

Now SMS/MMS Gateway v5.0

<http://www.nowsms.com>

NOW SMS/MMS GATEWAY	1
CONFIGURING SMSC CONNECTIONS	3
GSM MODEMS	4
SMPP SMSC	9
HTTP SMSC	13
UCP/EMI SMSC	18
CIMD2 SMSC	22
ADDITIONAL SMSC CONFIGURATION OPTIONS	25
ROUTING OPTIONS	27
RUNNING AS A SERVICE	29
CONFIGURING THE WEB INTERFACE	31
DEFINING SMS USER ACCOUNTS	33
WEB MENU INTERFACE	38
SEND TEXT MESSAGE	40
SEND BINARY MESSAGE	41
<i>Send Nokia Ring Tone</i>	41
<i>Send Nokia CLI (Group) Icon</i>	42
<i>Send Nokia Operator Logo</i>	43
<i>Send Nokia Picture Message</i>	45
<i>Send Binary Message Other</i>	46
SEND WAP PUSH MESSAGE	47
SEND MMS MESSAGE	48
SEND WAP MULTIMEDIA MESSAGE	50
SEND WAP OTA SETTINGS	51
SEND WAP OTA BOOKMARK	53
SEND OMA OTA SETTINGS	54
SEND SYNCML OTA SETTINGS	55
SEND XML SETTINGS DOCUMENT	56
SEND VOICE MAIL NOTIFICATION	57
SENDING TEXT MESSAGES	58
SENDING BINARY MESSAGES	60
SENDING WAP PUSH MESSAGES	62
SENDING WAP OTA MESSAGES	64
WAP BOOKMARK OTA MESSAGES	64
WAP CONFIGURATION OTA MESSAGES	65
<i>WAP Configuration OTA – URL Parameters</i>	65
<i>WAP Configuration OTA – OTA Documents</i>	66
SENDING OMA PROVISIONING CONTENT OTA MESSAGES	68
SENDING MMS NOTIFICATIONS AND CONTENT	71
CREATING MMS MESSAGE FILES – MMSCOMP	72
MMSC MESSAGING SERVER	75
E-MAIL – MMS GATEWAY	81
SENDING VOICE MAIL NOTIFICATION MESSAGES	83

2-WAY SMS SUPPORT.....	85
2-WAY MMS SUPPORT	88
CONFIGURING MMS VASP ACCOUNTS	91
SUBMITTING MMS MESSAGES TO THE GATEWAY.....	95
<i>Now SMS/MMS Proprietary URL Submission</i>	<i>96</i>
<i>MM7</i>	<i>98</i>
<i>MM4</i>	<i>101</i>
<i>MM1</i>	<i>103</i>
<i>EAIF</i>	<i>104</i>
CONNECTING TO AN OPERATOR MMSC	105
CONNECTING TO AN OPERATOR MMSC – USING A GPRS MODEM	105
CONNECTING TO AN OPERATOR MMSC – SENDING MMS MESSAGES	111
CONNECTING TO AN OPERATOR MMSC – RECEIVING MMS MESSAGES.....	117
URL PARAMETERS FOR SENDING MESSAGES	119
TECHNICAL BULLETINS.....	124
NOW SMS/MMS ACCOUNTING CALLBACKS	125
<i>MMS Accouting Callbacks.....</i>	<i>125</i>
<i>SMS Accouting Callbacks.....</i>	<i>127</i>
NOW MMSC OPERATOR CONSIDERATIONS	129
<i>Configuring the WAP3GX WAP Gateway to Forward MSISDN.....</i>	<i>131</i>
2-WAY SMS RETURNING A NON-TEXT RESPONSE.....	134
USING PORT 80 ON A PC RUNNING IIS.....	137
PROVISIONING MMSC USER ACCOUNTS VIA HTTP.....	139
FASTER GSM MODEM SPEEDS WITH SMS OVER GPRS	140
ROUTING MMS NOTIFICATIONS VIA A WAP PUSH PROXY GATEWAY	142

Now SMS/MMS Gateway

The Now SMS/MMS Gateway is a fast track to deploying and developing SMS, MMS and WAP Push applications.

It is an easy-to-install SMS Gateway, MMS Gateway, WAP Push Gateway and Multimedia Messaging Center (MMSC) for Windows NT/2000/XP, but don't take our word for it, download the free 60-day trial version at <http://www.nowsms.com> and try it!

For additional technical information about the Now SMS/MMS Gateway, please visit our on-line discussion board at <http://www.nowsms.com/messages>.

The gateway includes the following features:

- ❖ Supports SMS connectivity via one or more GSM modems (or GSM phones connected to a PC serial port), or over TCP/IP connections using SMPP, UCP/EMI and/or HTTP protocols.
- ❖ Supports least cost routing with pattern matching to route messages to different SMS connections based on destination.
- ❖ Supports sending and receiving MMS messages either via direct SMS/WAP delivery with its built-in MMSC, or can interface to operator MMSCs using the MM1, MM4 (SMTP), MM7 (XML-SOAP based HTTP POST API) and/or EAIF (Nokia proprietary API) protocols. The MM1 interface can talk to an operator MMSC over a GPRS/GSM modem without a special operator account.
- ❖ Supports easy generation and delivery of MMS messages, and includes an MMS compiler for generating the binary headers and message formats required for MMS content.
- ❖ Includes a powerful MMSC for processing MMS messages independent of the operator gateway. The MMSC supports dynamic content adaptation and content conversion to help simplify the process of delivering MMS content to devices with varied characteristics. The MMSC also includes a built-in SMTP e-mail gateway for bi-directional exchange of messages between MMS compatible devices and internet e-mail recipients.
- ❖ Supports the MM1, MM4, MM7 and EAIF protocols to allow applications and Value Added Service Providers (VASPs) to send and receive MMS messages via the gateway.

