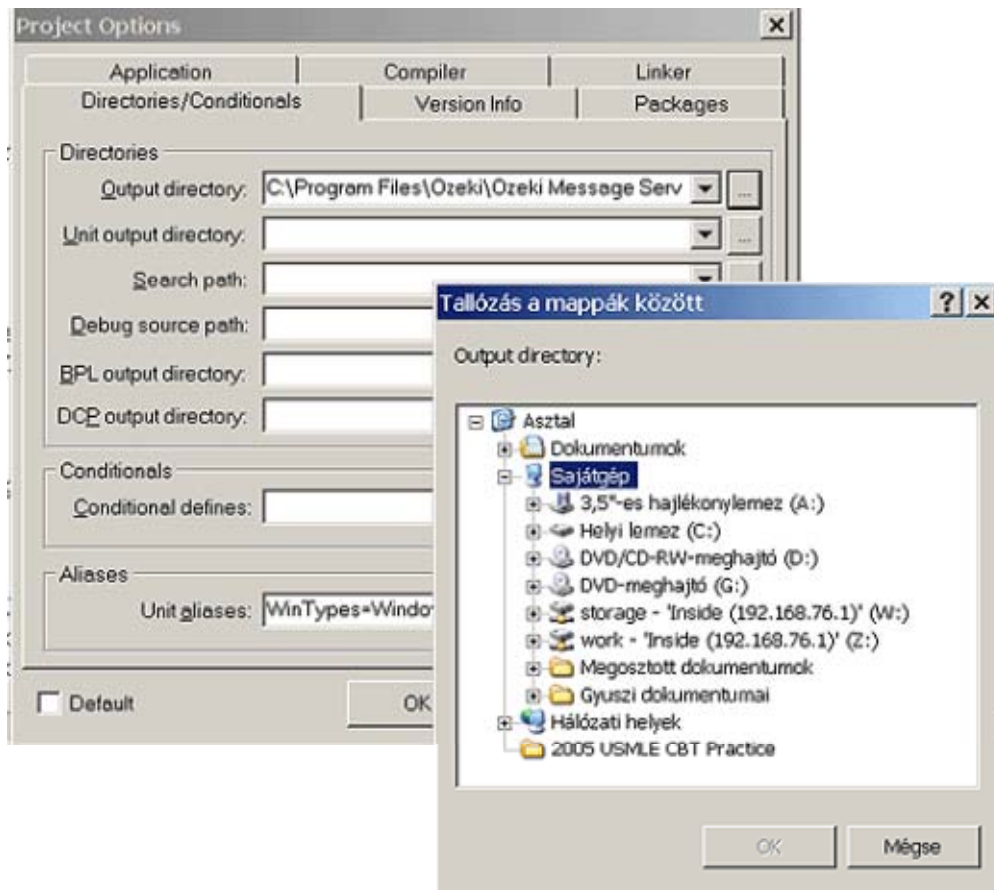


Figure 1 - Configuration of the output directory

Step 5:

Configure the run options in Delphi using the Run/Parameters menu item: Make sure the host application points to **C:\Program Files\Ozeki\Ozeki Message Server\MessageServer.exe** and the parameter is **/run**. This way the software will start in application mode, that helps debugging (Figure 2).

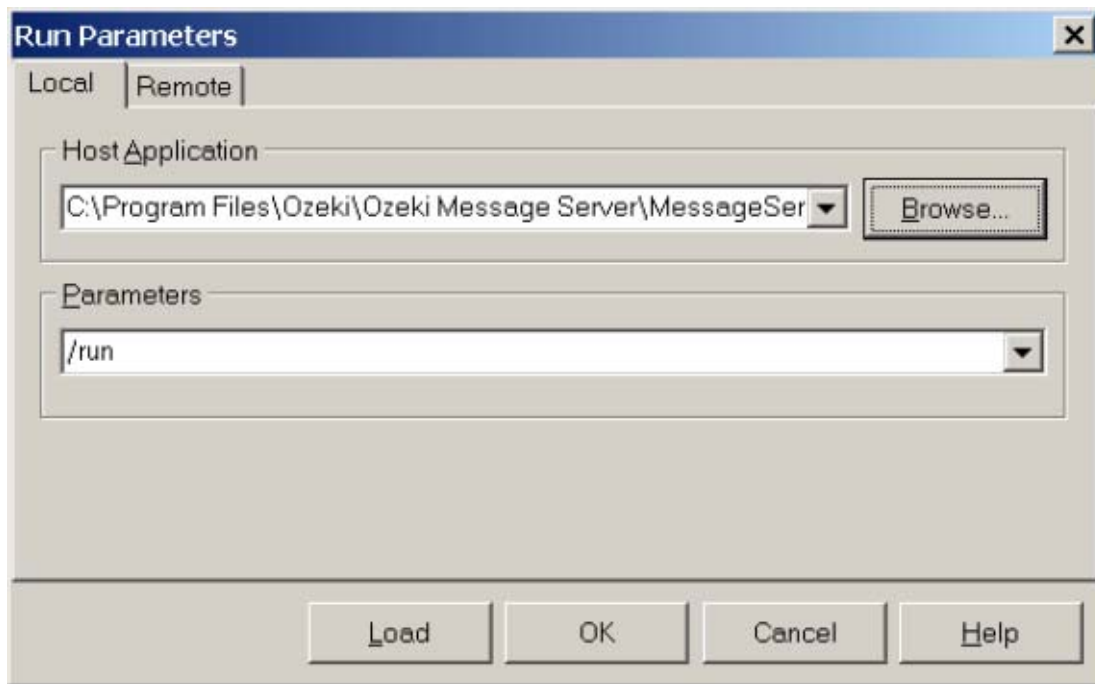


Figure 2 - Configuration of the run options

SMS API for Microsoft Access developers

In this step-by-step tutorial we explain how to create an Access application, which is capable of sending and receiving SMS messages. In order to use this tutorial, you need an installed copy of **OZEKI MESSAGE SERVER** and **MICROSOFT ACCESS**.



Step 1 - create an Access database

First you must have a database. You can give it any name. In this example we named it **ozeki_ms6_released**. In this database you need to create two tables. One of these tables is going to be used for sending SMS messages while the other is going to be used for receiving. It is important that the table used for sending messages is named **ozekimessageout** and the table used for receiving

messages is named **ozekimessagein**. The columns in the tables should be created as presented on the following screenshot. (*Figure 1.*) Make sure you set the field size of the msg column to 255. You can add additional columns to both tables, but you cannot change the name of the columns used for sms sending and receiving. You can download this database in **Access 2003** format from here: [ozeki_ms6_released.mdb](#). (174kb)

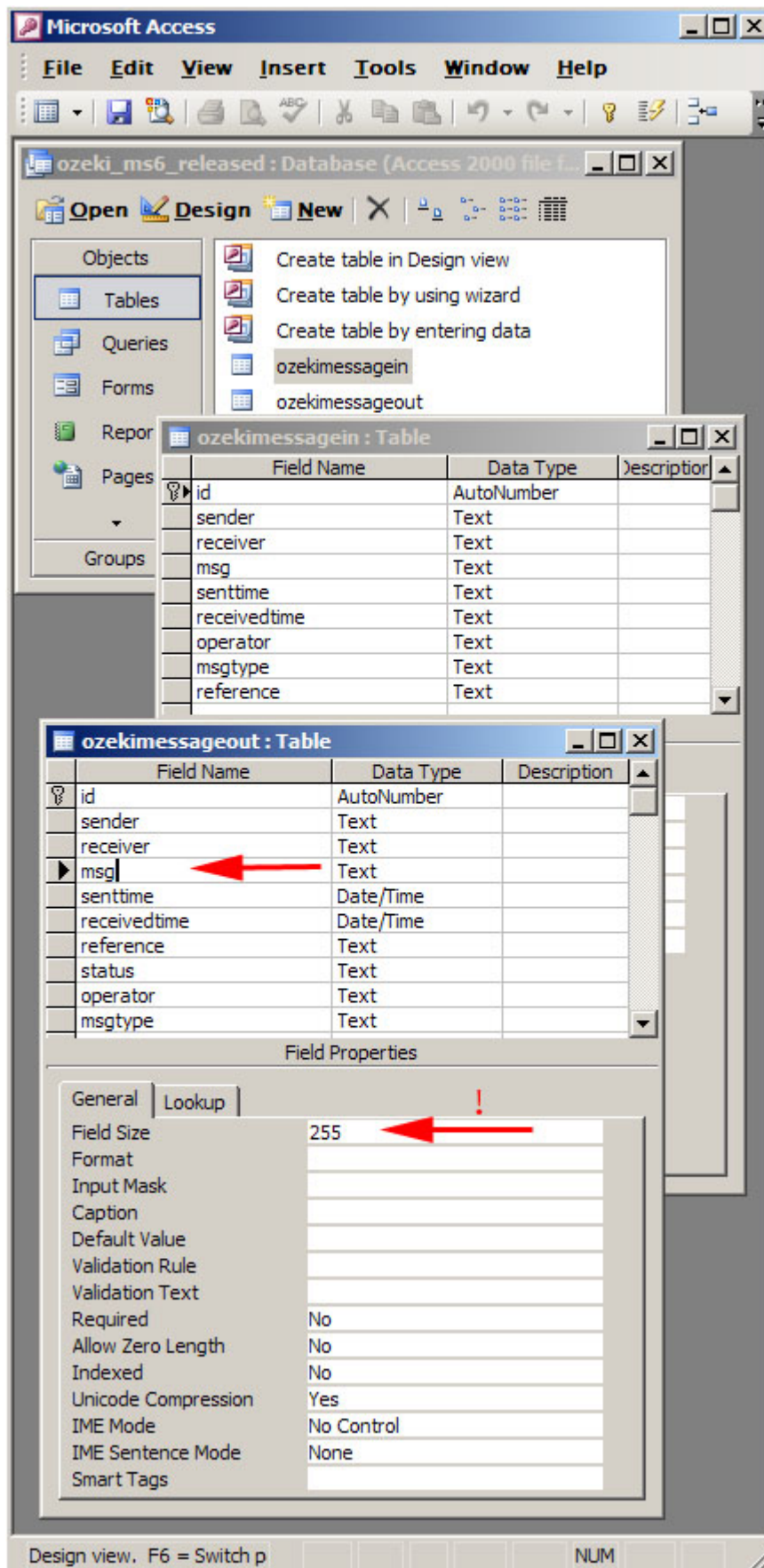


Figure 1 - Specifying of the Ozekimessagin and the Ozekimessageout table

Step 2 - Setting up windows ODBC driver

In order to use Access you need to set up an ODBC connection. Click on Start / Settings /Control Panel / **Administrative Tools** and click on the "Data Sources" menu item.(Figure 2) .

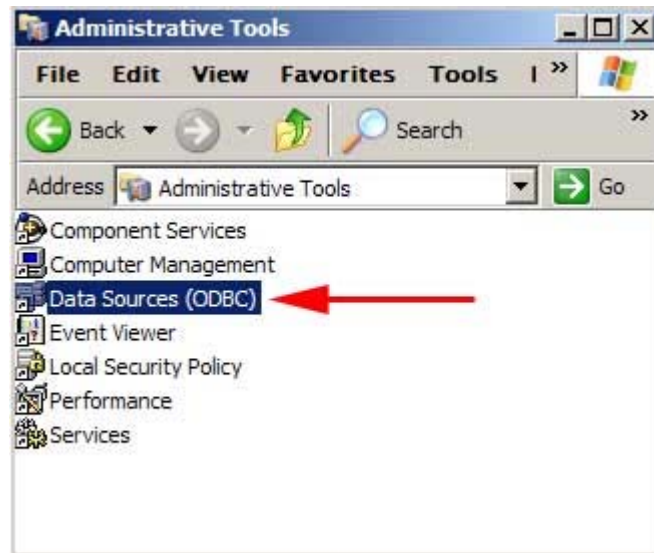


Figure 2 - Administrative Tools

The following window opens: **ODBC Data Source Administrator** (Figure 3.) In this window select the **System DSN** (1) tab. Then click on **Add** (2).

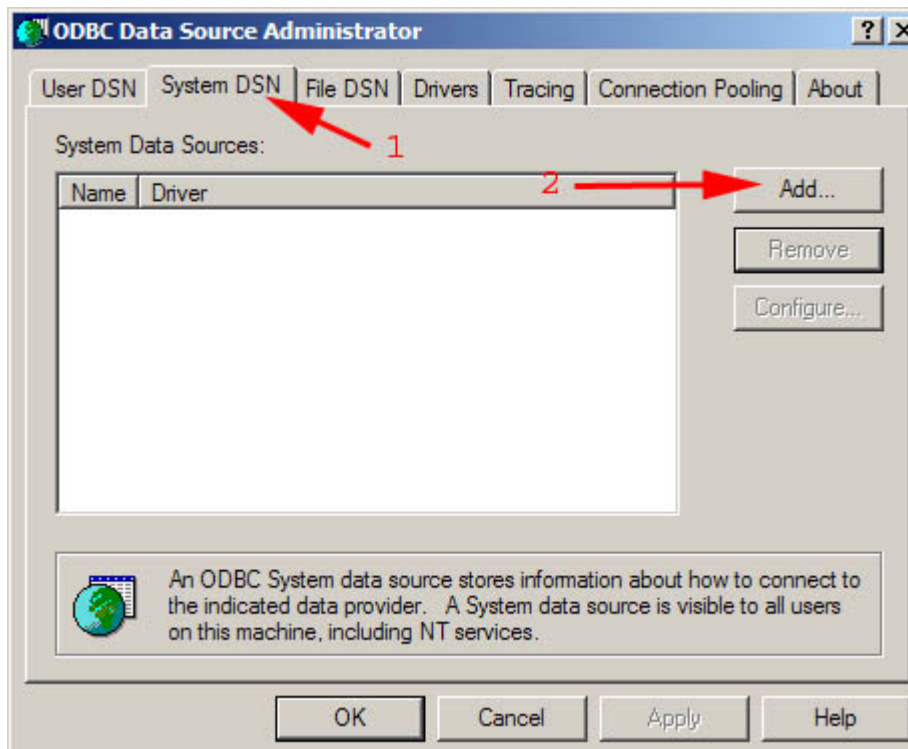


Figure 3 - System DSN

On the **Create New Data Source** form, select the **Microsoft Access Driver** (1) and click to **Finish** (2). (Figure 4.)

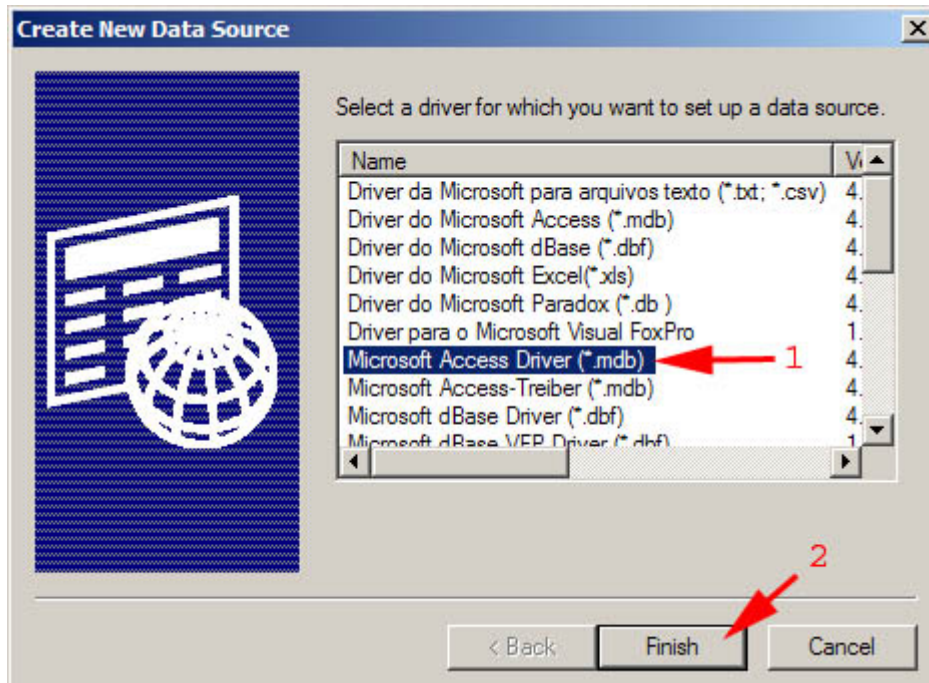


Figure 4 - Creation of the new data source

This brings up the **ODBC Microsoft Access Setup** (Figure 5.) form. Set the **Data Source Name** to MSAccessSystem (1) write some **Description** (2) Click on **Select** (3). Open the previously downloaded file, or the file you have created yourself: ozeki_ms6_released.mdb.

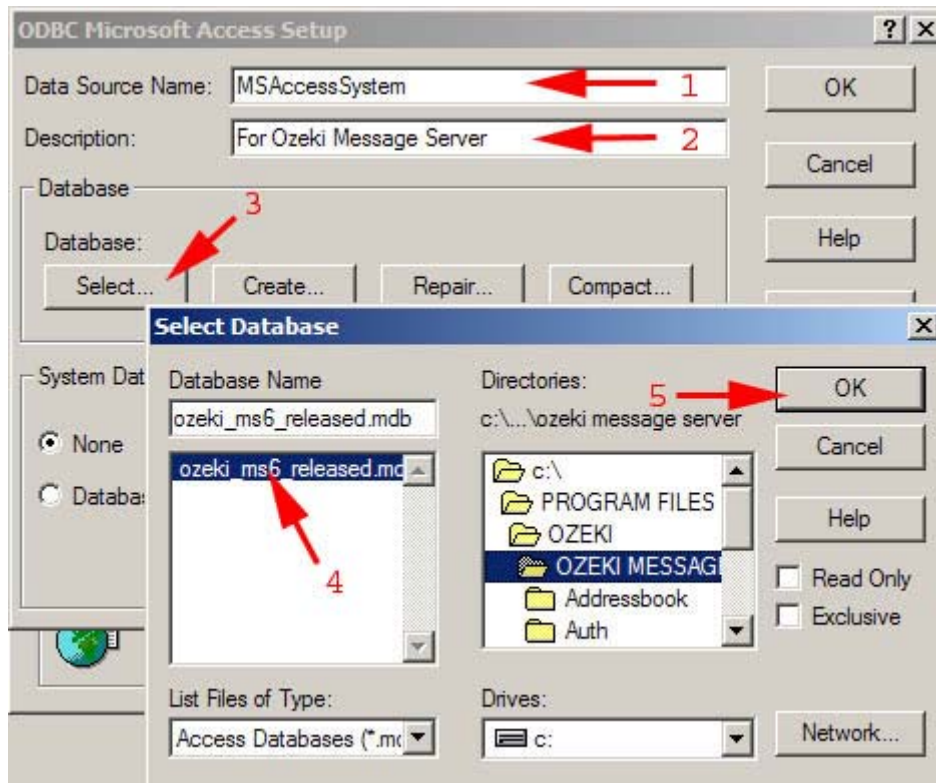


Figure 5 - Selecting database

Step 3 - Configure OZEKI MESSAGE SERVER to use database connection

This can be done by opening the **Plugins** menu. Select the **Installed Plugins** menuitem (Figure 6.)