- ❖ Supports Unicode (UTF-8) formats for both SMS and MMS messages, enabling deployment in multilingual environments.
- ❖ Supports easy generation and delivery of WAP Push messages, independent of the WAP gateway being used.
- ❖ Supports Multimedia WAP Push to simplify the delivery of multimedia objects and Java applications via WAP Push.
- ❖ Supports WAP OTA (“Over The Air”) configuration settings and bookmarks, including support for the Open Mobile Alliance (OMA) Provisioning Content v1.1.
- ❖ Supports 2-way SMS for interactive application development. SMS messages received by the gateway can trigger either an executable program to be run, or an HTTP request. Simple text responses back to the user can be returned as output of the request. More complex responses, including MMS or other binary SMS content, are also supported.
- ❖ Supports 2-way MMS for interactive multimedia application development. MMS messages received by the gateway are parsed into individual file components that can be easily processed by a user supplied tools. For example, received MMS images could be automatically posted to a web site.
- ❖ Includes an SMPP server, simplifying the process of connecting multiple gateways and applications.
- ❖ Provides an SMTP interface with SMTP Authentication support, allowing a user account to login via SMTP with an e-mail client to submit bulk delivery of SMS or MMS messages.
- ❖ Supports sending of other binary SMS formats, including EMS, ring tones, etc.
- ❖ Supports concatenated SMS for SMS text messages longer than 160 characters.
- ❖ Supports easy generation and delivery of new voice mail notification messages, simplifying the integration of office voice mail with mobile voice mail.

Configuring SMSC Connections

The Now SMS/MMS Gateway requires a connection to an SMSC (Short Messaging Service Centre) to interface with SMS and MMS networks. An SMSC connection can consist of one or more of the following:

GSM Modem - A GSM modem or phone connected to a PC serial port (or to a USB port with an appropriate modem driver).

SMPP (Short Message Peer to Peer Protocol) - A TCP/IP connection over the internet or a private network to a service that supports v3.3 or v3.4 of the SMPP protocol. (Note that the Now SMS/MMS Gateway also includes an SMPP server, which allows you to chain multiple gateways together.)

UCP/EMI (Universal Computer Protocol / External Machine Interface) - A TCP/IP connection over the internet or a private network to a service that supports v3.5 or v4.0 of the UCP/EMI protocol. UCP/EMI is primarily implemented by CMG SMSCs.

CIMD2 (Computer Interface to Message Distribution, version 2) - A TCP/IP connection over the internet to a service that supports the CIMD2 protocol. CIMD2 is implemented by Nokia SMSCs.

HTTP (Hyper Text Transport Protocol, e.g., the standard protocol for the "web") - A TCP/IP connection over the internet or private network to a service that accepts SMS messages via an HTTP "GET" based protocol.

GSM Modems

A GSM modem can be an external modem device, such as the Siemens MC35 or Wavecom FASTRACK external modems. Insert a GSM SIM card into this modem, and connect the modem to an available serial port on your computer.

A GSM modem can be a PC Card installed in a notebook computer, such as the Sierra Wireless Aircard 750.

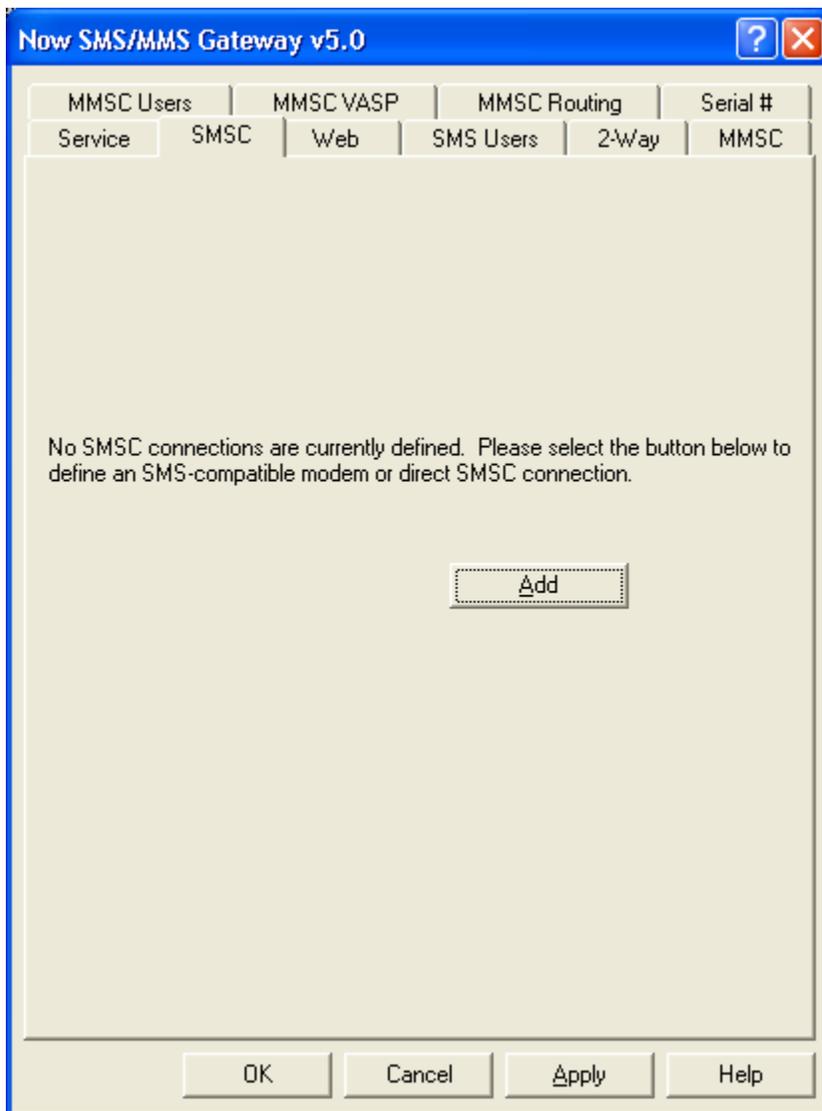
A GSM modem could also be a standard GSM mobile phone with the appropriate cable and software driver to connect to a serial port or USB port on your computer. Any phone that supports the "extended AT command set" for sending/receiving SMS messages, as defined in the ETSI GSM 07.05 Specification can be supported by the Now SMS/MMS Gateway.

A dedicated GSM modem (external or PC Card) is usually preferable to a GSM mobile phone. This is because of some compatibility issues that can exist with mobile phones. For example, if you wish to be able to receive inbound MMS messages with your gateway, most GSM phones will only allow you to send MMS messages. This is because the mobile phone automatically processes received MMS message notifications these messages, without forwarding them via the modem interface. Similarly some mobile phones will not allow you to correctly receive SMS text messages longer than 160 bytes (known as "concatenated SMS" or "long SMS"). This is because these long messages are actually sent as separate SMS messages, and the phone attempts to reassemble the message before forwarding via the modem interface. (We've observed this latter problem utilizing the Ericsson R380, while it does not appear to be a problem with many other Ericsson models.)

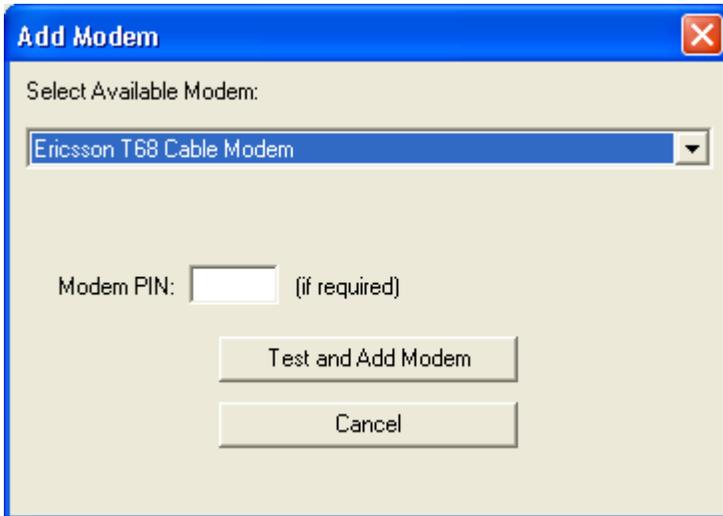
When you install your GSM modem, or connect your GSM mobile phone to the computer, be sure to install the appropriate Windows modem driver from the device manufacturer. To simplify configuration, the Now SMS/MMS Gateway will communicate with the device via this driver. If a Windows driver is not available for your modem, you can use either the "Standard" or "Generic" 19200 bps modem driver that is built into windows. A benefit of utilizing a Windows modem driver is that you can use Windows diagnostics to ensure that the modem is communicating properly with the computer.

The Now SMS/MMS gateway can simultaneously support multiple modems, provided that your computer hardware has the available communications port resources.

To define which modems are to be utilized by the gateway, select the "SMSC" tab from the gateway configuration dialog:

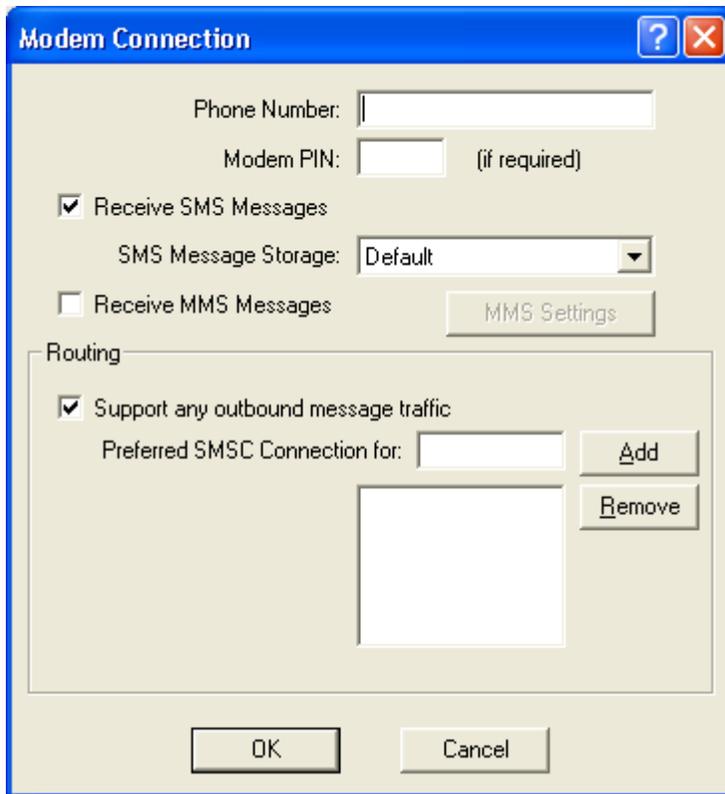


If no modems are yet to be defined, only the **“Add”** button will be available on this dialog. Select **“Add”**, and then **“GSM Phone or Modem”** to display a list of available modem drivers on your computer.