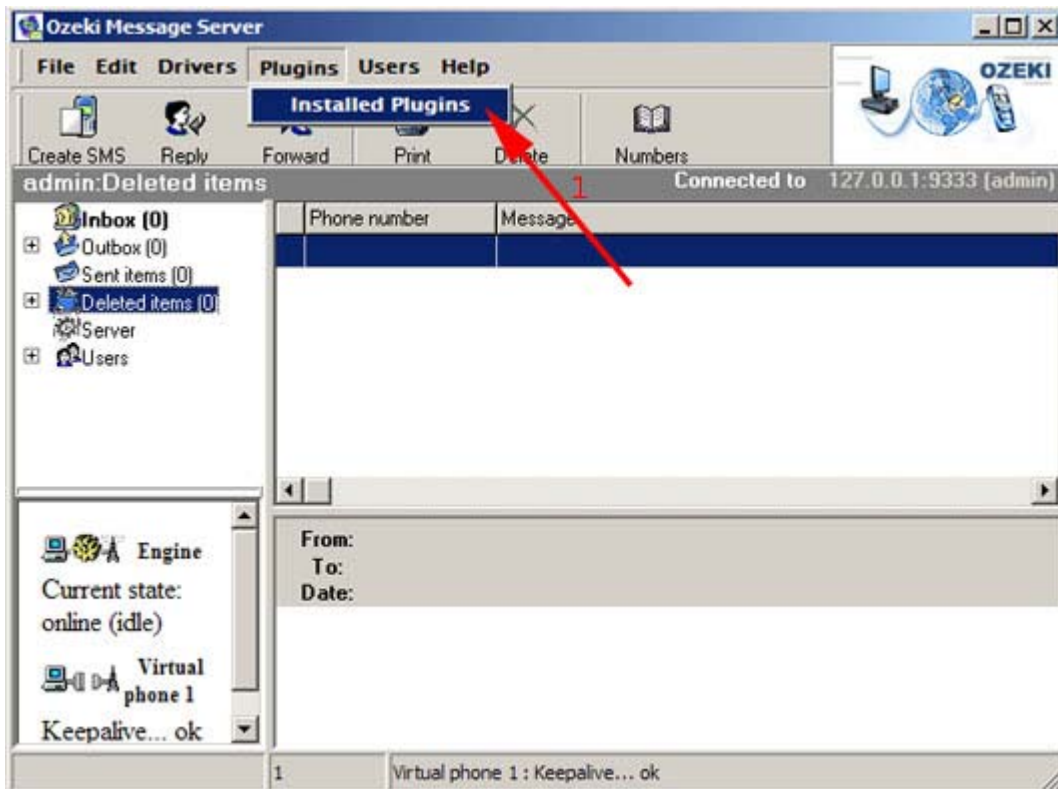


Figure 6 - Installation plugins

When the form is open, select the **database plugin** to **install**. (Figure 7 .)

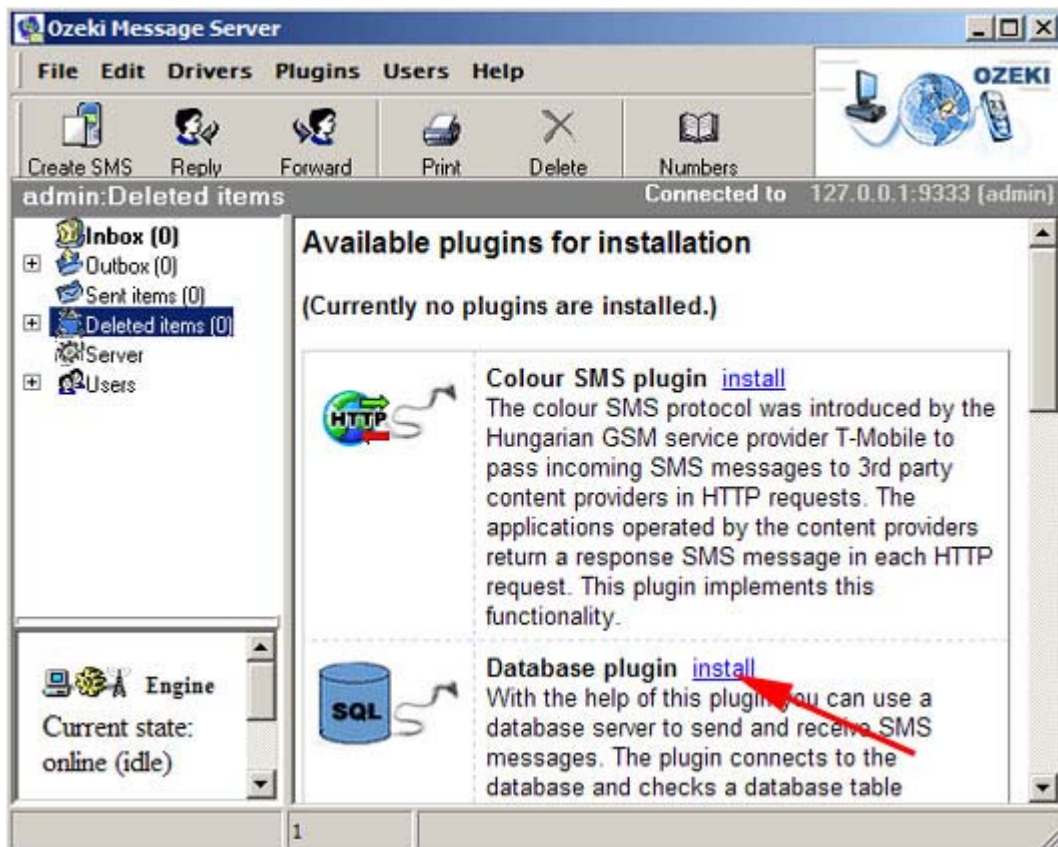


Figure 7 - Installation of the Database plugin

Next click on **Configure!** (Figure 8.)

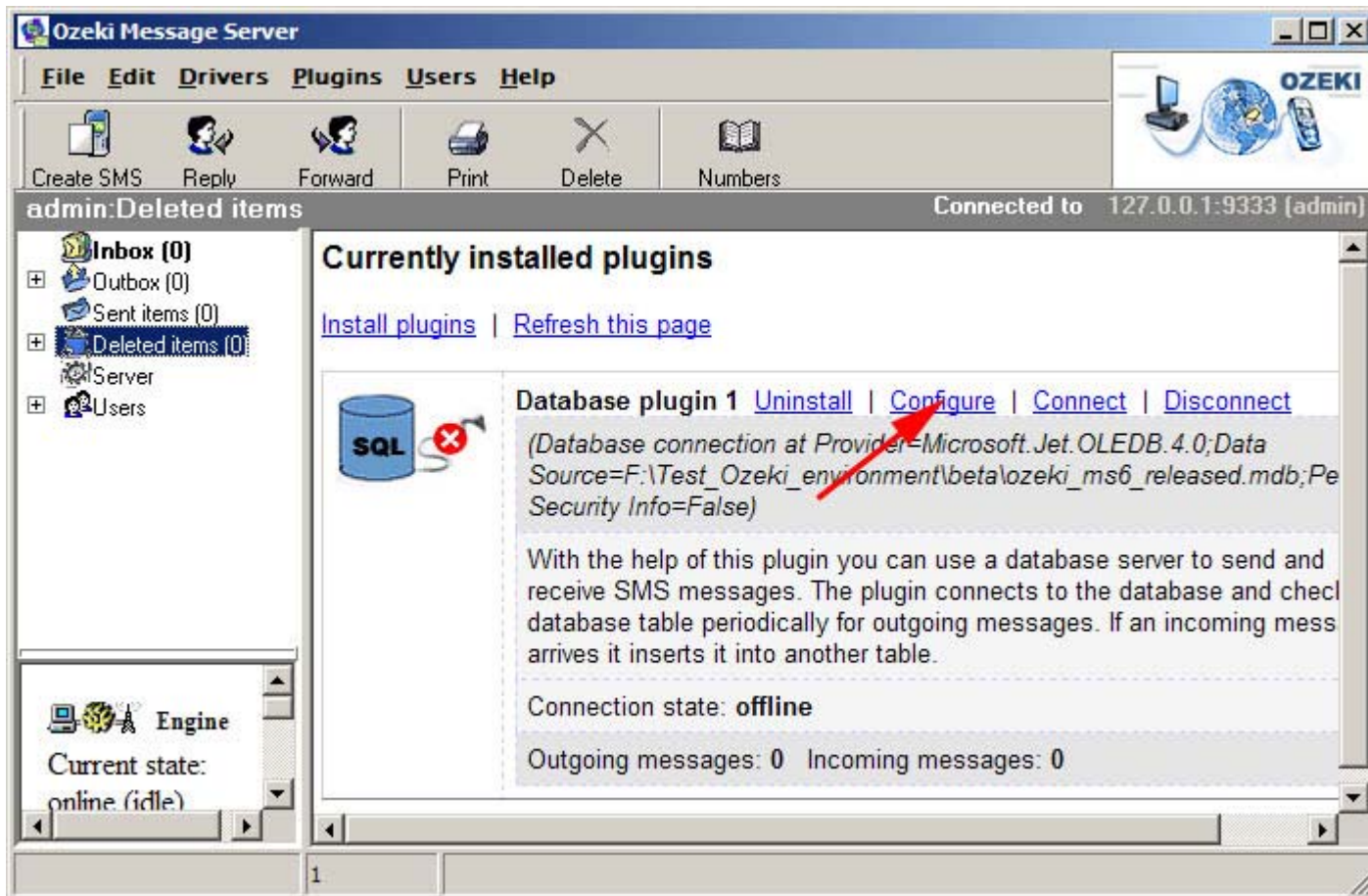


Figure 8 - Configuration of the Database plugin

If you want to use the database for sending and receiving messages place checkmarks into the **Use database for receiving SMS messages** and the **Use database for sending SMS messages checkbox** (2,3). Important: place a checkmark into the **Auto connect on startup** checkbox (4) as well, to make sure, when you re start the program the database connection will start up automatically. You can define a database connection by clicking on **Build database connection** (1) (Figure 9.)

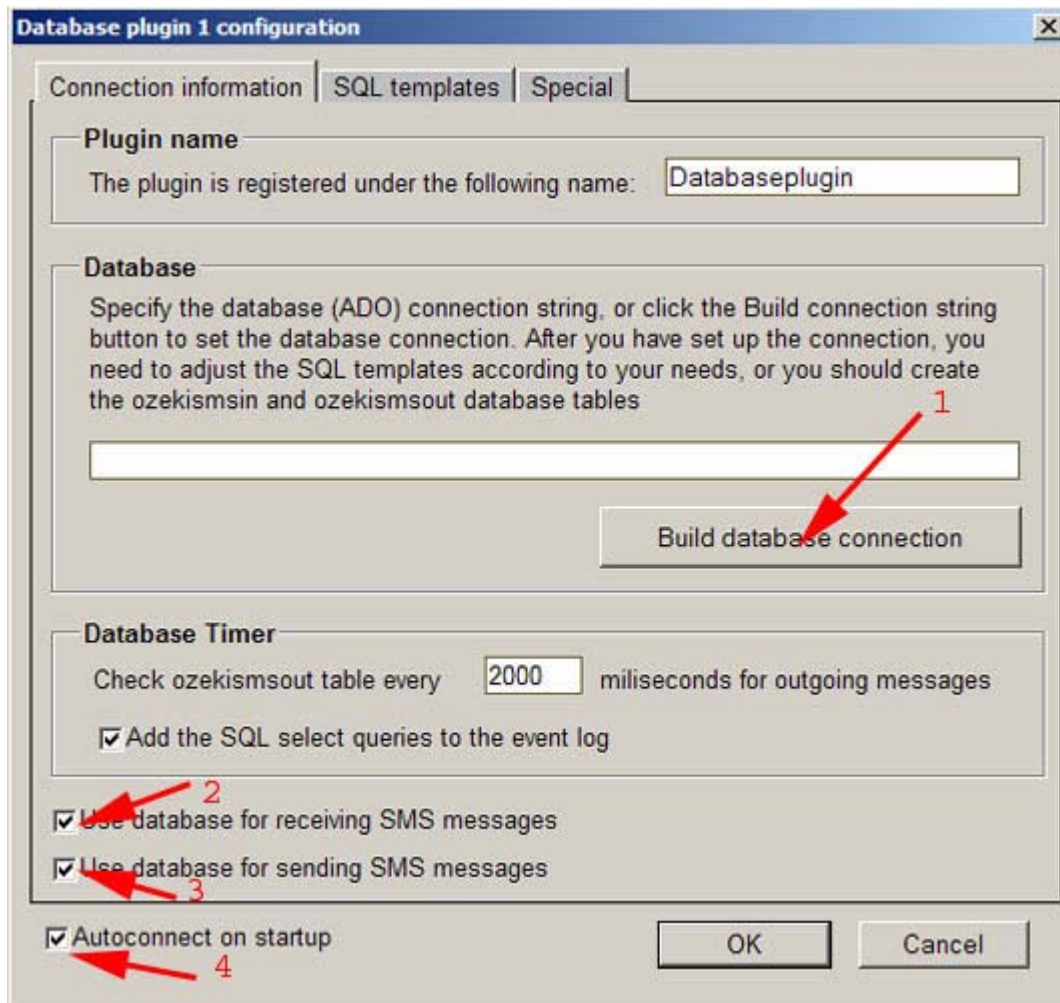


Figure 9 - Configuration in the Connection information tab

Select the **Use Connection String** radio button , and click on **build** (Figure 10.).

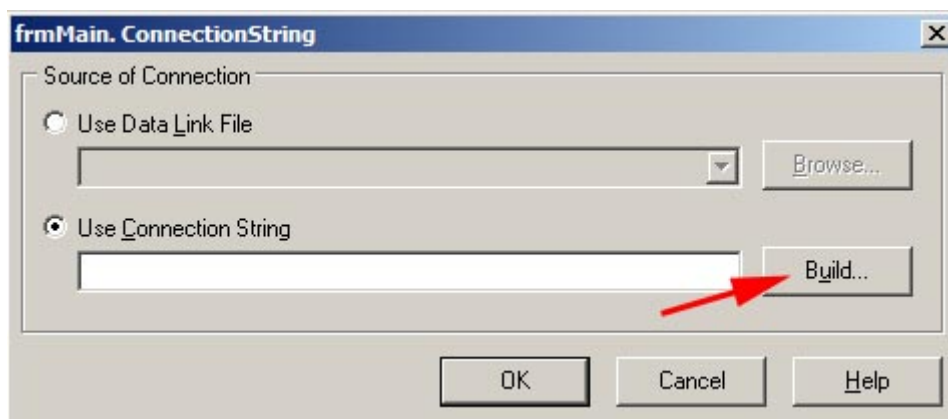


Figure 10 - Connection string

In the **Data Link Properties** window select the **Provider** tab and select the **OLE DB Provider for ODBC Drivers** lis item from the OLE DB Provider list box(1). Click on next (2).(Figure 11.)

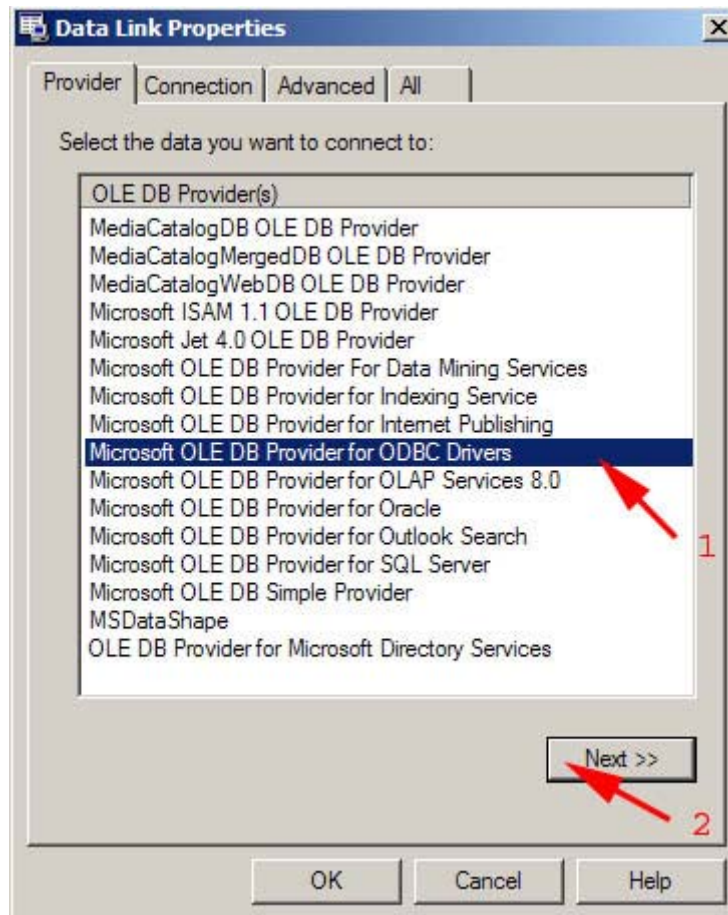


Figure 11 - Data link properties

The Connection tab will appear, Click the drop down menu called "Use data source name" and select **MSAccessSystem** list item (1,2). Remember in the second step we create the Data Source Name: MSAccessSystem (Figure 5). Make sure the **User name** is Admin (3). Click to **Blank password** (4) You should test the connection by clicking on **Test Connection** (5) (Figure 12.)