Select an available modem and press the **“Test and Add Modem”** button. The gateway will then attempt to initialize the modem, and confirm that the modem supports the necessary interfaces to send and receive SMS messages. The modem will only be added to the configuration if the gateway confirms that it can properly communicate with the modem. If the SIM card for the modem requires a 4-digit PIN, please supply it in the field provided.

After the Modem is added, there are additional Properties that can be configured for the modem connection. Highlight the modem name in the **“SMSC”** list, and press the **“Properties”** button:



The **“Phone Number”** field is used to tell NowSMS the phone number associated with this modem. It is not mandatory to supply a phone number, however if a phone number is specified, it will be possible to submit messages to the gateway in such a way that if multiple SMSC or GSM modem connections are defined, the message will be sent out via this connection. (This is possible by including a **“Sender”** parameter in a URL request to submit a message, where the value of this parameter matches the **“Phone Number”** field configured for a specific GSM modem. For more information, refer to [URL Parameter for Sending Messages](#).) When receiving SMS or MMS messages, the phone number will be supplied as the receiving address of the message, allowing applications to determine which modem received the message in an installation with multiple GSM modem connections.

A Modem PIN is a 4-digit code that is used for security purposes. If a PIN is configured on the SIM card installed in the modem, the phone or modem cannot be used until the PIN is entered. If you want the gateway to automatically supply the PIN to the modem upon startup, supply this PIN in the **“Modem PIN”** field.

If the Now SMS/MMS Gateway should process SMS messages received by the attached modem, the **“Receive SMS Messages”** setting should be enabled. The **“SMS Message Storage”** location should be left at **“Default”** unless otherwise instructed by technical support. For more information on how to process received SMS messages, please refer to [2-Way SMS Support](#).

If the Now SMS/MMS Gateway should process MMS messages received by the attached modem, the **“Receive MMS Messages”** setting should be enabled. The **“MMS Settings”** dialog will display a dialog with additional configuration settings that are required for enabling the gateway to be able to receive MMS messages from the operator network. Please note that a dedicated GSM modem device (not a phone acting as a modem) is required in most instances in order to support the receiving of MMS messages, and in most cases the modem device must also support GPRS. For additional configuration information on processing received MMS messages, please refer to [2-Way MMS Support](#).

SMPP SMSC

The gateway supports the SMPP (Short Message Peer to Peer) protocol, version 3.3 or 3.4, to connect to an SMSC over the internet or other private TCP/IP network. Some mobile operators provide SMPP connections for higher volumes of SMS traffic.

To add an SMPP connection, select **"Add"** from the **"SMSC"** tab of the gateway dialog. Then select **"SMPP over TCP/IP"**.



“**SMPP Version**” specifies the version of the SMPP protocol to use. The gateway supports “v3.3” and “v3.4”.

“**Server Host Name**” specifies the TCP/IP address or host name of the SMPP server.

“**Port**” specifies the TCP/IP port on the SMPP server to which the gateway should connect.

“**User Name**” specifies the user name (sometimes called System ID) for the gateway to use when connecting to the SMPP server.

“**Password**” specifies the password for the gateway to use when connecting to the SMPP server.

“**Address Range**” is a parameter used primarily when receiving messages. Set this field only if instructed to do so by your SMPP service provider.

“**System Type**” is an optional login parameter that should be set only if required by the SMPP server. The SMPP system administrator will provide this value, which when required, is usually a short text string.

“**Sender Address**” specifies the default sender address (phone number) to apply to outbound SMS messages. The SMPP server may override this setting.

Check “**Allow Sender Address Override**” if you want to allow messages submitted to the gateway to be able to specify a sender address. If this box is checked, and a sender address is present in a message being submitted to an SMPP based SMSC, the sender address in the message will be submitted to the SMSC. The SMPP server may override this setting.

Check the “**Receive SMS Messages**” box if you wish to receive messages from the SMPP server. When this box is checked, the gateway will connect to the SMPP server with two separate connections, one bound as a transmitter and the other bound as a receiver.

Check the “**Receive MMS Messages**” box if you will be receiving MMS Notification messages via the SMSC. Note that not all service providers support the routing of MMS notification messages via an SMSC connection. If MMS messages will be received via the SMSC connection, it is necessary to also configure additional MMS settings. The “**MMS Settings**” dialog will display a dialog with additional configuration settings that are required for enabling the gateway to be able to receive MMS messages from the operator network. For additional configuration information on processing received MMS messages, please refer to [2-Way MMS Support](#).

The “**Routing**” group of options is used when multiple SMSC connections are defined to the gateway. These options define what messages should be routed to this connection. The Routing options are common to SMPP, GSM Modem, UCP/EMI and HTTP connections, and are described in the [Routing Options](#) section.

When the “**OK**” button is selected, the gateway will attempt to connect to the SMPP server to verify the configuration information provided. Diagnostic information will be displayed if the connection fails. The connection will only be added to the configuration after a successful connection to the SMPP server.

Many SMPP SMSC providers will also tell you that you need to configure certain additional parameters in your SMPP software. You might be told to specify particular Source TON, Source NPI, Destination TON or Destination NPI values.

The Now SMS/MMS Gateway uses intelligent defaults for the TON and NPI values which are sufficient for 99.9% of SMPP connections.