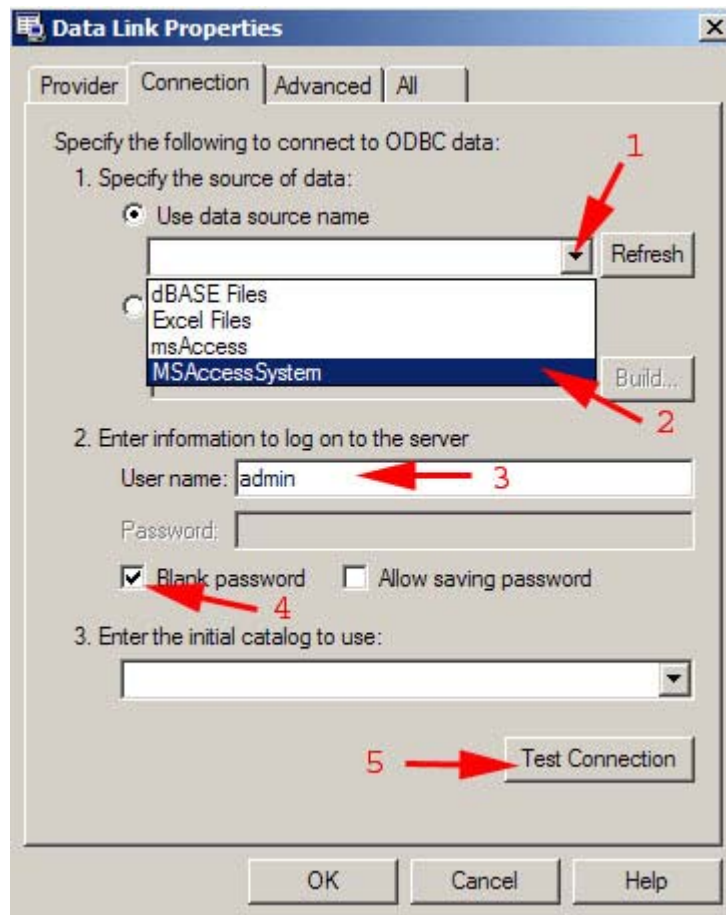


Figure 12 - Choosing of the data source

If the connection is ok you will see: (Figure 13.)



Figure 13- Connection is successfully

When you setup is correct the next window will appear: At the bottom of the window you can see the log if you click on the statusbar. When the Database connection is established the *Database plugin 1 : select id,receiver,msg,operator* text will appear in the log.

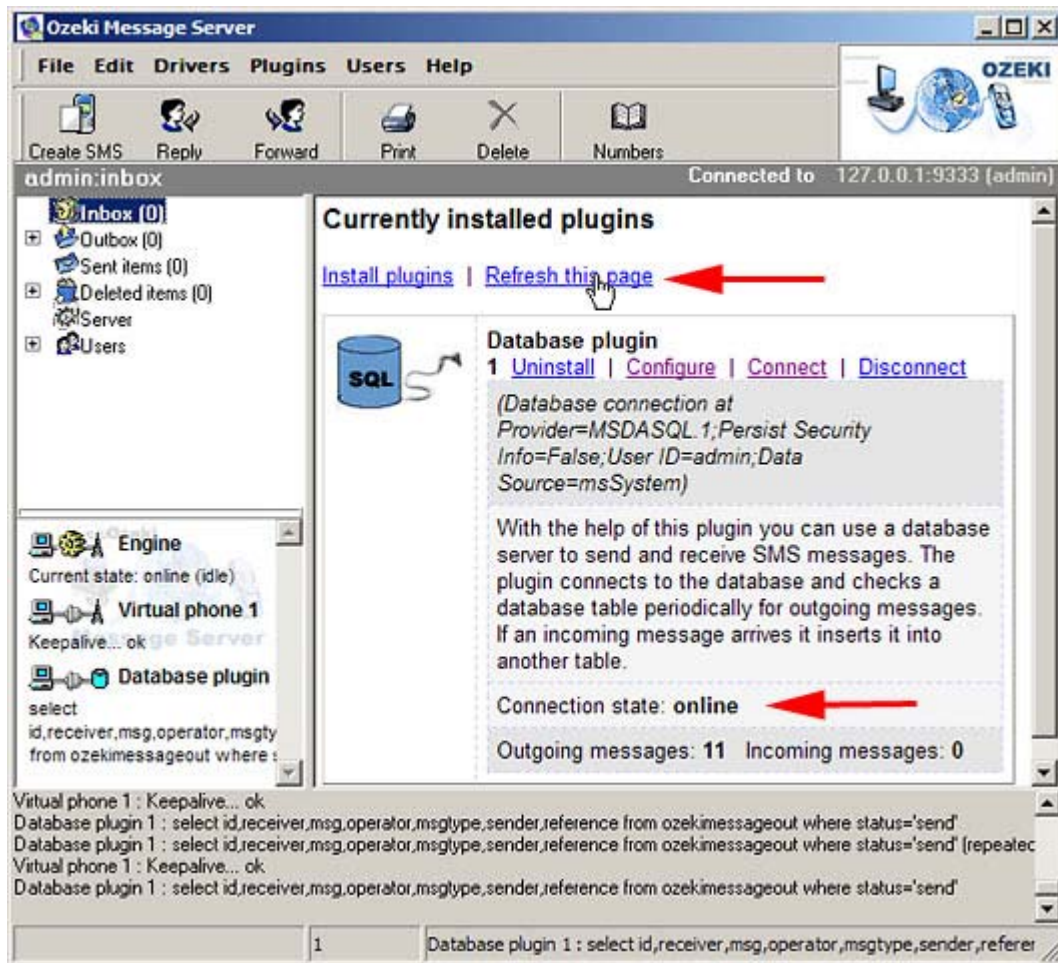


Figure 14 - Refreshing of the Currently installed plugins

Sending sms via MS Access

When the server starts, go to Access and send a message, by inserting a new record into the **ozekimessageout** table. You must specify the **receiver**, enter the message in the **msg** field and set the status to **send!** (The incoming SMS messages will appear in the **ozekimessagein** table.)

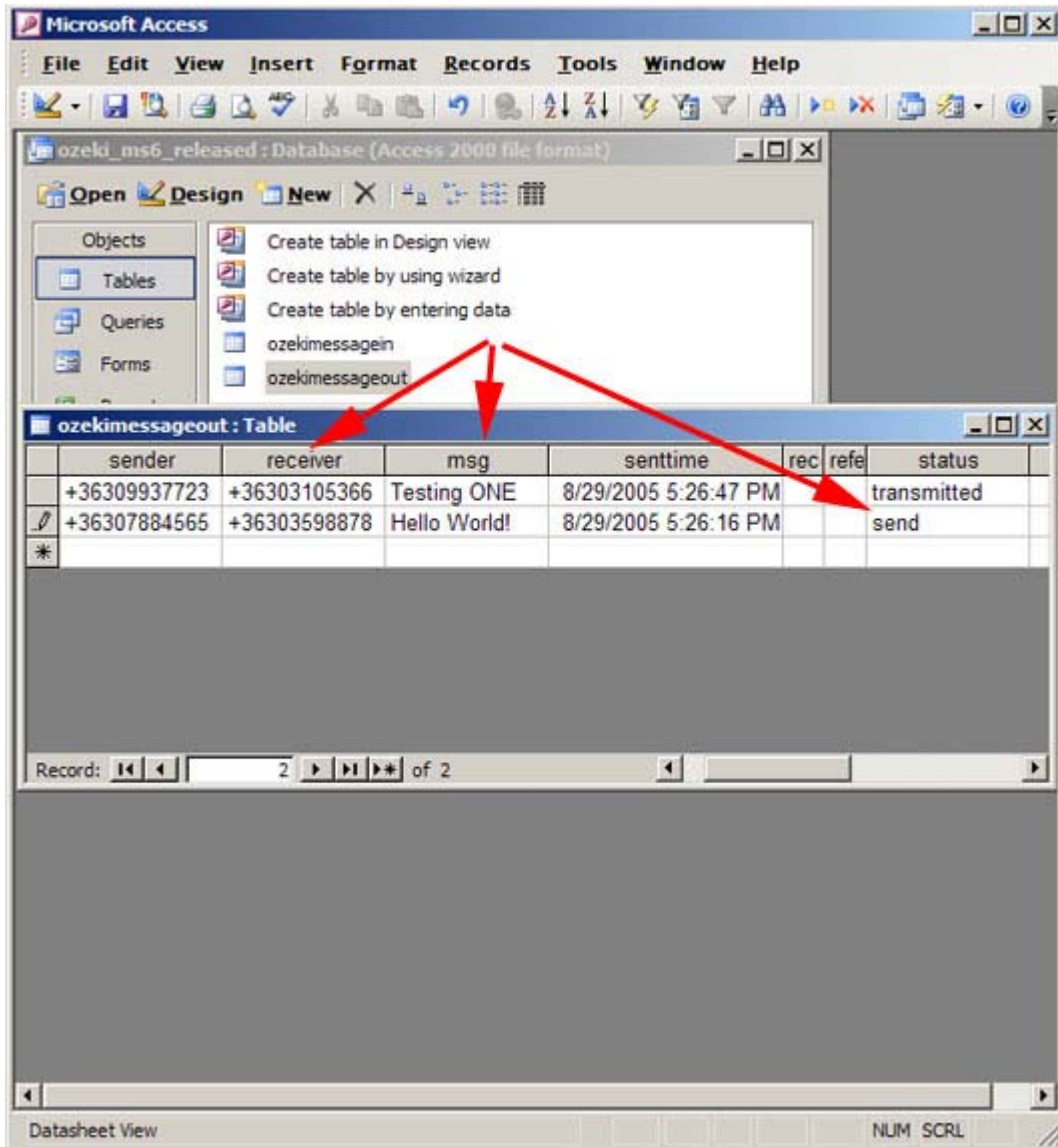


Figure 15 - Content of the Ozekimessageout table

OZEKI SMS allow you to control the **SQL queries** used to access the data in your **Access database** . In these versions, you must modify these queries to make things work. Here is what you should do:

Remove the apostrophe around the word \$id. ('\$id' should be changed to \$id)

Here is what you should get after modifying the SQL statements:

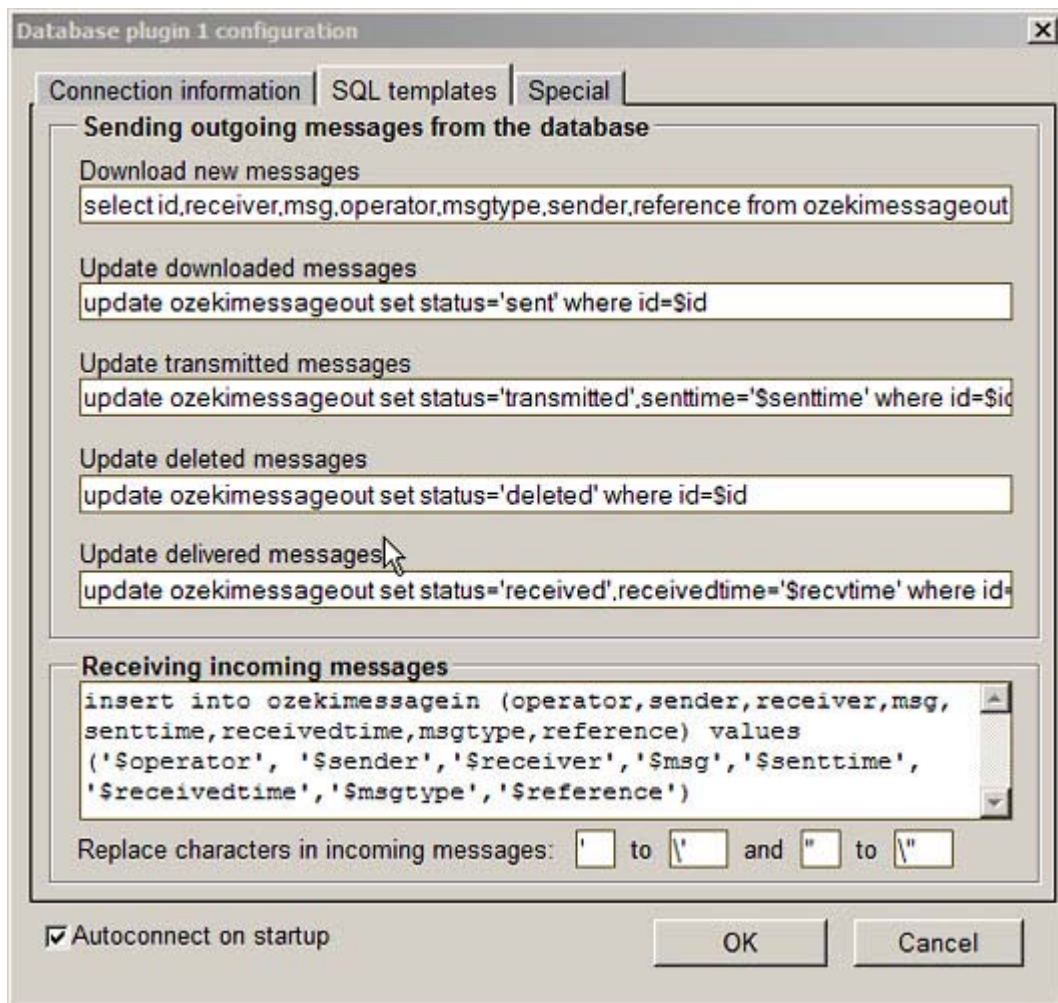


Figure 16 - Configuration in the SQL templates

Notice that the apostrophe around the id field is removed.

Appendix

A1 - SMS Message Types, OTA Bitamp format

A2 - SMS Message Types, Operator Logo XML format

A3 - SMS Message Types, Operator Logo format

A4 - SMS Message Types, Ringtone

A5 - SMS Message Types, WAP push

B - SMS Message Types, Service providers

C - SMS Message Types, SMSC numbers

D - SMS Message Types, GSM Operators

E - SMS Message Types, SMS alphabet

F - SMS Message Types, Error codes

G - SMS Message Types, ADO connection strings

H - SMS Message Types, References

Appendix "A1" - Message types - OTA bitmap

Introduction

An OTA bitmap is used for various SMS formats: Operator logo, CLI icon, Picture Message, and Downloadable Profile. In most phone the maximum size of the operator logo and the CLI icon is 72 x 14 pixels, while the maximum size of the picture message and the screen saver is 72 x 28 pixels. An OTA bitmap consists of a bitmap header and bitmap data. The size of the bitmap is specified in the header.

A typical OTA bitmap (72 x 14 pixels) header is: 00480E01

00	Infofield
48	Width of the bitmap is 72 pixels
0E	Height of the bitmap is 14 pixels
00	Number of colors or grey shades (only one color)

The image data is located after the header information and is encoded as follows. Each semi-octet in the OTA bitmap presents 4 pixels in the original bitmap. Because one row takes 18 semi-octets, the whole 72 x 14 (operator logo and CLI icon) bitmap takes $18 \times 14 = 252$ semi-octets = 126 octets. With picture message and screen saver, the entire 72 x 28 size bitmap takes $18 \times 28 = 504$ semi-octets = 252 octets. For example, if the first four pixels of the image are 1010 (1 - black, 0 - white), the first semi-octet of the OTA bitmap data is hex A.

Example

Here is an example of a simple OTA bitmap (72 x 14 pixels). In the picture, there are two black lines and several black dots:

```

FFFFFFFFFFFFFFFFFFFF <- First line black
000000000000000000 <- Second line white
FFFFFFFFFFFFFFFFFFFF
000000000000000000
10F000000000000000 <- Fourth pixel of this line is black and 9-12
000000000000000000 pixels are also black
000000000000000000
000000000000000000
000000000000000000
000000000000000000
000000000000000000
000000000000000000
000000000000000000
000000000000000000
000000000000000001 <- Last pixel of this row/bitmap is black

```


Appendix "A2" - Message types - Operator logo XML

Introduction

Operator logos are graphical images that are displayed on a phone when the phone signs into a GSM network. An operator logo is identified by a country code and a mobile network code. The GSM handsets supporting this technology can store more than one images at a time, and when the handset roams between networks, it can display the appropriate operator logo.

An operator logo is an OTA bitmap. The OTA bitmap is a black and white picture with a gray scale level of 1 to 255. OTA bitmap has a header and a body. The header has a predefined width and a height field, that specifies, how the stream of bits in the body should be broken into lines.

When designing operator logos, the designer must keep in mind, that different handset support operator logos with different dimensions. The most common bitmap size for operator logos is 72x14 pixels.

Support for Operator Logos in Ozeki Message Server 6

Because an operator logo has several parameters in addition to the bitmap stream, Ozeki uses an XML format to represent it. The following example illustrates the structure of this XML file:

```
<operatorlogo>
<version>0</version>
<mcc>123</mcc>
<mnc>45</mnc>
<otabitmap>
00480E00
FFFFFFFFFFFFFFFF FF00000000000000
0000FFFFFFFFFFFF FFFFFFF000000000
0000000010F00000 0000000000000000
0000000000000000 0000000000000000
0000000000000000 0000000000000000
0000000000000000 0000000000000000
0000000000000000 0000000000000000
0000000000000000 0000000000000000
0000000000000000 0000000000000001
</otabitmap>
</operatorlogo>
```

version	The operator logo version number
mcc	Mobile Country Code
mnc	Mobile Network Code
otabitmap	OTA bitmap data (The bit stream containing the data. Note: ozeki message server 6 will remove any spaces from the hexadecimal character string)

Sending an operator logo in SMS

Appendix "A3" - Message types - Operator logo

Introduction

Operator logos are graphical images that are displayed on a phone when the phone signs into a GSM network. An operator logo is identified by a country code and a mobile network code. The GSM handsets supporting this technology can store more than one images at a time, and when the handset roams between networks, it can display the appropriate operator logo.

An operator logo is an OTA bitmap. The OTA bitmap is a black and white picture with a gray scale level of 1 to 255. OTA bitmap has a header and a body. The header has a predefined width and a height field, that specifies, how the stream of bits in the body should be broken into lines.

When designing operator logos, the designer must keep in mind, that different handset support operator logos with different dimensions. The most common bitmap size for operator logos is 72x14 pixels.

Support for operator logos in Ozeki Message Server 6

To build an operator logo it must be encoded in a binary format. This format contains a header and a body. The header specifies the operator logo version number, the Mobile Country Code (MCC) and the Mobile Network Code (MNC) fields in addition to the dimensions and the color depth of the bitmap. The body holds the OTA bitmap.

Example:

```
3021F3540A00480E01FFFFFFFFFFFFFFFFFFFFFFFF00000000000000
000000FFFFFFFFFFFFFFFFFFFFFFFF00000000000000000000000010F000
000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000
```

Explanation:

30	Operator logo version number. ISO-8859-1 character "0"
21 F3	Mobile Country Code (MCC), octets 14 and 15, little-endian BCD, filled with F16, 123 -> 21 F3
54	Mobile Network Code (MNC) coding, little-endian BCD, filled with F16, 45 -> 54
0A	ISO-8859-1 "Line feed" character
00	InfoField
48	The width of the bitmap. Hex 48 -> 72 decimal
0E	The height of the bitmap. Hex 0E -> 14 decimal
01	The depth of the bitmap (number of gray scales)
FFFFFFFFFFFFFFFF	OTA bitmap data

Appendix "A4" - Message types - Ringtone

Introduction

Ringtones are little melodies and tunes created to give a unique sound to the GSM phone when the phone is ringing. Ringtones can be composed with an appropriate editor and they can be sent as SMS messages.

Support for operator logos in Ozeki Message Server 6

When a ringtone is sent as an SMS message it is represented as a bitstream. This bitstream can be pictured as a set of hexadecimal numbers. For example:

```
024A3A51D195CDD004001B20550590610560558550548540820849900000
```

The bit string contains the following sections:

00000010	<command-length> Number of command parts present
01001010	<ringing-tone-programming> Command part 1 (with filler bit)
0011101	<sound> Command part 2
001	<basic song type>
0100	<song title length> 4 characters (ISO-8859-1)
01110100	the first character T
01100101	the second character E
01110011	the third character S
01110100	the fourth character T
00000001	<song sequence length> 1 song pattern
000	<pattern header> pattern header ID
00	<pattern id> A-part
0000	<loop value> no loop
00001101	<pattern specifier> <length of the new pattern> 13 pattern instructions
100	<tempo instruction id>
10000	<beats per minute> 160 (i.e., length of 1/4 note = 0,38 sec.)
001	<note instruction id>
0101	<note value> note E
010	<note duration> L' note
00	<note duration specifier> no special duration
001	<note instruction id>
0110	<note value> note F
010	<note duration> L' note
00	<note duration specifier> no special duration
001	<note instruction id>
1000	<note instruction> note G
010	<note duration> L' note
00	<note duration specifier> no special duration
001	<note instruction id>

0101	<note value> note E
100	<note duration> 1/16 note
00	<note duration specifier> no special duration
001	<note instruction id>
0101	<note value> note E
011	<note duration> 1/8 note
00	<note duration specifier> no special duration
001	<note instruction id>
0101	<note value> note E
010	<note duration> L' note
00	<note duration specifier> no special duration
001	<note instruction id>
0101	<note value> note E
001	<note duration> ~ note
00	<note duration specifier> no special duration
001	<note instruction id>
0101	<note value> note E
000	<note duration> full note
00	<note duration specifier> no special duration
010	<scale instruction id>
00	<note scale> Scale-1 (i.e., note A is 440 Hz)
001	<note instruction id>
0000	<note value> pause
010	<note duration> L' note
00	<note duration specifier> no special duration
010	<scale instruction id>
01	<note scale> Scale-2 (i.e., note A is 880 Hz), default
001	<note instruction id>
1001	<note value> Gis ' i.e. As '
000	<note duration> full note
00	<note duration specifier> no special duration
0000000	filler bits
00000000	<command end> end of the ringing tone data

Sending a ringtone in SMS

To send a ringtone message with Ozeki Message Server 6, you need to specify a message type, that tells the software that the message body contains a ringtone. To send the above example, you would use the following message type: SMS:RINGTONE In this case, the message type is made up of two sections. The first section tells the message server that this is an SMS message. This way only the drivers supporting SMS will be considered to deliver the message. The second section is used by the driver to use the ringtone encoder. This encoder will create the protocol data unit (PDU) that will be sent as an SMS messages.

Example:

If you use the database plugin, to send and receive SMS messages in Ozeki Message Server 6, you can use the following SQL statement to send a ringtone message: insert into ozekimessageout (receiver,msgtype,msg,status) values ('+36205222245', 'SMS:RINGTONE','024A3A51D195CDD004001B20550590610560558550548540820849900000','send');

Appendix "A5" - Message types - Wap push

Introduction

The WAP Push technology makes it possible to send a specially encoded SMS message with a description and link to a WAP address (URL). When a GSM handset receives this WAP Push message, it will automatically give the user the option to access the WAP content. The WAP content can be WML page, a JAVA application or multimedia content.

WAP Push is also used to send polyphonic ringtones and wallpaper images to phones.

To instruct a phone to go to a certain WAP location, a Service Indication message needs to be used. This is an XML document that contains a URL for the handset to browse to. For example:

```
<si>
< indication href="http://wap.origo.hu/" action="signal-high">
  A WAP Push to the Origo site
< /indication>
</si>
```

Sending a WAP Push message in SMS

To send a WAP PUSH message with Ozeki Message Server 6, you need to specify a message type containing a MIME identification. To send the above example, you would use the following message type:

```
SMS: WAPPUSH: application/vnd.wap.sic
```

In this case, the message type is made up from three sections. The first section tells the message server that this is an SMS message. This way only the drivers supporting SMS will be considered to deliver the message. The second section is used by the driver to use the WAP PUSH SMS encoder. This encoder will use the application/vnd.wap.sic MIME type to build the Wireless Session Protocol (WSP) pdu encoded in WBXML format, and will use the Wireless Datagram Protocol (WDP) to form the SMS messages.

After the message type has been set, the message content should be created using the appropriate DTD. For example for a WAP PUSH service indication message you should use the following DTD.

Service Indication DTD

Terms and conditions of use are available from the Wireless Application Protocol Forum Ltd. web site at

<http://www.wapforum.org/docs/copyright.htm>.

-->


```

<!ENTITY % Datetime "CDATA">          <!-- ISO date and time -->
<!ENTITY % URI      "CDATA">          <!-- URI designating a hypertext node
-->
<!--===== The SI Element
----->
<!ELEMENT si (indication,info?)>
<!--===== The Indication Element
----->
<!ELEMENT indication (#PCDATA)>
<!ATTLIST indication
> href          %URI;                #IMPLIED
> si-id         CDATA                 #IMPLIED
> created       %Datetime;           #IMPLIED
> si-expires    %Datetime;           #IMPLIED
> action        (signal-none|signal-low|
>               signal-medium|signal-high|delete) "signal-medium"
>
<!--===== The INFO Element
----->
<!ELEMENT info (item+)>
<!ELEMENT item (#PCDATA)>
<!ATTLIST item
> class         NMTOKEN              #REQUIRED
>

```

Example

If you use the database plugin, to send and receive SMS messages in Ozeki Message Server 6, you can use the following SQL statement to send a WAP PUSH Service Indication message:

```

insert into ozekimessageout (receiver,msgtype,msg,status) values
('+36205222245', 'SMS:WAPPUSH:application/vnd.wap.sic', '<si><indication
href="http://wap.origo.hu/" action="signal-high">A WAP Push to the Origo
site</indication></si>', 'send');

```

Appendix "B" - GSM service providers

Albania +35538 - Albanian Mobile Comms	Algeria +2131 - PTT Algeria
Andorra +37632 - S.T.A. +37632 - STA	Australia +61404 - Vodafone +61410 - Vodafone +61414 - Vodafone Australia +61415 - Vodafone Australia +61416 - Vodafone Australia +61640 - OneTel
Austria +43650 - Teling +43664 - mobilkom +43676 - Max.Mobil +43699 - Connect Austria	Azerbaijan +99450 - AzerCell +99455 - JV BAKCELL
Bahrain +97390 - Batelco +97391 - Batelco +97392 - BATELCO +97394 - Batelco +97396 - Batelco +97398 - BATELCO	Belgium +32485 - KPN ORANGE BELGIUM NV +32486 - KPN ORANGE BELGIUM NV +32494 - Mobistar +32495 - Mobistar +32496 - Mobistar +32497 - Mobistar +32498 - MOBISTAR
Bosnia and Herzegovina +387387 - Eronet +38766 - GSM BIH	
Canada +1204 - Microcell Telecom Inc	Botswana +26771 - Mascom Wireless +26772 - Vista Cellular (Pty) Limited
China +86 - China Mobile Communication Corp. +86130 - China Unicom +86135 - China Telecom +86136 - China Telecom +86137 - China Telecom +86138 - China Telecom +86139 - China Telecom China Telecom +86 13800210500 China Telecom +86 13800210500 China Unicom +86 13010314500	Chile +56980 - ENTEL PCS +56982 - ENTEL PCS +56987 - ENTEL PCS +56988 - ENTEL PCS
Croatia +38591 - Vip-Net +38598 - Croatian Telecom	Cote D'Ivoire +22508 - S.I.M. +22509 - SIM IVOIRIS +2255 - Loteny Telecom (SA) +22581 - SIM IVOIRIS +22582 - SIM IVOIRIS +22590 - SIM IVOIRIS +22593 - SIM IVOIRIS +22598 - SIM IVOIRIS
Czech Republic +420601 - Eurotel +420602 - Eurotel +420603 - RadioMobil +420604 - RadioMobil +420605 - RADIOMOBIL	Cyprus +3579 - CY.T.A
	Egypt

+420606 - Eurotel	+2010 - Misrfone
+420607 - Eurotel	Telecommunications Co.
+42072 - EURO TEL PRAHA	
+42073 - RADIOMOBIL	Equatorial Guinea
+42093 - RADIOMOBIL	+2407 - GETESA
+420962 - EURO TEL PRAHA	
+420965 - RADIOMOBIL	Estonia
+420966 - EURO TEL PRAHA	+37256 - RadioLinja Eesti AS
Denmark	F.Y.R Macedonia
+45201 - Tele Danmark Mobil	+389 - Makedonski
+45202 - Tele Danmark Mobil	Telekomunikacii AD
+45203 - Tele Danmark Mobil	
+45204 - Tele Danmark Mobil	Georgia
+45205 - TELE DENMARK	+99577 - GeoCell
+45206 - Sonofon	+99599 - Magticom
+45207 - Sonofon	
+45208 - Sonofon	Germany
+45209 - Sonofon	+49160 - T-Mobil
+4521 - TELE DENMARK	+49162 - Mannesmann
+45221 - SONOFON	+49170 - T-Mobil
+45222 - SONOFON	+49171 - T-Mobil
+45223 - SONOFON	+49172 - Mannesmann
+45224 - SONOFON	+49173 - Mannesmann
+45225 - SONOFON	+49174 - Mannesmann
+45226 - SONOFON	+49175 - T-Mobil
+45227 - SONOFON	
+45228 - SONOFON	Ghana
+45229 - SONOFON	+23324 - ScanCom Ltd
+4523 - TELE DENMARK	
+4524 - TELE DENMARK	Gibraltar
+4529 - TELE DENMARK	+35056 - Gibraltar Telecoms Intl
+45401 - Tele Danmark Mobil	+35057 - Gibtel
+45402 - Tele Danmark Mobil	+35058 - Gibtel
+45403 - Tele Danmark Mobil	
+45404 - Tele Danmark Mobil	Hong Kong
+45405 - Sonofon	+852601 - SmarTone Mobile
+45406 - Sonofon	Comms Ltd
+45407 - Sonofon	+852609 - Peoples Telephone
+45408 - Sonofon	Co Ltd
+45409 - Sonofon	+852901 - SmarTone Mobile
+45721 - SONOFON	Comms Ltd
+45722 - SONOFON	+852902 - Cable & Wireless
	HKT Ltd
Gabon	+852902 - HK Telecom CSL
+241 308009 - Telecel Gabon	+852903 - HK Telecom CSL
	+852904 - Hutchinson
Greece	Telephone
+3093 - Telestet	+852909 - HK Telecom CSL
+3097 - Cosmote	+852910 - HK Telecom CSL
	+852912 - HUTCHISON
Hungary	+852913 - SmarTone Mobile
+3620 - Pannon	Comms Ltd
+3630 - Westel 900 GSM	+852915 - HK Telecom CSL
+3670 - Vodafone	+852916 - SmarTone Mobile

Iceland	Comms Ltd
+35469 - TAL LTD.	+852917 - SmarTone Mobile
	Comms Ltd
	+852918 - HK Telecom CSL
India	+852919 - HK Telecom CSL
+919811 - STERLING CELLULAR LTD	+852920 - Peoples Telephone Co Ltd
+919812 - Escotel (UP West)	+852922 - SmarTone Mobile
+919814 - SPICE PUNJAB	Comms Ltd
+919820 - HUTCHISON MAX TELECOM LTD	+852924 - SmarTone Mobile
+919821 - BPL MOBILE COMMUNICATIONS LTD	Comms Ltd
+919822 - BIRLA Maharastra	+852925 - New World PCS Ltd
+919823 - BPL MOBILE COMMUNICATIONS LTD	+852926 - HK Telecom CSL
+919824 - BIRLA Gujarat	+852928 - Hutchinson Telephone
+919830 - USHA MARTIN TELEKOM	+852930 - HK Telecom CSL
+919837 - Escotel	+852931 - SmarTone Mobile
+919840 - Skycell	Comms Ltd
+919841 - RPG CELLULAR	+852934 - Peoples Telephone Co Ltd
+9198440 - SPICE KARNATAKA	+852938 - SmarTone Mobile
+9198450 - Airtel Karnataka	Comms Ltd
+919846 - BPL Mobile	+852940 - HK Telecom CSL
+919847 - Escotel	+852943 - SmarTone Mobile
+919849 - Bharti Cellular	Comms Ltd
	+852944 - SmarTone Mobile
	Comms Ltd
	+852945 - SmarTone Mobile
Indonesia	Comms Ltd
+62811 - TelkomSel	+852946 - HK Telecom CSL
+62812 - TelkomSel	+852947 - Hutchinson Telephone
+62815 - Satelindo	+852948 - Hutchinson Telephone
+62816 - Satelindo	+852949 - Hutchinson Telephone
+62817 - Excelcomindo	+852951 - Peoples Telephone Co Ltd
+62818 - Excelcomindo	+852955 - NEW WORLD PCS LIMITED
Ireland	+852958 - Peoples Telephone Co Ltd
+35385 - Meteor Mobile	+852960 - Peoples Telephone Co Ltd
+35386 - Esat Digifone	+852961 - HK Telecom CSL
	+852962 - NEW WORLD PCS LIMITED
Israel	+852965 - NEW WORLD PCS LIMITED
+972 - Partner Communications Company Ltd	+852966 - SmarTone Mobile
+97254 - Orange Israel	Comms Ltd
+97255 - Partner orange	+852968 - SmarTone Mobile
+97259 - PALESTINE TELECOMMUNICATIONS CO. P.L.C	Comms Ltd
	+852971 - Peoples Telephone Co Ltd
Italy	
+39320 - WIND	
TELECOMUNICAZIONI S.P.A.	
+39328 - WIND	
TELECOMUNICAZIONI S.P.A.	
+39329 - WIND	

TELECOMUNICAZIONI S.P.A.	+852972 - SmarTone Mobile
+39330 - Telecom Italia Mobile	Comms Ltd
+39333 - Telecom Italia Mobile	+852973 - SmarTone Mobile
+39334 - Telecom Italia Mobile	Comms Ltd
+39335 - Telecom Italia Mobile	+852974 - HUTCHISON
+39336 - Telecom Italia Mobile	+852975 - HUTCHISON
+39337 - Telecom Italia Mobile	+852976 - HK Telecom CSL
+39338 - Telecom Italia Mobile	+852978 - NEW WORLD PCS
+39339 - Telecom Italia Mobile	LIMITED
+39340 - OMNITEL PRONTO	+852984 - SmarTone Mobile
ITALIA	Comms Ltd
+39347 - Omnitel Pronto Italia	+852985 - SmarTone Mobile
+39348 - Omnitel Pronto Italia	Comms Ltd
+39349 - Omnitel Pronto Italia	+852986 - SmarTone Mobile
+39360 - Telecom Italia Mobile	Comms Ltd
+39368 - Telecom Italia Mobile	+852987 - NEW WORLD PCS
+39380 - BLU	LIMITED
+39388 - BLU	+852988 - HK Telecom CSL
+39389 - BLU	
	Jordan
Lesotho	+962 - Jordan M.T.S.
+266 - VCL	+96277 - Mobilecom
	+96279 - FASTLINK
Libyan Arab Jamahiriya	
+91 - EL MADAR TEL	Kazakhstan
	+7 - GSM Kazakhstan JSC
Liechtenstein	
+423 - TANGO	Kenya
+4236979 -	+25472 - Safaricom
OBILKOM(LIECHTENSTEIN)AG	+254733 - Kencell
+42379 -	
MOBILKOM(LIECHTENSTEIN)AG	Kuwait
	+9656 - Wataniya
Madagascar	+96594 - MTC-Kuwait
+261 - SMM	+96596 - MTC-Kuwait
+26133 - MADACOM	+96597 - MTC-Kuwait
	+96598 - MTC-Kuwait
Malaysia	
+6012 - MAXIS	Latvia
+6013 - TMTOUCH	+37192 - LMT
+6016 - DiGi	+37195 - BALTCOM GSM
+6017 - Time Cell	+37196 - BALTCOM GSM
+6019 - Celcom	
Mauritius	Lebanon
+230 - EMTEL	+96132 - FTML
+23025 - CellPlus	+96133 - FTML
+23075 - Cellplus	+96134 - FTML
+23076 - Cellplus	+96135 - FTML
	+96136 - Libancell
	+96137 - Libancell
	+96138 - Libancell
Netherlands	+96139 - Libancell
+31610 - KPN Telecom	
+31611 - Vodafone	Luxembourg
+31612 - KPN Telecom	+352021 - Luxembourg P&T
+31613 - KPN Telecom	