The Source TON and NPI are settings that apply to the sender address that is associated with messages that are submitted by the Now SMS/MMS Gateway to the SMSC. NowSMS automatically sets the Source TON to "1" if the sender address is in international format (starts with a "+" character). If the sender address is not in international format, NowSMS automatically sets the Source TON to "0". In both cases, the Source NPI is set to "1". If the sender address is contains alphabetic characters and is not a valid phone number, the Source TON is set to "5", and the Source NPI is set to "0".

The Destination TON and NPI are settings that apply to the recipient addresses for messages that are submitted by the Now SMS/MMS Gateway to the SMSC. NowSMS automatically sets the Source TON to "1" if the sender address is in international format (starts with a "+" character). If the sender address is not in international format, NowSMS automatically sets the Source TON to "0". In both cases, the Source NPI is set to "1".

If it is necessary to adjust these TON and NPI settings, they can be adjusted via configuration settings in the SMSGW.INI file. Create or modify a section of the SMSGW.INI file with a [SMPP] section header, and the following settings are supported under this header:

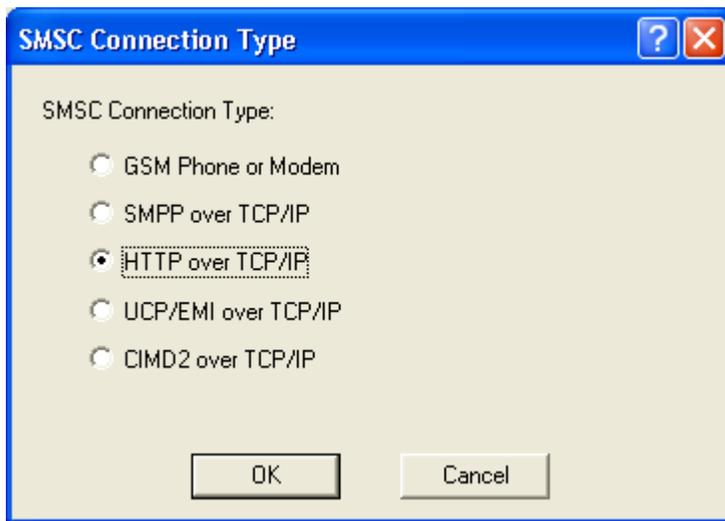
```
SourceTON=  
SourceNPI=  
DestTON=  
DestNPI=
```

If a TON or NPI value is explicitly set in the SMSGW.INI file, this value will be used in place of the automatic determination described above.

HTTP SMSC

The gateway supports the HTTP (Hyper Text Transport Protocol) protocol to connect to an SMSC over the internet or other private TCP/IP network. In addition to connecting to other services, this allows you to make connections to another Now SMS/MMS gateway, which can be useful for “least cost routing” of SMS messages between gateways over TCP/IP.

To add an HTTP connection, select “Add” from the “SMSC” tab of the gateway dialog. Then select “HTTP over TCP/IP”.



Host Name: smsc.example.only.com

Port Number: 8888

Use HTTP Proxy

Proxy Server:

User Name: example

Password: *****

Send login credentials using HTTP Authorization headers

Server Type: Now SMS

URL Template Text: Number@@&Text=@@Text@@

URL Template Binary: =@@pid@@&dcS=@@dcS@@

Sender Address: +447778001210 (optional)

Allow Sender Address Override

Remove '+' from Recipient phone number

Send long messages without segmentation

Routing

Support any outbound message traffic

Preferred SMSC Connection for:

“**Host Name**” specifies a DNS host name or TCP/IP address for connecting to the service.

“**Port Number**” specifies the HTTP port number to use when connecting to the service.

Check “**Use HTTP Proxy**” if the gateway must connect to the HTTP server via a proxy server, and supply the host name or TCP/IP address and port number of the proxy server in the “**Proxy Server**” field using a format of “host.name:9999”, where “host.name” is the DNS host name or TCP/IP address of the proxy server, and “9999” is the port number of the proxy server.

“**User name**” and “**Password**” specify a user account and password to use when connecting to the service.

“**Send login credentials using HTTP Authorization headers**” refers to how the user name and password information is sent to the HTTP server. The login information can either be sent as replaceable parameters in the URL request, or using the “HTTP Authorization” header. Check this box to use the “HTTP Authorization” header. This box should be checked when connecting with another Now SMS/MMS gateway.

“**Server Type**” provides pre-defined templates for connections to common gateway interfaces. If you are connecting to one of the servers with a pre-defined template, such as another Now SMS/MMS gateway, select its server type here. Otherwise, select “Custom” to define a custom template.

“**URL Template Text**” is a URL template that is used when sending text SMS messages. When the gateway has a text SMS message to send, it connects to the HTTP server and issues the URL request specified in this field, replacing the “replaceable parameters” with values for the message to be sent. A complete list of “replaceable parameters” is provided below.

“**URL Template Binary**” is a URL template that is used when sending binary SMS messages. When the gateway has a text SMS message to send, it connects to the HTTP server and issues the URL request specified in this field, replacing the “replaceable parameters” with values for the message to be sent. A complete list of “replaceable parameters” is provided below.

URL Template Replaceable Parameters:

@@UserName@@	The user name configured for this connection (optional)
@@Password@@	The password configured for this connection (optional)
@@PhoneNumber@@	The phone number of the recipient to receive this SMS message (required)
@@Text@@	The text of the SMS message (required for text messages)
@@Data@@	The data of the SMS message in binary format as a string of hexadecimal characters (either this or @@DataBin@@ required for binary messages)
@@DataBin@@	The data of the SMS message in binary format as the actual binary data in URL escaped format (either this or @@Data@@ required for binary messages)
@@UDH@@	The “User Data Header” of a binary message as a string of hexadecimal characters (either this or @@UDHBin@@ required for binary messages)

@@UDHBin@@	The “User Data Header” of a binary message in binary format as the actual binary data in URL escaped format (either this or @@UDH@@ required for binary messages)
@@PID@@	SMS “Protocol ID” field as a hexadecimal value
@@PIDdecimal@@	SMS “Protocol ID” field as a decimal value
@@DCS@@	SMS “Data Coding Scheme” field as a hexadecimal value
@@DCSdecimal@@	SMS “Data Coding Scheme” field as a decimal value.
@@Sender@@	Phone number to be included as the sender of this message.