+31614 - T-Mobile	+352091 - TANGO
+31616 - Telfort	
+31617 - Telfort	Malawi
+31618 - Orange	+26590 - CELTEL Malawi
+31619 - Telfort	+26591 - CELTEL Malawi
+31620 - KPN Telecom	+26593 - CELTEL Malawi
+31621 - Vodafone	+26594 - CELTEL Malawi
+31622 - KPN Telecom	+26595 - CELTEL Malawi
+31623 - KPN Telecom	+26596 - CELTEL Malawi
+31624 - T-Mobile	+26597 - CELTEL Malawi
+31625 - Vodafone	+26598 - CELTEL Malawi
+31626 - Telfort	+26599 - CELTEL Malawi
+31627 - Vodafone	
+31628 - Orange	Morocco
+31629 - Vodafone	+21261 - Itissalat Al-Maghrib S.A
+31641 - T-Mobile	
+31644 - Telfort	Mozambique
+31645 - Telfort	+25882 - T.D.M.
+31648 - Orange	
+31650 - Vodafone	Namibia
+31651 - KPN Telecom	+26481 - MTC
+31652 - KPN Telecom	
+31653 - KPN Telecom	New Zealand
+31654 - Vodafone	+6421 - Vodafone
+31655 - Vodafone	
Portugal	Pakistan
+351 - Optimus	+92300 - Mobilink
Telecomunicacoes, S.A.	+923008 - Mobilink
+351 - Telecomunicacoes Moveis Nacionais S.A.	Philippines
+35191 - Telecel	+63916 - Globe Telecom, GMCR Inc.
+262 - Societe Reunionnais	+63917 - Globe Telecom, GMCR Inc.
+26265 - SRR	
+26266 - SRR	
+26285 - SRR	Poland
+26286 - SRR	+48 - Polska Telefonia Cyfrowa
+26287 - SRR	+48501 - Centertel
	+48502 - Centertel
Saudi Arabia	+48503 - Centertel
+96653 - STC	+48600 - ERA GSM
+96654 - STC	+48601 - Polkomtel
+96655 - STC	+48602 - ERA GSM
	+48603 - Polkomtel
Seychelles	+48604 - ERA GSM
+248 - Cable & Wireless (Seychelles) Ltd	+48605 - Polkomtel
+248 - Telecom (Seychelles Limited)	+48606 - ERA GSM
	+48607 - Polkomtel S.A.
	+48608 - ERA GSM
Singapore	Romania
+659001 ~ 9009 - Starhub	+4091 - MobilFon
+659010 ~ 9019 - SingTel Mobile	+4092 - MobiFon/Connex
+659020 ~ 9029 - Starhub	+4093 - MobilFon/Connex
+659030 ~ 9039 - SingTel Mobile	

+659040 ~ 9049 - M1	Russian Federation
+659050 ~ 9059 - SingTel Mobile	+7094 - Mobile Telesystems
+659060 ~ 9069 - Starhub	+7300 - K'CELL
+659070 ~ 9079 - M1	+734793 - Zao Smarts
+659081 ~ 9089 - SingTel Mobile	+7353255 - Zao Smarts
+659090 ~ 9099 - M1	+7501 - MOBILE
+659100 ~ 9109 - Starhub	TELESYSTEMS
+659110 ~ 9119 - SingTel Mobile	+775 - MOBILE TELESYSTEMS
+659120 ~ 9127 - SingTel Mobile	+781 - MOBILE TELESYSTEMS
+659130 ~ 9139 - SingTel Mobile	+7812 - Northwest
+659140 ~ 9149 - Starhub	+7814 - Northwest
+659150 ~ 9159 - SingTel Mobile	+7815 - Northwest
+659160 ~ 9169 - Starhub	+7816 - Northwest
+659170 ~ 9179 - SingTel Mobile	+7817 - Northwest
+659180 ~ 9186 - Starhub	+7818 - Northwest
+659190 ~ 9192 - M1	+7820 - Northwest
+659230 ~ 9236 - SingTel Mobile	+7821 - MOBILE
+659237 ~ 9240 - Starhub	TELESYSTEMS
+659270 ~ 9272 - Starhub	+7834 - NORTH-WEST GM
+659273 ~ 9279 - M1	+7841253 - Zao Smarts
+659340 ~ 9349 - M1	+7842249 - Zao Smarts
+659350 ~ 9359 - SingTel Mobile	+7843 - MOBILE
+659360 ~ 9369 - M1	TELESYSTEMS
+659370 ~ 9379 - SingTel Mobile	+7845 - NORTH-WEST GM
+659380 ~ 9389 - Starhub	+7851231 - Zao Smarts
+659390 ~ 9399 - SingTel Mobile	+79017 - KB Impuls
+659420 ~ 9422 - SingTel Mobile	+79021 - Mobile Telesystems
+659430 ~ 9439 - M1	+7902327 - Zao Smarts
+659440 ~ 9449 - SingTel Mobile	+790234 - Zao Smarts
+659450 ~ 9459 - Starhub	+7902363 - Zao Smarts
+659460 ~ 9469 - SingTel Mobile	+7902364 - Zao Smarts
+659470 ~ 9479 - M1	+790237 - Zao Smarts
+659480 ~ 9489 - Starhub	+790239 - MOBILE
+659610 ~ 9679 - SingTel Mobile	TELESYSTEMS
+659680 ~ 9699 - M1	+7902467 - MOBILE
+659710 ~ 9716 - SingTel Mobile	TELESYSTEMS
+659720 ~ 9739 - SingTel Mobile	+79026 - MOBILE
+659740 ~ 9749 - M1	TELESYSTEMS
+659750 ~ 9759 - SingTel Mobile	+79027 - MOBILE
+659760 ~ 9769 - M1	TELESYSTEMS
+659770 ~ 9789 - SingTel Mobile	+79036 - KB IMPULS
+659790 ~ 9799 - M1	+79037 - KB IMPULS
+659806 ~ 9807 - SingTel Mobile	
+659810 ~ 9812 - SingTel Mobile	Slovakia
+659815 ~ 9839 - SingTel Mobile	+421903 - Eurotel-Bratislava
+659840 ~ 9849 - M1	+421904 - Eurotel
+659850 ~ 9859 - Starhub	+421905 - Globtel GSM a.s.
+659860 ~ 9869 - SingTel Mobile	+421907 - Globtel
+659870 ~ 9879 - M1	+38631 - MOBITEL
+659890 ~ 9899 - SingTel Mobile	+38640 - SI.Mobil d.d.
	+386641 - Mobitel
Sri Lanka	
+9477 - MTN Networks (Pvt) Ltd	South Africa
	+2772 - Vodacom
Swaziland	+2773 - MTN

+268 - Swazi MTN (Ltd)

Switzerland

+4176 - diAx

+4178 - Orange

+4179 - Swisscom

United Kingdom

+44121276 - Orange UK

+44151571 - Orange UK

+44161380 - Orange UK

+44171362 - Orange UK

+44181264 - Orange UK

+44207362 - Orange UK

+44208264 - Orange UK

+44370 - Vodafone UK

+44374 - Vodafone UK

+44378 - Vodafone UK

+44385 - Vodafone UK

+44421 - Vodafone UK

+44441 - Vodafone UK

+444481 - Guernsey Telecom

+444624 - MANX

+44467 - Vodafone UK

+44468 - Vodafone UK

+44498 - Vodafone UK

+445957 - One-2-One

+447000 - Vodafone UK

+447002 - Vodafone UK

+447074 - Vodafone UK

+447624 - MANX TELECOM

+447701 - BT Cellnet

+447702 - BT Cellnet

+447703 - BT Cellnet

+447710 - BT Cellnet

+447711 - BT Cellnet

+447712 - BT Cellnet

+447713 - BT Cellnet

+447714 - BT Cellnet

+447715 - BT Cellnet

+447718 - BT Cellnet

+447719 - BT Cellnet

+447720 - BT Cellnet

+447729 - BT Cellnet

+447730 - BT Cellnet

+447731 - BT Cellnet

+447732 - BT Cellnet

+447734 - BT Cellnet

+447736 - BT Cellnet

+447740 - BT Cellnet

+447747 - Vodafone UK

+447748 - Vodafone UK

+447750 - BT Cellnet

+447751 - BT Cellnet

+447752 - BT Cellnet

+2782 - Vodacom

+2783 - MTN

Spain

+34600 - Airtel Movil

+34607 - Airtel Movil

+34610 - Airtel Movil

+34617 - Airtel Movil

+34627 - Airtel Movil

+34666 - Airtel Movil

+34667 - Airtel Movil

+34670 - Airtel Movil

+34677 - Airtel Movil

+34678 - Airtel Movil

+34679 - Telefonica Moviles

Sweden

+46450 - Telia Mobitel AB

+465918 - Telia Mobitel AB

+46592 - Telia Mobitel AB

+46593 - Telia Mobitel AB

+46595 - Telia Mobitel AB

+46596 - Telia Mobitel AB

+467011 - Europolitan AB

+467018 - Telia Mobitel AB

+46702 - Telia Mobitel AB

+46703 - Telia Mobitel AB

+46704 - Comviq GSM AB

+46705 - Telia Mobitel AB

+46706 - Telia Mobitel AB

+46707 - Comviq GSM AB

+46708 - Europolitan AB

+46709 - Europolitan AB

+46730 - Telia Mobitel AB

+46733 - Europolitan AB

+46736 - Comviq GSM AB

+46737 - Comviq GSM AB

+46739 - Comviq GSM AB

Tanzania

+255742 - TRITEL

+255812 - Tritel

Taiwan

+886 - Mobitai Communications Corp.

+88690 - CHUNGHWA

TELECOMS LDM

+886912 - CHUNGHWA

TELECOMS LDM

+886915 - KG

TELECOMMUNICATIONS CO.

+886916 - Far EastTone

Telecoms

+886917 - Far EastTone

+447753 - BT Cellnet	Telecoms
+447754 - BT Cellnet	+886919 - CHUNGHWA
+447759 - BT Cellnet	TELECOMS LDM
+447761 - BT Cellnet	+886921 - CHUNGHWA
+447771 - Vodafone UK	TELECOMS LDM
+447773 - Orange UK	+886925 - KG
+447775 - Vodafone UK	TELECOMMUNICATIONS CO.
+447776 - Vodafone	+886926 - Far EasTone
+447779 - Orange UK	Telecoms
+447781 - Guernsey Telecom	+886927 - KG
+447787 - Vodafone UK	TELECOMMUNICATIONS CO.
+447788 - Vodafone UK	+886928 - CHUNGHWA
+447790 - Orange UK	TELECOMS LDM
+447796 - Vodafone UK	+886930 - Far EasTone
+447799 - Vodafone UK	Telecoms
+447801 - BT Cellnet	+8869310 - Far EasTone
+447802 - BT Cellnet	Telecoms
+447803 - BT Cellnet	+8869311 - Far EasTone
+447808 - BT Cellnet	Telecoms
+447809 - BT Cellnet	+8869312 - Far EasTone
+447810 - Vodafone UK	Telecoms
+447811 - Orange UK	+8869313 - Far EasTone
+447812 - Orange UK	Telecoms
+447818 - Vodafone UK	+886932 - CHUNGHWA
+447831 - Vodafone UK	TELECOMS LDM
+447833 - Vodafone UK	+886933 - CHUNGHWA
+447850 - BT Cellnet	TELECOMS LDM
+447860 - BT Cellnet	+886936 - Far EasTone
+447867 - Vodafone UK	Telecoms
+447879 - Vodafone UK	+886937 - CHUNGHWA
+447880 - Vodafone UK	TELECOMS LDM
+447881 - Vodafone UK	+886955 - Far EasTone
+447885 - BT Cellnet	Telecoms
+447887 - Vodafone UK	
+447889 - BT Cellnet	Thailand
+447899 - Vodafone UK	+6618 - AIS (Mobitai)
+447901 - Vodafone UK	+66192 - AIS (Mobitai)
+447903 - Mercury One-to-one	+66193 - AIS (Mobitai)
+447904 - Mercury One-to-one	+661942 - AIS (Mobitai)
+447905 - Mercury One-to-one	+661970 - AIS (Mobitai)
+447909 - Vodafone UK	+661973 - AIS (Mobitai)
+447930 - Mercury One-to-one	+661975 - AIS (Mobitai)
+447931 - Mercury One-to-one	+661983 - AIS (Mobitai)
+447932 - Mercury One-to-one	+661990 - AIS (Mobitai)
+447939 - Mercury One-to-one	+661993 - AIS (Mobitai)
+447940 - Mercury One-to-one	+661994 - AIS (Mobitai)
+447941 - Mercury One-to-one	
+447944 - Mercury One-to-one	Togo
+447946 - Mercury One-to-one	+22801 - Togo Cellulaire
+447947 - Mercury One-to-one	+2284 - TOGO CELLULAIRE
+447949 - Mercury One-to-one	+2285 - TOGO CELLULAIRE
+447950 - Mercury One-to-one	
+447951 - Mercury One-to-one	Tunisia
+447952 - Mercury One-to-one	+2169 - Tunisie Telecom
+447953 - Mercury One-to-one	

+447956 - One-2-One	Uganda
+447957 - One-2-One	+25677 - MTN Uganda
+447958 - One-2-One	
+447959 - Mercury One-to-one	Ukraine
+447960 - Mercury One-to-one	+38067 - Kyivstar GSM JSC
+447961 - One-2-One	
+447962 - Mercury One-to-one	Turkey
+447966 - Orange UK	+90542 - Telsim
+447967 - Orange UK	+90543 - TELSIM
+447970 - Orange UK	+90545 - TELSIM
+447971 - Orange UK	+90546 - TELSIM
+447973 - Orange UK	
+447974 - Orange UK	United Arab Emirates
+447976 - Orange UK	+97150 - Etisalat
+447977 - Orange UK	
+447979 - Vodafone UK	United States
+447980 - Orange UK	+1 - Pacific Bell Wireless
+447989 - Orange UK	+1218 - VCS Midwest
+447990 - Vodafone UK	+1250 - MICROCELL
+44831 - Vodafone UK	+1252 - Bellsouth Mobility
+44836 - Vodafone UK	+1281 - VCS Midwest
+44930 - Mercury One-to-one	+1301 - APC Network
+44956 - Mercury One-to-one	+1316 - VCS Midwest
+44958 - Mercury One-to-one	+1316 - VCS Oklahoma
+44966 - Orange UK	+1320 - VCS Midwest
+44973 - Orange UK	+1321 - VCS Midwest
+44976 - Orange UK	+1330 - VCS New Mexico
+449797 - Jersey Telecom	+133433 - Powertel, Inc.
	+1336 - Bellsouth Mobility
Vietnam	+1336 - Bellsouth Mobility
+8490 - Veitnam VMS	+1352 - VCS Midwest
	+1403 - MICROCELL
Zimbabwe	+1405 - VCS Oklahoma
+263 - P+T Corporation	+1407 - VCS Midwest
+26311 - NETONE	+1409 - VCS Midwest
+26323 - Telecel	+1410 - APC Network
+26391 - Econet Wireless	+1410 - VCS Oklahoma
	+1412 - VCS Midwest
	+1416 - MICROCELL
	+1418 - MICROCELL
	+1423 - Bellsouth Mobility
	+1423 - Bellsouth Mobility
	+1450 - MICROCELL
	+1450 - VCS New Mexico
	+1451 - Bellsouth Mobility
	+1458 - Bellsouth Mobility
	+1460 - Bellsouth Mobility
	+1488 - Bellsouth Mobility
	+1491 - Bellsouth Mobility
	+1505 - VCS New Mexico
	+1507 - VCS Midwest
	+1512 - VCS Oklahoma
	+1514 - MICROCELL
	+1515771 - VCS Iowa
	+1519 - MICROCELL

+1602 - VCS New Mexico
+1604 - MICROCELL
+1605 - VCS Midwest
+1612 - VCS Midwest
+1613 - MICROCELL
+1614 - VCS Midwest
+1650 - Pacific Bell Wireless
+1651 - VCS Midwest
+1660 - VCS Midwest
+1701 - VCS Midwest
+1704 - Bellsouth Mobility
+1704 - Bellsouth Mobility
+1705 - MICROCELL
+1706 - Bellsouth Mobility
+1713 - VCS Midwest
+1715 - VCS Midwest
+1717 - APC Network
+1724 - VCS Midwest
+1727 - VCS Midwest
+1785 - VCS Midwest
+1801635 - VCS Utah
+1801859 - VCS Utah
+1803 - Bellsouth Mobility
+1803 - Bellsouth Mobility
+1803 - Bellsouth Mobility
+1813 - VCS Midwest
+1814 - VCS Midwest
+1816 - VCS Midwest
+1819 - MICROCELL
+1828 - Bellsouth Mobility
+1828 - Bellsouth Mobility
+1843 - Bellsouth Mobility
+1864 - Bellsouth Mobility
+1864 - Bellsouth Mobility
+1865 - Bellsouth Mobility
+1904 - VCS Midwest
+1905 - MICROCELL
+1910 - Bellsouth Mobility
+1912 - Bellsouth Mobility
+1913 - VCS Midwest
+1915 - VCS New Mexico
+1919 - Bellsouth Mobility
+1931 - Bellsouth Mobility
+1941 - VCS Midwest
+1977 - APC Network
+1990 - VCS Oklahoma

Appendix "C" - SMS service center addresses**ALBANIA**

AMC +3553820

ANDORRASTA-MOBILAND
+376301004**AUSTRALIA**OPTUS +6141190000
OPTUS +6141190001
OPTUS +61412025989
OPTUS +61411990003
TELSTRA AUSTRALIA
+61418706700
TELSTRA AUSTRALIA
+61418706800
TELSTRA AUSTRALIA
+61418706900
VODAFONE +61415011501
ONE.TEL AUSTRALIA
+61411990001**AUSTRIA**A1 +4366477
MAX.MOBIL +43676021
MAX.MOBIL +43676023
ONE +436990001999
TELE.RING
+4365009000000**AZERBAIJAN**AZERCELL +994509103300
BAKCELLI +994557070707**BAHARAIN**

BATELCO +9739600179

BANGLADESHGRAMEENPHONE
+88017099999
GRAMEENPHONE
+88017500569
GRAMEENPHONE
+880170000600**BELGIUM**PROXIMUS +3275161616
PROXIMUS +3275161612**ISRAEL**

ORANGE +97254120032

ITALYOMNITEL +393492000200
OMNITEL +393492000300
OMNITEL +393492000400
OMNITEL +393492000500
TIM +393359609600
TIM +39338960960
TIM +39338980000
WIND TELECOMUNICAZIONI
+393205858500**LATVIA**

LMT +3719202020

LEBANON

LEBANSELL +961348888

LITHUANIABITE GSM +3709950115
OMNITEL +3709899992**LUXEMBOURG**TANGO +352091000030
PTT +352021100003**MALAYSIA**CELCOM +60193900000
MAXIS +60126000015
MUTIARA +60122999000**MACEDONIA**

MOBIMAK +38970000501

MEXICO

Telcel +5294100001410

NETHERLANDSPTT +31653131314
PTT +31010600000
PTT +31653131313
LIBERTEL +316540881000
TELFORT +31626000230
DUTCHTONE +31628500516
BEN +31624000000**NORWAY**NETCOM +4792001600
TELENOR +4790002100

MOBISTAR +23495955205
MOBISTAR +32495002530
ORANGE +32486000005

**BOSNIA AND
HERZEGOVINA**

PTT GSM BIH
+38766125522

BOTSWANA

MASCOM WIRELESS
+26771010024
VISTA CELLULAR
+26772000003

BRUNEI DARUSSALAM

+673879500

BULGARIA

MOBILTEL +35988000301

CANADA

FIDO +15149931123

CHILE

ENTEL PCS +56-98890005

CHINA

CHINA TELECOM
+861390591500
CHINA TELECOM
+8613800100500

CROATIA

VIPNET +385910401
CRONET +385980501

CYPRUS

CYPRUS TELECOM
+3093599000

CZECH

EUROTEL +420602909909
RADIOMOBIL
+420603052000
RADIOMOBIL +42603051

DENMARK

SONOFON +4540590000
SONOFON +4543625250
TELEDENMARK
+4540590000

PHILIPPINE

SMART +639180000101
GLOBE +6391702

POLAND

ERA GSM +48602951111
POLKOMTEL +48601000310
POLKOMTEL +48601000311

PORTUGAL TELECEL

+351911616161
TMN +351962100000

ROMANIA

CONNEY +4092004000
DIALOG +4094946000

RUSSIA

MTS RUSSIA +70957699100
NORTH WEST +8129600096
BEELINE 1800 +790173100

SINGAPORE

SINGTEL +6596197777
SINGTEL +6596400001
SINGTEL +6596500001
SINGTEL +6598189999
M1 +6596845999

SLOVAKIA

GLOBTEL GSM +421905303303
EUROTEL +421903333000

SLOVENIA

MOBITEL +38641001333

SOUTH-AFRICA

MTN +27831000002
VODACOM +27829119
VODACOM +27829129

SPAIN

AMENA +34656000311
TELEFONICA +3460909090
TELEFONICA +3409090909
AIRTEL +34607133000
AIRTEL +34607003110

SWEDEN

AMENA +346500311
COMVIQ +46707990001
COMVIQ +46707990002
COMVIQ +46707990003

TDK-MOBIL +4540390999
TDK-MOBIL +4540320000
TDK-MOBIL +4540620000
TDK-MOBIL +4540690999
TELIA DK +4528187000
MOBILIX +4526265151

EGYPT

CLICK GSM +20105996500
MOBINIL +20122000020

ESTONIA

EMT +372509900
EMT +3725099000
EMT +372509000
Q-GSM +372550991
RADIOLINJA
+372568771010

FINLAND

RADIOLINJA
+358508771030
RADIOLINJA
+358508771010
SONERA +358405202999
SONERA +358405202000

FRANCE

BOUYGUES +33660003000
SFR +33609001930
SFR +33609001390
ITINERIS +33689004581
ITINERIS +33689004431
ITINERIS +33689004000
FRANCE TELECOM
+33007001080
FRANCE TELECOM
+33836620766
FRANCE TELECOM
+33836620766
ORGANE FRANCE
+33689004000

GERMANY

D1 +491712092522
D1 +491710760000
D1 +491710762222
D1 +491715990000
D1 +491710760900
D2 +491712227033
D2 +491722270000
D2 +491722270222

COMVIQ +46707773078
EUROPOLITON +46708000708
TELIA +46705008999

SWITZERLAND

SWISSCOM +41794998999
SWISSCOM +41794998123
SWISSCOM +41794999000
ORANGE COM +41787777070
DIAX-COMMUNICATIONS
+4176598000

THAILAND

AIS +6618110888
WP1800 +6616110400

TURKEY

TURKCELL +905329030000
TURKCELL +905329010000
TURKCELL +905329040000
TELSIM PTT +905429800033

UK

VODAFONE +44385016005
VODAFONE +447785014208
CELLNET +447802000332
CELLNET +447802092035
ONE2ONE +447958879879
ONE2ONE +447958879855
ONE2ONE +447958879856
ONE2ONE +44958879852
ORANGE +44973100973
ORANGE +44973100974
PRONTO GSM +447624299955

UKRAINE

KIEVSTAR +380672020000
GOLDEN +380444990000

UNITED ARAB EMIRTES

ETISALAT +971506014994

USA

SPRINT PCS +1410258953
OMNIPOINT PCS +19179070004
BELLSOUTH +17044100000
VOICESTREAM +12063130004
AERIAL +18132630025
AERIAL +12812350025
PACIFIC BELL +12099042020
POWERTELL +13313338200
DIGI PCS +13342090307

VENEZUELA

D2 +491722270333
D2 +491722270300
D2 +49172227033
D2 +491722270258
E-PLUS +491770600000
E-PLUS +491770610000
E-PLUS +491770620000
DEBITEL +491722270222
IC3S +491722270201
DR. MATERNA
+491722270111
MOBIDIG +491722270217

GREECE

TELESTET +3093599000
TELESTET +3093597000
PANAFON +3094219000
PANAFON +303850116005
PANAFON +30385016005
COSMOTE +3094219000
COSMOTE +3097100000

HONG KONG

TELECOM CSL +85290217
TELECOM CSL
+85290215772
HONG KONG TELE
+85290217572
HONG KONG TELE
+8529088000
HUCHKINSON
+85294985795
SMARTONE +85290100000

HUNGARY

PANNON +36209300099
T-MOBILE +36309888000
VODAFONE +36709996500

ICELAND

TELECOM +3548900100
TAL +3546999099
LANDSSIMINN
+35489800100

INDIA

BPL MOBILE
+919821000005
MAXTOUCH
+919820005446
PT PRATAMA
+62818445009

INFONET +58180000013

VIETNAM

VINAPHONE +8491020005
MOBIFONE +8490400012
MOBIFONE +8490500012
MOBIFONE +8490700000

YUGOSLAVIA

TELEKOM SRBIJA
+381650000900
PRO MONTE +38169200000
MOBTEL +38163100100
MOBTEL +38163100200
MOBTEL +38163100300
MOBTEL +38163100400
MOBTEL +3816310030034
MOBTEL +381631004003

ZAMBIA

VODACOM +27829119
VODACOM +2782 9129

ZIMBABWE

ECONET WIRELESS +263-
91010030
ZIMBABWE NET*ONE
+26311191201

INDONESIA

TELKOMSEL +62811130004

TELKOMSEL +62811000000

EXELCOMINDO

+62818445009

SATELINDO +6281615

IRELAND

EIRCELL +35387799985

EIRCELL +35387699989

ESAT +353868002000

Appendix "D" - GSM operator and country codes for operator Logos

The following country and operator codes can be used when an operator logo is being sent to a certain GSM device.

202-01	COSMOTE - Greece
202-05	Greece Panafon S A Panafon
202-10	Greece STET Hellas STET Hellas
204-04	Vodafone NL (Libertel) (GSM 900)
204-08	KPN Telecom (GSM 900)
204-12	Telfort (GSM 1800)
204-16	Ben (GSM 1800)
204-20	Dutchtone (GSM 1800)
206-01	Belgium Belgacom BEL PROXI
206-10	Belgium Mobistar BEL mobistar
206-20	KPN Orange Belgium SA
208-01	France France Telecom F Itineris
208-01	Monaco France Telecom F Itineris
208-10	France SFR F SFR
208-10	Monaco SFR F SFR
208-	France Bouygues Telecom Bouygues
213-03	Andorra STA Andorra AND M-AND
214-01	Spain Airtel Airtel
214-03	Spain AMENA
214-07	Spain Telefonica Moviles MSTAR
216-01	Hungary Pannon GSM Pannon
216-30	Hungary Westel H-Westel 900
216-70	Hungary Vodafone (GSM 900/1800)
218-01	Bosnia Cronet Cronet
218-90	Bosnia PTT Bosnia BIH PTT-GSM
219-01	Croatia HPT Croatian PTT HR-CRONET
219-10	Croatia VIPNET
220-01	Serbia Mobile Telecomm. MOBTEL
220-03	YU PTT
220-02	Montenegro PROMONTE GSM
220-04	Montenegro MoNet GSM
220-07	Serbia Promonte PROMONTE GSM
222-01	Italy Telecom Italia Mobile TIM
222-01	San Marino Telecom Italia Mobile TIM
222-01	The Vatican Telecom Italia Mobile TIM
222-10	Italy Omnitel Pronto Italia Omni
222-10	San Marino Omnitel Pronto Italia Omni

222-10	The Vatican Omnitel Pronto Italia Omni
222-88	Wind Telecomunicazioni S.P.A, Vatican and San Marino, Italy
222-98	Blu s.p.a. Telefonia mobile Italy
226-01	Romania MobiFon
226-03	Romania Cosmorom
226-10	Romania MobilRom
228-01	Switzerland Swiss PTT CH NAT D
228-01	Liechtenstein Swiss PTT CH NAT D
228-03	Orange Switzerland
228-08	Tele2 Switzerland
230-01	Czech republic Radio Mobil CZ Paegas
230-02	Czech republic Eurotel Praha EUROTEL-CZ
230-03	Czech republic OSKAR
231-01	Slovakia Globtel
231-02	Slovakia Eurotel Bratislava Eurotel Bratislava
232-01	Austria Mobilkom A1
232-	Austria O Call TS
232-03	Austria Max Mobil Telekoms A max.
232-05	Connect Austria Gesellschaft für Telekommunikation GmbH - ONE
232-07	tele.ring Telekom Service GmbH - Teling
234-10	UK Cellnet Cellnet
234-15	UK Vodafone Vodafone
234-30	UK One2One/Virgine
234-33	UK Orange
234-90	Jersey Jersey Telecoms Jer 1
234-55	Guersney Guersney Telecom GSY-TEL
234-58	Isle of Man Manx Telecom Manx
235-	UK
238-01	Denmark Tele Danmark Mobil DK TDK-MOBIL
238-02	Denmark SONOFON
238-	Denmark Telia Denmark Telia DK
238-	Denmark Mobilix Mobilix
240-01	Sweden Telia Mobitel S TELIA MOBITEL
240-07	Sweden Comviq S COMVIQ
240-08	Sweden Europolitan Europolitan
240-	Sweden Telenordia
242-01	Norway Telenor Mobil N Tele-mobil
242-02	Norway NetCom GSM A/S N NetCom GSM
244-05	Finland Telecom Finland Telecom Finland
244-91	Finland OY Radiolinja Radiolinja

244-	Finland Telivo
244-	Áland Alands Mobil
246-01	Lithuania Omnitel LT OMNITEL
246-02	Lithuania Mobilios Telekomunikacijos LT BITE
246-03	Lithuania LT TELE2
246-04	ir Lithuanian X-GSM Tele2
247-01	Latvia Latvias Mobilais Telefons LV LMT GSM
247-02	Latvia Baltcom LV BALTCOM
248-01	Estonia Eesti Mobiltelefon EE EMT GSM
248-02	Estonia Radiolinja Eesti AS RLE-GSM
248-03	Estonia Ritabell Q-GSM
290-01	Russia Mobil Telesystems MTS-RUS
290-02	Russia NW GSM St Petersburg RUS NWGSM
290-	Russia DonTelecom
290-	Russia United Tele Moscow UTM
290-38	Russia Wireless Technology
290-	Russia Extel Mobile Comms Systems
290-99	Russia KB Impuls
255-01	UKR FLASH
257-01	VELCOM Belarus
260-01	Poland Polkomtel SA PL-PLUS
260-02	Poland Polska Telefonicyfrowa PL-ERA GSM
260-03	Poland Polska Telefonikomorkowa Idea Centertel
262-01	Germany DeTeMobil D1 D1-Telekom
262-02	Germany Mannesmann Mobilfunk D2 D2-Privat
262-	Germany E-plus E-plus
262-	Germany VIAG E2
265-01	Ukraine UA UMC UA UMC
265-	Ukraine Ukrainian Radio System
265-	Ukraine Bancomsvyaz
266-01	Gibraltar Gibtel Gibtel GSM
268-01	Portugal Telecel Telecel
268-06	Portugal Telemovel P TMN
268-03	Optimus
270-01	Luxembourg P&T Luxembourg LUXGSM
270-77	Luxemburg (900/1800) Millicom.SA L TANGO
272-01	EIR Eircell IRL EIR-GSM
272-02	EIR Esat Digifone IRL DIGIFONE
274-01	Iceland Postur og simi IS SIMINN
276-01	Albania AMC AMC
278-	Malta Advanced
278-01	Malta Telecell Telecell

280-01	Cyprus Cyprus Telecom Auth CY CYTAGSM
283-01	RA-ARMGSM
284-01	Bulgaria MobiTel CITRON BG
286-01	Turkey TurkCell TURKCELL
286-02	Turkey Türk Telekom TR TELSİM
293-41	Slovenia Mobitel DD SI-GSM
293-	Slovenia Digitel Digitel
294-01	Macedonia PTT Makedonija MKD-MOBIMAK
302-37	Canada (PCS) Microcell FIDO
310-02	USA (PCS) Sprint Spectrum Sprint
310-15	USA (PCS) BellSouth Mobility PCS
310-	USA (PCS) Pacific Bell Mobile Svcs
310-	USA (PCS) Western Wireless Corp
310-20	USA (PCS) American Portable Telecoms Sprint
310-16	USA (PCS) Omnipoint Corporation
310-17	USA (PCS)
310-27	USA (PCS) Powertel PCS Partners
310-66	USA (PCS) DigiPH DigiPH
400-01	Azerbaijan AzerCell ACELL
401-01	Kazakhstan K-Mobile
401-02	Kazakhstan K-Cell
404-07	India TATA INA-TATA
404-10	India Airtel AIRTEL
404-11	India Essar ESSAR
404-12	India Escotel INA-ESCOTEL
404-19	India Escotel INA-ESCOTEL
404-20	India Max Touch MAXTOUCH
404-21	India BPL Mobile BPL MOBILE
404-27	India BPL Mobile BPL MOBILE
404-30	India Command COMMAND
404-31	India Mobilenet MOBILENET
404-40	India Skycell SKYCELL
404-41	India RPG MAA RPG MAA
404-	India Usha Martin
404-	India Modi Telstra
404-	India Sterling Cellular SCL
404-	India Mobile Telecom
404-	India Airtouch
404-43	India BPL Mobile BPL MOBILE
404-46	India BPL USWest BPL MOBILE
404-	India Koshiki
404-	India Bharti Telenet

404-	India Birla Comm
404-	India Cellular Comms
404-56	India Escotel INA-ESCOTEL
404-	India JT Mobiles
404-	India Evergrowth
404-	India Modicom
404-	India Fascel
410-01	Pakistan Mobilink MOBILINK
413-02	Sri Lanka MTN Networks Pvt Ltd SRI DALOG
415-01	Lebanon Cellis (FTML) RL Cellis
415-03	Lebanon Libancell RL LibCL
416-01	Jordan FastLink JOR FSTLNK
417-09	Syria Mobile Syria SYR MOB
418-	Iraq Iraq Telecoms + Posts
419-02	Kuwait Mobile Telecom KT MTCNet
420-01	Saudi Arabia MoPTT, DMTS-1 KSA ALJAWWAL
420-07	Saudi Arabia EAE KSA EAE
422-02	Oman General Telecoms GTO
424-01	Un Arab Emirat Etisalat UAE ETSLT
424-02	Un Arab Emirat Etisalat UAE EG2
425-	Israel Cellcom Israel Ltd Cellcom
426-01	Bahrein Batelco BHR MPLUS
427-01	Qatar QTel QAT Q-NET
432-11	Iran T.C.I.
432-	Iran Celcom
432-	Iran Kish Free Zone
438-01	Turkmenistan BCTI
452-01	Vietnam MTCS VMS-GSM
452-02	Vietnam Vinaphone (GSM 900)
454-00	Hong Kong Hong Kong Telecom HK TCSL GSM
454-04	Hong Kong Hutchinson HutchinsonGSM
454-06	Hong Kong Smartone Mobile Comm SmarTone
454-	Hong Kong Peoples Telephone Co Ltd
455-01	Macao C.T.M. CTM GSM
457-01	Laos Lao Shinawatra Telecom
460-00	China Guangdong MCC CHNTELGSM
460-01	China China United Telecom CHN-CUGSM
460-	China Guangxi PTB
460-	China Liaoning PPTA
460-	China Beijing Wireless
460-	China Zhuhai Comms
460-	China Jiaxing PTT

460-	China Tjianjin Toll Telecom
460-	China DGT MPT
466-92	Taiwan Shungwa Telecom LDM ROCLDGSM
902-01	Malaysia My Digi MY DIGI
902-02	Malaysia My BSB MY BSB
902-12	Malaysia Binariang Maxis Mobile MY maxis
902-13	Malaysia Telecom Malaysia Touch MY MRTEL
902-16	Malaysia Mutiara Telekom DIGI 1800
902-17	Malaysia Sapura Digital Adam PHS MY ADAM
902-19	Malaysia Celcom GSM Celcom
905-01	Australia Telstra Corporation Limited Telstra
905-02	Australia Cable & Wireless Optus Limited YES OPTUS
905-03	Australia Vodafone Network Pty Limited VODAFONE
905-08	Australia One-Tel (GSM 1800)
510-	Indonesia PT Kartika Ekamas
510-01	Indonesia PT Satelindo IND SAT-C
510-10	Indonesia Telekomsel TELKOMSELGSM
510-11	Indonesia Excelcom IND-EXCELCOM
510-15	Indonesia Telekomindo Telekomindo
515-01	The Philipines IslaCom ISLA
515-02	The Philipines Globe Telecom GLOBE
515-03	SMART telecommunications, Phlippines
520-01	Thailand Advanced Info Serv Pcl TH AIS GSM
520-18	Thailand Tacs WP1800
525-01	Singapore Singapore Telecom ST-GSM-SGP
525-03	Singapore Mobile One M1-GSM-SGP
528-01	Brunei Jabatan Telekom
528-11	Brunei DSTCom
530-01	New Zealand Bell South BELLSOUTH021
542-01	Fiji Vodafone Fiji VODAFONE
546-01	New Caledonia Mobilis Mobilis
547-20	Fr Polynesia Tikiphone F VINI Fr Pacific Isls Guinea International Wireless
604-01	Morocco ONPT Marocko MOR ONPT
605-02	Tunesia
608-01	Senegal Sonatel
612-01	Ivory Coast Comstar
612-03	Ivory Coast Ivoiris
612-05	Telecel
617-01	Mauritius Cellplus Mobile Comms MRU- CELLPLUS

624-01	Cameroon PTT Cameroon Cellnet CAM CELLNET
633-01	The Seychelles SEZ SEYCEL SEZ SEYCEL
634-01	Sudan MobiTel SDN MobiTel
636-01	Ethiopia ETA ETH-MTN
640-01	Tanzania TriTel TZ-Tritel
641-01	Uganda Celtel Cellular CELTEL
641-10	Uganda Mobile Telephone Networks MTN Uganda.
646-01	Madacom (Madagascar)
646-02	Antaris SMM (Madagascar)
646-03	SACEL (Madagascar)
647-10	Reunion SRR F SFR RU
648-01	Zimbabwe PTC Zimbabwe NET ONE
649-01	Namibia MTC Network NAM MTC
690-01	Malawi Telekom Network MW CP 900
651-01	Lesotho Vodacom VCL COMMS
652-01	Botswana, Mascom Wireless
652-02	Botswana, Vista Cellular
655-01	South Africa Vodacom VodaCom-SA
655-10	South Africa Mobile Telephone Networks MTN-SA

Appendix "E" - The 7 bit default alphabet of GSM phones

Your PC and your GSM phone uses two different character sets the ISO-8859-1 and the GSM 7 bit alphabet. Ozeki SMS makes a conversion automatically between the corresponding character codes in these two tables. In some cases the automatic conversion might not be fit for your needs. If you want to be sure, that a specific GSM character is used in your SMS message, you can enter its Hex value into the message. Before we explain how this is possible, please take a look at what characters are there in GSM.