“**Sender Address**” specifies the default sender address (phone number) to apply to outbound SMS messages. The SMSC to which you are connecting may override this setting. (Note: The sender number is only transmitted if the @@Sender@@ variable is included in your URL template string.)

Check “**Allow Sender Address Override**” if you want to allow messages submitted to the gateway to be able to specify a sender address. If this box is checked, and a sender address is present in a message being submitted to an HTTP based SMSC, the sender address in the message will be submitted to the SMSC. The SMSC may override this setting.

Check “**Remove ‘+’ from Recipient Phone Number**” if the gateway should remove the “+” character from international phone numbers before submitting the message to the HTTP SMSC. HTTP SMSC interfaces based upon the Kannel product expect the “+” character to be removed.

Some types of messages processed by the Now SMS/MMS Gateway may require multiple SMS messages to transmit a single logical message. This is because the maximum size of an SMS message is 160 text characters or 140 binary bytes of data. The Now SMS/MMS Gateway automatically segments larger messages and submits them as multiple SMS messages that can be reassembled by the receiving client. Some HTTP based SMSCs prefer to split larger messages themselves. Check “**Send long messages without segmentation**” if you want the SMSC to split larger messages into multiple SMS messages, or leave this setting unchecked to allow the Now SMS/MMS Gateway to perform necessary segmentation of large messages.

The “**Routing**” group of options is used when multiple SMSC connections are defined to the gateway. These options define what messages should be routed to this connection. The Routing options are common to SMPP, GSM Modem, UCP/EMI and HTTP connections, and are described in the [Routing Options](#) section.

When the “OK” button is selected, the gateway will attempt to connect to the HTTP server to verify the configuration information provided. Diagnostic information will be displayed if the connection fails. The connection will only be added to the configuration after a successful connection to the HTTP server. Note that although a connection attempt was successful, you should attempt to send a message through the interface to verify that the URL templates are defined correctly.

UCP/EMI SMSC

The gateway supports the UCP/EMI (Universal Computer Protocol / External Machine Interface) protocol, version 3.5 or 4.0, to connect to an SMSC over the internet or other private TCP/IP network. Some mobile operators provide UCP connections for higher volumes of SMS traffic.

To add a UCP/EMI connection, select “Add” from the “SMSC” tab of the gateway dialog. Then select “UCP/EMI over TCP/IP”.



UCP/EMI Connection

Server Host Name:

Server Port:

User Name:

Password:

Local Port: (optional)

Sender Address:

Allow Sender Address Override

Keep-alive: seconds (blank=disable)

Receive SMS Messages

Receive Port: (optional)

Receive MMS Messages

Routing

Support any outbound message traffic

Preferred SMSC Connection for:

“**Server Host Name**” specifies the TCP/IP address or host name of the UCP/EMI server.

“**Server Port**” specifies the TCP/IP port on the UCP/EMI server to which the gateway should connect.

“**User Name**” specifies the user name for the gateway to use when connecting to the UCP/EMI server.

“**Password**” specifies the password for the gateway to use when connecting to the UCP/EMI server.

Note: Some UCP/EMI systems may not require a username and password, and validate your account based only on the TCP/IP address of your system. If the username and password fields are left blank, the gateway will not send a UCP-60 bind message to the server to login.

“Local Port”, if set, should be set to the value of a TCP/IP port number on the gateway PC. If set, the gateway will initiate all connections to the UCP/EMI server from this port.

“Sender Address” specifies the default sender address (phone number) to apply to outbound SMS messages. The UCP/EMI server may override this setting.

Check **“Allow Sender Address Override”** if you want to allow messages submitted to the gateway to be able to specify a sender address. If this box is checked, and a sender address is present in a message being submitted to an UCP/EMI based SMSC, the sender address in the message will be submitted to the SMSC. The UCP/EMI server may override this setting.

Some UCP/EMI servers may require that clients submit keep-alive messages to the server every so many seconds or minutes, or the server will time out the connection. To enable keep-alive messages, specify a value in seconds for the **“Keep-Alive”** setting. (Note: The gateway uses a UCP-31 message for the keep-alive message.)

Check the **“Enable Receive Messages”** box if you wish to receive messages from the UCP/EMI server. When this box is checked, the gateway can either receive messages using a single connection to the UCP/EMI server, or the **“Receive Port”** setting can specify a TCP/IP port number on the gateway PC that will listen for connections from the UCP/EMI server, and receive messages. (Note: The **“Receive Port”** should be left blank for most configurations. This setting should only be specified if the UCP/EMI service provider will initiate connections to your server when it has a message to deliver, which is a rare configuration. Most service connections require that your server initiate all connections to the service provider, and the **“Receive Port”** setting is not used in those configurations.)

Check the **“Receive MMS Messages”** box if you will be receiving MMS Notification messages via the SMSC. Note that not all service providers support the routing of MMS notification messages via an SMSC connection. If MMS messages will be received via the SMSC connection, it is necessary to also configure additional MMS settings. The **“MMS Settings”** dialog will display a dialog with additional configuration settings that are required for enabling the gateway to be able to receive MMS messages from the operator network. For additional configuration information on processing received MMS messages, please refer to [2-Way MMS Support](#).