This is the 7 bit default alphabet as specified by GSM 03.38. The corresponding ISO-8859-1 decimal codes are shown in the rightmost column. Note that the euro sign (€) is a member of the ISO-8859-15 character set.

Hex	Dec	Character name	Character	ISO-8859-1 DEC
0x00	0	COMMERCIAL AT	@	64
0x01	1	POUND SIGN	£	163
0x02	2	DOLLAR SIGN	\$	36
0x03	3	YEN SIGN	¥	165
0x04	4	LATIN SMALL LETTER E WITH GRAVE	è	232
0x05	5	LATIN SMALL LETTER E WITH ACUTE	é	233
0x06	6	LATIN SMALL LETTER U WITH GRAVE	ù	250
0x07	7	LATIN SMALL LETTER I WITH GRAVE	ì	236
0x08	8	LATIN SMALL LETTER O WITH GRAVE	ò	242
0x09	9	LATIN CAPITAL LETTER C WITH CEDILLA	Ç	199
0x0A	10	LINE FEED		10
0x0B	11	LATIN CAPITAL LETTER O WITH STROKE	Ô	216
0x0C	12	LATIN SMALL LETTER O WITH STROKE	ô	248
0x0D	13	CARRIAGE RETURN		13
0x0E	14	LATIN CAPITAL LETTER A WITH RING ABOVE	Å	197
0x0F	15	LATIN SMALL LETTER A WITH RING ABOVE	å	229
0x10	16	GREEK CAPITAL LETTER DELTA	Δ	
0x11	17	LOW LINE	_	95
0x12	18	GREEK CAPITAL LETTER	Φ	

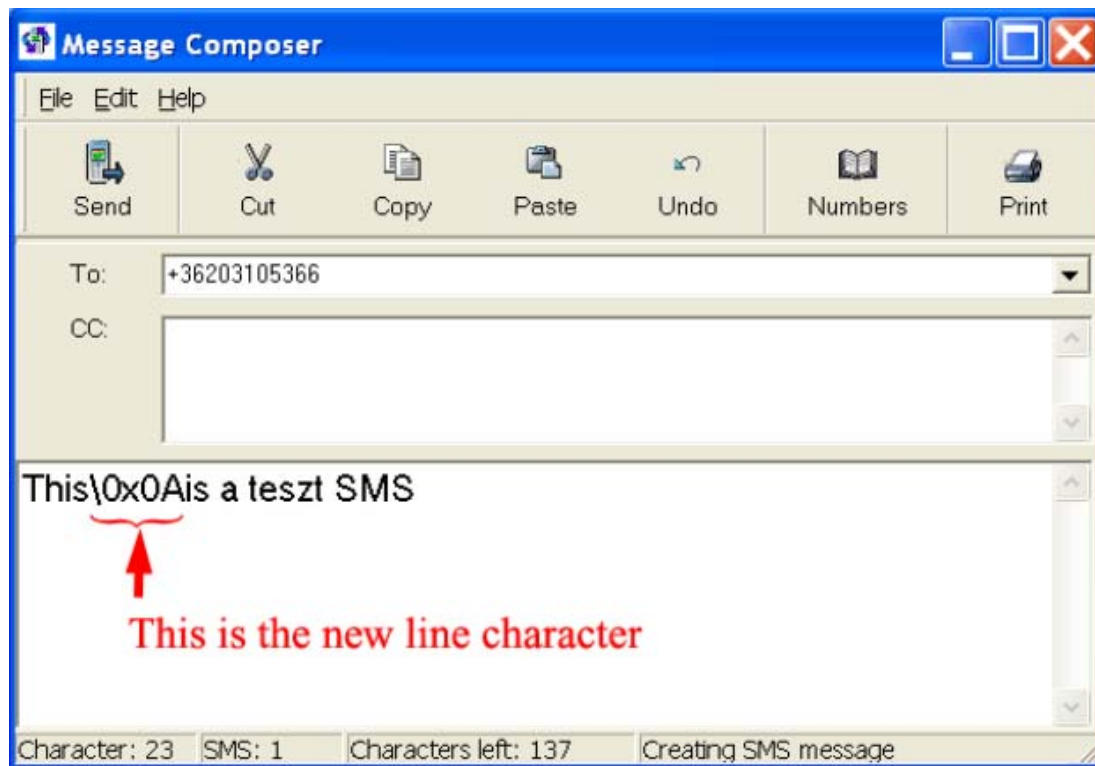
		PHI		
0x13	19	GREEK CAPITAL LETTER GAMMA	Γ	
0x14	20	GREEK CAPITAL LETTER LAMBDA	Λ	
0x15	21	GREEK CAPITAL LETTER OMEGA	Ω	
0x16	22	GREEK CAPITAL LETTER PI	Π	
0x17	23	GREEK CAPITAL LETTER PSI	Ψ	
0x18	24	GREEK CAPITAL LETTER SIGMA	Σ	
0x19	25	GREEK CAPITAL LETTER THETA	Θ	
0x1A	26	GREEK CAPITAL LETTER XI	Ξ	
0x1B	27	ESCAPE TO EXTENSION TABLE		
0x1B0A	27 10	FORM FEED		12
0x1B14	27 20	CIRCUMFLEX ACCENT	^	94
0x1B28	27 40	LEFT CURLY BRACKET	{	123
0x1B29	27 41	RIGHT CURLY BRACKET	}	125
0x1B2F	27 47	REVERSE SOLIDUS (BACKSLASH)	\	92
0x1B3C	27 60	LEFT SQUARE BRACKET	[91
0x1B3D	27 61	TILDE	~	126
0x1B3E	27 62	RIGHT SQUARE BRACKET]	93
0x1B40	27 64	VERTICAL BAR		124
0x1B65	27 101	EURO SIGN	€	164 (ISO-8859-15)
0x1C	28	LATIN CAPITAL LETTER AE	Æ	198
0x1D	29	LATIN SMALL LETTER AE	æ	230
0x1E	30	LATIN SMALL LETTER SHARP S (German)	ß	223
0x1F	31	LATIN CAPITAL LETTER E WITH ACUTE	É	202

0x20	32	SPACE		32
0x21	33	EXCLAMATION MARK	!	33
0x22	34	QUOTATION MARK	"	34
0x23	35	NUMBER SIGN	#	35
0x24	36	CURRENCY SIGN	¤	164 (ISO-8859-1)
0x25	37	PERCENT SIGN	%	37
0x26	38	AMPERSAND	&	38
0x27	39	APOSTROPHE	'	39
0x28	40	LEFT PARENTHESIS	(40
0x29	41	RIGHT PARENTHESIS)	41
0x2A	42	ASTERISK	*	42
0x2B	43	PLUS SIGN	+	43
0x2C	44	COMMA	,	44
0x2D	45	HYPHEN-MINUS	-	45
0x2E	46	FULL STOP	.	46
0x2F	47	SOLIDUS (SLASH)	/	47
0x30	48	DIGIT ZERO	0	48
0x31	49	DIGIT ONE	1	49
0x32	50	DIGIT TWO	2	50
0x33	51	DIGIT THREE	3	51
0x34	52	DIGIT FOUR	4	52
0x35	53	DIGIT FIVE	5	53
0x36	54	DIGIT SIX	6	54
0x37	55	DIGIT SEVEN	7	55
0x38	56	DIGIT EIGHT	8	56
0x39	57	DIGIT NINE	9	57
0x3A	58	COLON	:	58
0x3B	59	SEMICOLON	;	59
0x3C	60	LESS-THAN SIGN	<	60
0x3D	61	EQUALS SIGN	=	61
0x3E	62	GREATER-THAN SIGN	>	62
0x3F	63	QUESTION MARK	?	63
0x40	64	INVERTED EXCLAMATION MARK	¿	161
0x41	65	LATIN CAPITAL LETTER A	A	65
0x42	66	LATIN CAPITAL LETTER B	B	66
0x43	67	LATIN CAPITAL LETTER C	C	67
0x44	68	LATIN CAPITAL LETTER D	D	68
0x45	69	LATIN CAPITAL LETTER E	E	69
0x46	70	LATIN CAPITAL LETTER F	F	70

0x47	71	LATIN CAPITAL LETTER G	G	71
0x48	72	LATIN CAPITAL LETTER H	H	72
0x49	73	LATIN CAPITAL LETTER I	I	73
0x4A	74	LATIN CAPITAL LETTER J	J	74
0x4B	75	LATIN CAPITAL LETTER K	K	75
0x4C	76	LATIN CAPITAL LETTER L	L	76
0x4D	77	LATIN CAPITAL LETTER M	M	77
0x4E	78	LATIN CAPITAL LETTER N	N	78
0x4F	79	LATIN CAPITAL LETTER O	O	79
0x50	80	LATIN CAPITAL LETTER P	P	80
0x51	81	LATIN CAPITAL LETTER Q	Q	81
0x52	82	LATIN CAPITAL LETTER R	R	82
0x53	83	LATIN CAPITAL LETTER S	S	83
0x54	84	LATIN CAPITAL LETTER T	T	84
0x55	85	LATIN CAPITAL LETTER U	U	85
0x56	86	LATIN CAPITAL LETTER V	V	86
0x57	87	LATIN CAPITAL LETTER W	W	87
0x58	88	LATIN CAPITAL LETTER X	X	88
0x59	89	LATIN CAPITAL LETTER Y	Y	89
0x5A	90	LATIN CAPITAL LETTER Z	Z	90
0x5B	91	LATIN CAPITAL LETTER A WITH DIAERESIS	Ä	196
0x5C	92	LATIN CAPITAL LETTER O WITH DIAERESIS	Ö	214
0x5D	93	LATIN CAPITAL LETTER N WITH TILDE	Ñ	209
0x5E	94	LATIN CAPITAL LETTER U WITH DIAERESIS	Ü	220
0x5F	95	SECTION SIGN	§	167
0x60	96	INVERTED QUESTION MARK	¿	191
0x61	97	LATIN SMALL LETTER A	a	97
0x62	98	LATIN SMALL LETTER B	b	98
0x63	99	LATIN SMALL LETTER C	c	99
0x64	100	LATIN SMALL LETTER D	d	100
0x65	101	LATIN SMALL LETTER E	e	101
0x66	102	LATIN SMALL LETTER F	f	102
0x67	103	LATIN SMALL LETTER G	g	103
0x68	104	LATIN SMALL LETTER H	h	104
0x69	105	LATIN SMALL LETTER I	i	105
0x6A	106	LATIN SMALL LETTER J	j	106

0x6B	107	LATIN SMALL LETTER K	k	107
0x6C	108	LATIN SMALL LETTER L	l	108
0x6D	109	LATIN SMALL LETTER M	m	109
0x6E	110	LATIN SMALL LETTER N	n	110
0x6F	111	LATIN SMALL LETTER O	o	111
0x70	112	LATIN SMALL LETTER P	p	112
0x71	113	LATIN SMALL LETTER Q	q	113
0x72	114	LATIN SMALL LETTER R	r	114
0x73	115	LATIN SMALL LETTER S	s	115
0x74	116	LATIN SMALL LETTER T	t	116
0x75	117	LATIN SMALL LETTER U	u	117
0x76	118	LATIN SMALL LETTER V	v	118
0x77	119	LATIN SMALL LETTER W	w	119
0x78	120	LATIN SMALL LETTER X	x	120
0x79	121	LATIN SMALL LETTER Y	y	121
0x7A	122	LATIN SMALL LETTER Z	z	122
0x7B	123	LATIN SMALL LETTER A WITH DIAERESIS	ä	228
0x7C	124	LATIN SMALL LETTER O WITH DIAERESIS	ö	246
0x7D	125	LATIN SMALL LETTER N WITH TILDE	ñ	241
0x7E	126	LATIN SMALL LETTER U WITH DIAERESIS	ü	252
0x7F	127	LATIN SMALL LETTER A WITH GRAVE	à	224

If you wish to use any of these characters, you can do it by entering a special string into your SMS messages. For example if you wish to add a new line character you should enter `\0x0A` instead pressing enter. Make sure you write out all the characters of this code. It is a five-letter-statement!



Appendix "F" - GSM error codes

If a message send attempt fails, the Ozeki Message Server creates a log entry, containing one of the following error codes:

1	Unassigned (unallocated) number
8	Operator determined barring
10	Call barred
21	Short message transfer rejected
27	Destination out of service
28	Unidentified subscriber
29	Facility rejected
30	Unknown subscriber
38	Network out of order
41	Temporary failure
42	Congestion
47	Resources unavailable, unspecified
50	Requested facility not subscribed
69	Requested facility not implemented
81	Invalid short message transfer reference value
95	Invalid message, unspecified
96	Invalid mandatory information
97	Message type non-existent or not implemented
98	Message not compatible with short message protocol state
99	Information element non-existent or not implemented
111	Protocol error, unspecified
127	Interworking, unspecified
128	Telematic interworking not supported
129	Short message Type 0 not supported
130	Cannot replace short message
143	Unspecified TP-PID error
144	Data coding scheme (alphabet) not supported
145	Message class not supported
159	Unspecified TP-DCS error
160	Command cannot be actioned
161	Command unsupported
175	Unspecified TP-Command error
176	TPDU not supported
192	SC busy
193	No SC subscription

194	SC system failure
195	Invalid SME address
196	Destination SME barred
197	SM Rejected-Duplicate SM
198	TP-VPF not supported
199	TP-VP not supported
208	D0 SIM SMS storage full
209	No SMS storage capability in SIM
210	Error in MS
211	Memory Capacity Exceeded
212	SIM Application Toolkit Busy
213	SIM data download error
255	Unspecified error cause
300	ME failure
301	SMS service of ME reserved
302	Operation not allowed
303	Operation not supported
304	Invalid PDU mode parameter
305	Invalid text mode parameter
310	SIM not inserted
311	SIM PIN required
312	PH-SIM PIN required
313	SIM failure
314	SIM busy
315	SIM wrong
316	SIM PUK required
317	SIM PIN2 required
318	SIM PUK2 required
320	Memory failure
321	Invalid memory index
322	Memory full
330	SMSC address unknown
331	no network service
332	Network timeout
340	NO +CNMA ACK EXPECTED
500	Unknown error
512	User abort

Appendix "G" - Database plugin connection strings

This section gives you the connection strings you can use to connect to your database server in the database plugin of Ozeki Message Server 6. To configure the database plugin, you have to enter one of these connection strings on the database plugin configuration form (Figure 1). In some situations the SQL templates need to be modified in order to have database connectivity. This can be done on the SQL Templates tab of this form.

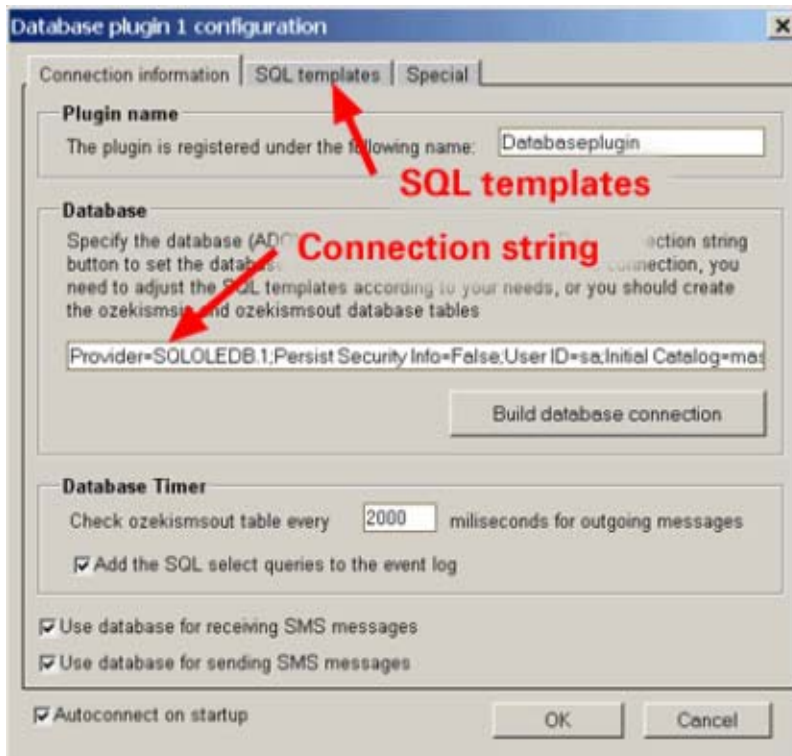


Figure 1. - Database plugin configuration form

The connection strings are organized into groups according to the following table of contents:

- [Overview](#)
- [DSN-Less Connection](#)
 - [dBASE](#)
 - [Excel](#)
 - [Text File](#)
 - [Visual FoxPro](#)
 - [Access database](#)
 - [SQL Server](#)
 - [Oracle](#)
 - [MySQL](#)
 - [AS400](#)
 - [Sybase](#)
 - [Sybase SQL AnyWhere](#)
- [DSN Connection](#)
 - [ODBC DSN](#)
- [OLE DB Provider](#)
 - [SQL Server](#)
 - [MySQL](#)
 - [AS400](#)

- [Active Directory](#)
- [DB2](#)
- [Microsoft Jet](#)
- [ODBC Databases](#)
- [OLAP](#)
- [Oracle](#)
- [Visual FoxPro](#)
- [Index Server](#)
- **[OLE DB Data Link Connections](#)**
 - [Data Link File - UDL](#)

Overview

Generally, one of the first steps when you are trying to work with databases is open it. You can find several types of those, and each have a different mode of connection. When you try to connect with your database sometimes, you don't know the correct *connection string* that you must use. It is for that this chapter is written. It contains the connection strings to the majority of known databases...

ODBC DSN Less Connection ODBC Driver for dBASE

```
Connection string: "Driver={Microsoft dBASE Driver (*.dbf)};DriverID=277;Dbq=c:\DatabasePath;"
```

Note: You must specify the filename in the SQL template statement... For example:

```
"Select Name, Address From Clients.dbf"
```

ODBC Driver for Excel

```
Connection string: "Driver={Microsoft Excel Driver (*.xls)};DriverId=790;bq=C:\DatabasePath\DBSpreadSheet.xls;DefaultDir=c:\databasepath;"
```

ODBC Driver for Text

```
Connection string: "Driver={Microsoft Text Driver (*.txt; *.csv)};Dbq=C:\DatabasePath\;Extensions=asc, csv, tab, txt;"
```

If you are using tab delimited files, you must create the [schema.ini](#) file, and you must inform the `Format=TabDelimited` option in your connection string.

Note: You must specify the filename in the SQL template statement... For example:

```
"Select Name, Address From Clients.csv"
```

Visual FoxPro

If you are using a *database container*, the connection string is the following:

```
Connection string: "Driver={Microsoft Visual Foxpro Driver};UID=;SourceType=DBC;
SourceDB=C:\DatabasePath\MyDatabase.dbc;Exclusive=No"
```

If you are working without a *database container*, you must change the `SourceType` parameter by `DBF` as in the following connection string:

```
Connection string: "Driver={Microsoft Visual Foxpro Driver};UID=;SourceType=DBF;
SourceDB=C:\DatabasePath\MyDatabase.dbc;Exclusive=No"
```

ODBC Driver for Access

```
Connection string: "Driver={Microsoft Access Driver (*.mdb)};
Dbq=c:\DatabasePath\dbaccess.mdb;Uid=;Pwd=;"
```

If you are using a Workgroup (System database): you need to inform the *SystemDB Path*, the *User Name* and its *password*. For that, you have two solutions: inform the *user* and *password* in the connection string or in the moment of the *open* operation. For example:

```
Connection string: "Driver={Microsoft Access Driver (*.mdb)};
Dbq=C:\VC Projects\ADO\Samples\AdoTest\dbTestSecurity.mdb;
SystemDB=C:\Program Files\Microsoft Office\Office\SYSTEM.mdw;
Uid=Carlos Antollini;Pwd=carlos"
```

or may be:

```
Connection string: "Driver={Microsoft Access Driver (*.mdb)};
Dbq=C:\VC Projects\ADO\Samples\AdoTest\dbTestSecurity.mdb;
SystemDB=C:\Program Files\Microsoft Office\Office\SYSTEM.mdw;"
```

If you want to open in Exclusive mode:

```
Connection string: "Driver={Microsoft Access Driver (*.mdb)};
Dbq=c:\DatabasePath\dbaccess.mdb;Exclusive=1;"
```

ODBC Driver for SQL Server

For Standard security:

```
Connection string: "Driver={SQL Server};Server=MyServerName;
Trusted_Connection=no;Database=MyDatabaseName;Uid=MyUserName;Pwd=MyPassword;"
```

For Trusted Connection security (Microsoft Windows NT integrated security):

```
Connection string: "Driver={SQL Server};Server=MyServerName;
Database=myDatabaseName;Uid=;Pwd=;"
```

Also, you can use the parameter `Trusted_Connection` that indicates that you are using the Microsoft Windows NT Authentication Mode to authorize user access to the SQL Server database. For example:

```
Connection string: "Driver={SQL Server};Server=MyServerName;
Database=MyDatabaseName;Trusted_Connection=yes;"
```

If the SQL Server is running in the same computer, you can replace the name of the server by the word `(local)` like in the following sample:

```
Connection string: "Driver={SQL Server};Server=(local);
Database=MyDatabaseName;Uid=MyUserName;Pwd=MyPassword;"
```

If you want to connect with a remote SQL Server, you must inform the address, the port, and the Network Library to use:

The `Address` parameter must be an IP address and must include the port. The `Network` parameter can be one of the following:

- `dbnmpntw` Win32 Named Pipes
- `dbmssocn` Win32 Winsock TCP/IP
- `dbmsspxn` Win32 SPX/IPX
- `dbmsvinn` Win32 Banyan Vines
- `dbmsrpcn` Win32 Multi-Protocol (Windows RPC)

```
Connection string: "Driver={SQL
Server};Server=130.120.110.001;Address=130.120.110.001,1052;
Network=dbmssocn;Database=MyDatabaseName;Uid=myUsername;Pwd=myPassword;"
```

ODBC Driver for Oracle

For the current Oracle ODBC driver from Microsoft:

```
Connection string: "Driver={Microsoft ODBC for Oracle};
Server=OracleServer.world;Uid=MyUsername;Pwd=MyPassword;"
```

For the older Oracle ODBC driver from Microsoft:

```
Connection string: "Driver={Microsoft ODBC Driver for Oracle};
ConnectionString=OracleServer.world;Uid=myUsername;Pwd=myPassword;"
```

ODBC Driver for MySQL

If you want to connect to a local database, you can use a connection string like the following:

```
Connection string: "Driver={MySQL ODBC 3.51 Driver};Server=localhost;
Database=MyDatabase;User=MyUserName;Password=MyPassword;Option=4;"
```

If you want to connect with a remote database, you need to specify the name of the server or its IP in the `server` parameter. If the Port is distinct to 3306 (default port), you must specify it.

```
Connection string: "Driver={mySQL ODBC 3.51 Driver};Server=MyRemoteHost;
Port=3306;Option=4;Database=MyDatabase;Uid=MyUsername;Pwd=MyPassword;"
```

The parameter `option` can be one or more of the following values:

- 1 - The client can't handle that MyODBC returns the real width of a column.

- 2 - The client can't handle that MySQL returns the true value of affected rows. If this flag is set then MySQL returns 'found rows' instead. One must have MySQL 3.21.14 or newer to get this to work.
- 4 - Make a debug log in *c:\myodbc.log*. This is the same as putting `MYSQL_DEBUG=d:t:0,c:\myodbc.log` in *AUTOEXEC.BAT*.
- 8 - Don't set any packet limit for results and parameters.
- 16 - Don't prompt for questions even if driver would like to prompt.
- 32 - Enable or disable the dynamic cursor support. This is not allowed in MyODBC 2.50.
- 64 - Ignore use of database name in 'database.table.column'.
- 128 - Force use of ODBC manager cursors (experimental).
- 256 - Disable the use of extended fetch (experimental).
- 512 - Pad CHAR fields to full column length.
- 1024 - `SQLDescribeCol()` will return fully qualified column names.
- 2048 - Use the compressed server/client protocol.
- 4096 - Tell server to ignore space after function name and before '(' (needed by PowerBuilder). This will make all function names keywords!
- 8192 - Connect with named pipes to a MySQLd server running on NT.
- 16384 - Change LONGLONG columns to INT columns (some applications can't handle LONGLONG).
- 32768 - Return 'user' as `Table_qualifier` and `Table_owner` from `SQLTables` (experimental).
- 65536 - Read parameters from the client and ODBC groups from *my.cnf*.
- 131072 - Add some extra safety checks (should not be needed but...).

If you want to have multiple options, you should add the above flags! For example: **16 + 1024 = 1030** and use `Option= 1030;`.

For more information, go to [MyODBC Reference Manual](#).

ODBC Driver for AS400

```
Connection string: "Driver={Client Access ODBC Driver (32-bit)};
System=myAS400;Uid=myUsername;Pwd=myPassword;"
```

ODBC Driver for SyBase

```
Connection string: "Driver={Sybase System 10};Srvr=MyServerName;
Uid=MyUsername;Pwd=myPassword;"
```

ODBC Driver for Sybase SQL AnyWhere

```
Connection string: "ODBC;Driver=Sybase SQL Anywhere 5.0;DefaultDir=C:\DatabasePath\;
Dbf=C:\SqlAnywhere50\MyDatabase.db;Uid=MyUsername;Pwd=MyPassword;Dsn="";"
```

DSN Connection ODBC DSN

```
Connection string: "DSN=MyDSN;Uid=MyUsername;Pwd=MyPassword;"
```

OLE DB Provider OLE DB Provider for SQL Server

For Standard security:

```
Connection string: "Provider=sqloledb;Data Source=MyServerName;  
Initial Catalog=MyDatabaseName;User Id=MyUsername;Password=MyPassword;"
```

For Trusted Connection security (Microsoft Windows NT integrated security):

```
Connection string: "Provider=sqloledb;Data Source=MyServerName;  
Initial Catalog=MyDatabaseName;Integrated Security=SSPI;"
```

If you want to connect to a "Named Instance" (SQL Server 2000), you must to specify `Data Source=Server Name\Instance Name` like in the following example:

```
Connection string: "Provider=sqloledb;Data Source=MyServerName\MyInstanceName;  
Initial Catalog=MyDatabaseName;User Id=MyUsername;Password=MyPassword;"
```

If you want to connect with a SQL Server running on the same computer, you must specify the keyword `(local)` in the `Data Source` like in the following example:

```
Connection string: "Provider=sqloledb;Data Source=(local);  
Initial Catalog=myDatabaseName;User ID=myUsername;Password=myPassword;"
```

To connect to SQL Server running on a remote computer (via an IP address):

```
Connection string: "Provider=sqloledb;Network Library=DBMSSOCN;  
Data Source=130.120.110.001,1433;Initial Catalog=MyDatabaseName;  
User ID=MyUsername;Password=MyPassword;"
```

OLE DB Provider for MySQL (By Todd Smith)

```
Connection string: "Provider=MySQLProv;Data Source=test"
```

Where `test` is the name of MySQL database. Also, you can replace the name of the database by the following connection string: `server=localhost;DB=test`.

OLE DB Provider for AS400

```
Connection string: "Provider=IBMDA400;Data source=myAS400;  
User Id=myUsername;Password=myPassword;"
```

For more information, see: [Using the OLE DB Provider for AS/400 and VSAM](#).

OLE DB Provider for Active Directory

```
Connection string: "Provider=ADSDSOObject;User  
Id=myUsername;Password=myPassword;"
```

For more information, see: [Microsoft OLE DB Provider for Microsoft Active Directory Service](#).

OLE DB Provider for DB2

If you are using a TCP/IP connection:

```
Connection string: "Provider=DB2OLEDB;Network Transport Library=TCPIP;
Network Address=130.120.110.001;Initial Catalog=MyCatalog;
Package Collection=MyPackageCollection;Default Schema=MySchema;
User ID=MyUsername;Password=MyPassword;"
```

If you are using APPC connection:

```
Connection string: "Provider=DB2OLEDB;APPC Local LU Alias=MyLocalLUAlias;
APPC Remote LU Alias=MyRemoteLUAlias;Initial Catalog=MyCatalog;
Package Collection=MyPackageCollection;Default Schema=MySchema;
User ID=MyUsername;Password=MyPassword;"
```

For more information, see: [Using the OLE DB Provider for DB2](#).

OLE DB Provider for Microsoft Jet Connecting to an Access file using the JET OLE DB Provider:

Using Standard security:

```
Connection string: "Provider=Microsoft.Jet.OLEDB.4.0;
Data Source=C:\DatabasePath\MmDatabase.mdb;User Id=admin;Password=";
```

If you are using a Workgroup (System database):

```
Connection string: "Provider=Microsoft.Jet.OLEDB.4.0;
Data Source=C:\DataBasePath\mydb.mdb;Jet OLEDB:System Database=MySystem.mdw;"
```

Connecting to an Excel Spreadsheet using the JET OLE DB Provider:

```
Connection string: "Provider=Microsoft.Jet.OLEDB.4.0;
Data Source=C:\DatabasePath\DBSpreadSheet.xls;
Extended Properties="\"Excel 8.0;HDR=Yes;\";"
```

Note: If "HDR=Yes", the provider will not include the first row of the selection into the recordset. If "HDR=No", the provider will include the first row of the cell range (or named ranged) into the recordset.

Connecting to a Text file using the JET OLE DB Provider:

```
Connection string: "Provider=Microsoft.Jet.OLEDB.4.0;
Data Source=C:\DatabasePath\;Extended Properties="\"text;
HDR=Yes;FMT=Delimited;\";"
```

Note: You must specify the filename in the SQL template statement... For example:

```
"Select Name, Address From Clients.txt"
```

Connecting to an Outlook 2000 personal mail box using the JET OLE DB Provider: (By J. Cardinal)

```
Connection string: "Provider=Microsoft.Jet.OLEDB.4.0;
Outlook 9.0;MAPILEVEL=;DATABASE=C:\Temp\;"
```

Replace *c: \temp* with any temporary folder. It will create a schema file in that folder when you open it which shows all the fields available. Blank `MAPILEVEL` indicates top level of folders).

Connecting to an Exchange mail box through JET: (By J. Cardinal)

```
Connection string: "Provider=Microsoft.Jet.OLEDB.4.0;Exchange 4.0;
MAPILEVEL=Mailbox - Pat Smith|;DATABASE=C:\Temp\;"
```

You must replace *c: \temp* with any temporary folder.

Replace *Pat Smith* with the name of the mail box and you must keep vertical pipe character `|` to indicate top level of folders. Place sub folder after vertical pipe if accessing specific folder.

Note: you can enter queries against the mail store just like a database... For example:

```
Sql template: SELECT Contacts.* FROM Contacts;"
```

For more information, see: [The Jet 4.0 Exchange/Outlook IISAM](#).

OLE DB Provider for ODBC Databases

If you want to connect with a Microsoft Access database:

```
Connection string: "Provider=MSDASQL;Driver={Microsoft Access Driver (*.mdb)};
Dbq=c:\DatabasePath\MyDatabase.mdb;Uid=MyUsername;Pwd=MyPassword;"
```

If you want to connect with a SQL Server database:

```
Connection string: "Provider=MSDASQL;Driver={SQL Server};Server=MyServerName;
Database=MyDatabaseName;Uid=MyUsername;Pwd=MyPassword;"
```

If you want to use DSN:

```
Connection string:
"Provider=MSDASQL;PersistSecurityInfo=False;Trusted_Connection=Yes;
Data Source=MyDSN;catalog=MyDatabase;"
```

For more information, see: [Microsoft OLE DB Provider for ODBC](#).

OLE DB Provider for OLAP

Microsoft® OLE DB for Online Analytical Processing (OLAP) is a set of objects and interfaces that extends the ability of OLE DB to provide access to multidimensional data stores.

```
Connection string: "Provider=MSOLAP;Data Source=MyOLAPServerName;  
Initial Catalog=MyOLAPDatabaseName;"
```

Connection using HTTP:

This feature enables a client application to connect to an Analysis server through Microsoft® Internet Information Services (IIS) by specifying a URL in the `Data Source` property in the client application's connection string. This connection method allows PivotTable® Service to tunnel through firewalls or proxy servers to the Analysis server. A special Active Server Pages (ASP) page, *Msolap.asp*, enables the connection through IIS. The directory in which this file resides must be included as part of the URL when connecting to the server (for example, *http://www.myserver.com/myolap/*).

Using a URL

```
Connection string: "Provider=MSOLAP;Data Source=http://MyOLAPServerName/  
Initial Catalog=MyOLAPDatabaseName;"
```

Using SSL

```
Connection string: "Provider=MSOLAP;Data Source=https://MyOLAPServerName/  
Initial Catalog=MyOLAPDatabaseName;"
```

For more information, see: [OLE DB for OLAP](#), [Connecting Using HTTP](#).

OLE DB Provider for Oracle OLE DB Provider for Oracle (from Microsoft)

The Microsoft OLE DB Provider for Oracle allows ADO to access Oracle databases.

```
Connection string: "Provider=MSDAORA;Data Source=MyOracleDB;  
User Id=myUsername; Password=myPassword;"
```

For more information, see: [Microsoft OLE DB Provider for Oracle](#).

OLE DB Provider for Oracle (from Oracle).

For Standard security:

```
Connection string: "Provider=OraOLEDB.Oracle;  
Data Source=MyOracleDB; User Id=myUsername;Password=myPassword;"
```

For a Trusted connection:

OS Authenticated connect setting user ID to "/":

```
Connection string: "Provider=OraOLEDB.Oracle;  
Data Source=MyOracleDB;User Id=/;Password=;"
```

OS Authenticated connect using OSAuthent:

```
Connection string: "Provider=OraOLEDB.Oracle;Data Source=MyOracleDB;OSAuthent=1;"
```

Note: "Data Source=" must be set to the appropriate Net8 name which is known to the naming method in use. For example, for Local Naming, it is the alias in the *tnsnames.ora* file; for Oracle Names, it is the Net8 Service Name.

For more information, see: [Oracle Provider for OLE DB Developer's Guide](#).

OLE DB Provider for Visual FoxPro

```
Connection string: "Provider=vfpoledb;  
Data Source=C:\DatabasePath\MyDatabase.dbc;"
```

For more information, see: [Microsoft OLE DB Provider for Visual FoxPro](#).

OLE DB Provider for Index Server (By Chris Maunder)

```
Connection string: "provider=msidxs;Data Source=MyCatalog;"
```

For more information, see: [Microsoft OLE DB Provider for Microsoft Indexing Service](#).

OLE DB Data Link Connections

Data Link File - UDL

```
Connection string: "File Name=c:\DataBasePath\DatabaseName.udl;"
```

If you want to create a Data Link File, you can make a new empty text file, change its extension by *.udl*, then double click over the file, and the operating system calls for you the Data Link app.

Source: <http://www.codeproject.com/database/connectionstrings.asp>

Appendix "H" - References

Product website

<http://www.ozeki.hu/ms6>

European Telecommunications Standards Institute

<http://www.etsi.org>

GSM World SMS page

<http://www.gsmworld.com/technology/sms/intro.shtml>