The **“Routing”** group of options is used when multiple SMSC connections are defined to the gateway. These options define what messages should be routed to this connection. The Routing options are common to SMPP, GSM Modem,

UCP/EMI and HTTP connections, and are described in the [Routing Options](#) section.

When the “**OK**” button is selected, the gateway will attempt to connect to the UCP/EMI server to verify the configuration information provided. Diagnostic information will be displayed if the connection fails. The connection will only be added to the configuration after a successful connection to the UCP/EMI server.

CIMD2 SMSC

The gateway supports the CIMD2 (Computer Interface to Machine Distribution, version 2) protocol to connect to an SMSC over the internet or other private TCP/IP network. This protocol is implemented by Nokia SMSCs.

To add a CIMD2 connection, select **"Add"** from the **"SMSC"** tab of the gateway dialog. Then select **"CIMD2 over TCP/IP"**.



“**Server Host Name**” specifies the TCP/IP address or host name of the CIMD2 server.

“**Server Port**” specifies the TCP/IP port on the CIMD2 server to which the gateway should connect.

“**User Name**” specifies the user name for the gateway to use when connecting to the CIMD2 server.

“**Password**” specifies the password for the gateway to use when connecting to the CIMD2 server.

“**Sender Address**” specifies the default sender address (phone number) to apply to outbound SMS messages. The CIMD2 server may override this setting.

Many CIMD2 connections allocate multiple phone numbers to an individual SMSC account. The “**Sender Address Prefix**” setting specifies the prefix associated with all phone numbers allocated to the SMSC account.

Check **“Allow Sender Address Override”** if you want to allow messages submitted to the gateway to be able to specify a sender address. If this box is checked, and a sender address is present in a message being submitted to a CIMD2 based SMSC, the sender address in the message will be submitted to the SMSC. The CIMD2 server may override this setting.

Some CIMD2 servers may require that clients submit keep-alive messages to the server every so many seconds or minutes, or the server will time out the connection. To enable keep-alive messages, specify a value in seconds for the **“Keep-Alive”** setting.

Check the **“Enable Receive Messages”** box if you wish to receive messages from the CIMD2 server.

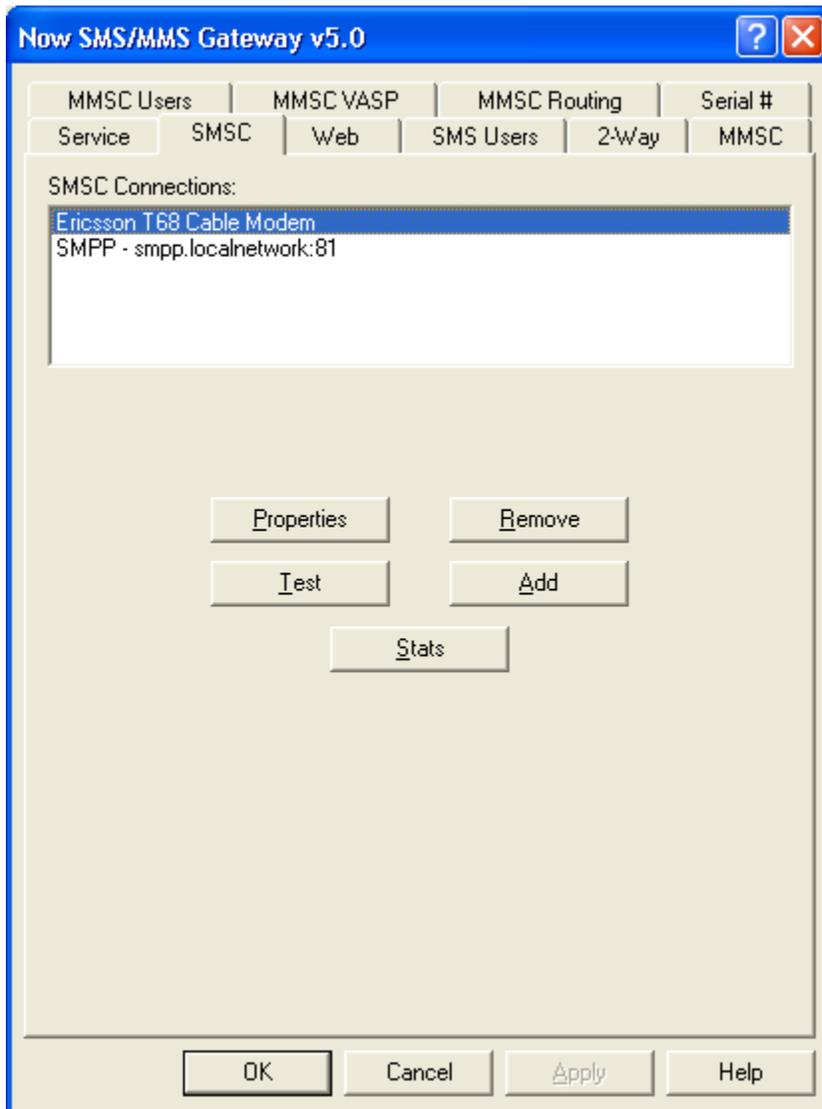
Check the **“Receive MMS Messages”** box if you will be receiving MMS Notification messages via the SMSC. Note that not all service providers support the routing of MMS notification messages via an SMSC connection. If MMS messages will be received via the SMSC connection, it is necessary to also configure additional MMS settings. The **“MMS Settings”** dialog will display a dialog with additional configuration settings that are required for enabling the gateway to be able to receive MMS messages from the operator network. For additional configuration information on processing received MMS messages, please refer to [2-Way MMS Support](#).

The **“Routing”** group of options is used when multiple SMSC connections are defined to the gateway. These options define what messages should be routed to this connection. The Routing options are common to SMPP, GSM Modem, UCP/EMI and HTTP connections, and are described in the [Routing Options](#) section.

When the **“OK”** button is selected, the gateway will attempt to connect to the UCP/EMI server to verify the configuration information provided. Diagnostic information will be displayed if the connection fails. The connection will only be added to the configuration after a successful connection to the UCP/EMI server.

Additional SMSC Configuration Options

Once one or more modems are defined, additional options are displayed under the “SMSC” tab.



The “**Properties**” button allows you to configure properties for the selected connection. For SMSC connections other than GSM modems, the standard configuration dialogs appear. For modem connections, the [Routing Options](#) dialog will be displayed, defining what SMS message recipients should be routed via this connection.

The “**Test**” button allows you to test the selected connection and confirm that the gateway software is still able to communicate properly with the modem or

service. Note that if the gateway service is currently active, the service will be temporarily stopped while the test is running.

The **“Remove”** button allows you to remove the selected connection from the configuration, so that the gateway will no longer attempt to use the connection.

After changes are made, use the **“Apply”** button to save any changes. Use the **“Ok”** button to close the dialog.

Routing Options

The “**Routing Options**” dialog for a modem connection contains a field to set the **Phone Number** of the GSM modem for that connection. This field is not required, but if the field is set, then the following will be enabled:

- ❖ INBOUND MESSAGES ARRIVING ON THAT GSM MODEM CONNECTION WILL BE ASSOCIATED WITH THIS PHONE NUMBER. FOR 2-WAY SMS APPLICATIONS, THE PHONE NUMBER IS INCLUDED IN THE @@RECIP@@ PARAMETER. ADDITIONALLY, ALL INBOUND MESSAGES ARRIVING ON THIS INTERFACE CAN BE QUEUED FOR DELIVERY TO AN SMPP CLIENT ACCOUNT THAT USES THIS GATEWAY AS ITS SMPP SERVER.
- ❖ WHEN A SENDER ADDRESS IS INCLUDED IN A MESSAGE SUBMITTED TO THE GATEWAY FOR DELIVERY, AND THE SENDER ADDRESS MATCHES THE CONFIGURED PHONE NUMBER FOR A PARTICULAR GSM MODEM CONNECTION, THE GATEWAY WILL ENSURE THAT THE MESSAGE IS SENT VIA THIS GSM MODEM.

“**Routing Options**” are available for all SMSC connections defined to the gateway. For SMPP, UCP/EMI and HTTP connections, the “**Routing**” group of options is displayed on the standard configuration dialog. For modem connections, a separate dialog is displayed when “**Properties**” is selected for the connection.

The screenshot shows the "Modem Connection" dialog box. It features a blue title bar with a question mark and a close button. The main area is light beige and contains the following elements:

- Phone Number:** A text input field.
- Modem PIN:** A text input field with the text "(if required)" to its right.
- Receive SMS Messages:** A checked checkbox.
- SMS Message Storage:** A dropdown menu currently showing "Default".
- Receive MMS Messages:** An unchecked checkbox.
- MMS Settings:** A button located to the right of the "Receive MMS Messages" checkbox.
- Routing:** A section containing:
 - Support any outbound message traffic:** A checked checkbox.
 - Preferred SMSC Connection for:** A list box that is currently empty.
 - Add:** A button to the right of the list box.
 - Remove:** A button to the right of the list box.
- OK:** A button at the bottom left.
- Cancel:** A button at the bottom right.

If **“Support any outbound message traffic”** is checked, this connection is available to route any messages, unless the recipient of the message appears as a **“Preferred SMSC Connection”** for another connection.

If **“Support any outbound message traffic”** is not checked, this connection will only route messages when the recipient of the message appears in the **“Preferred SMSC Connection”** list for this connection.

The **“Preferred SMSC Connection”** list specifies one or more patterns to match to determine if a message should be routed by a particular connection. Patterns consist of a phone number string, and can include the wildcard characters **“*”** and **“?”**. The **“*”** character matches any number of characters, and the **“?”** character matches any single character. When a pattern is defined on a preferred connection list, it means that any messages to recipients that match this pattern will be routed **ONLY** by this connection (unless another connection shares the same preferred connection pattern).

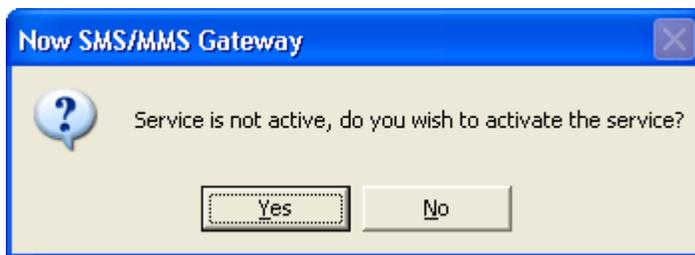
In the example above, the connection will route any messages to recipients in the **“+44”** country code (because **“+44*”** is on the preferred SMSC connection list for this connection). Also, this connection will route messages for any other recipients where the recipient does **NOT** match a pattern on the preferred list for any other connection.

In the example above, any other defined connections would not route messages for recipients in the **“+44”** country code, unless **“+44*”** was repeated on the preferred connection list for another connection.

Running as a Service

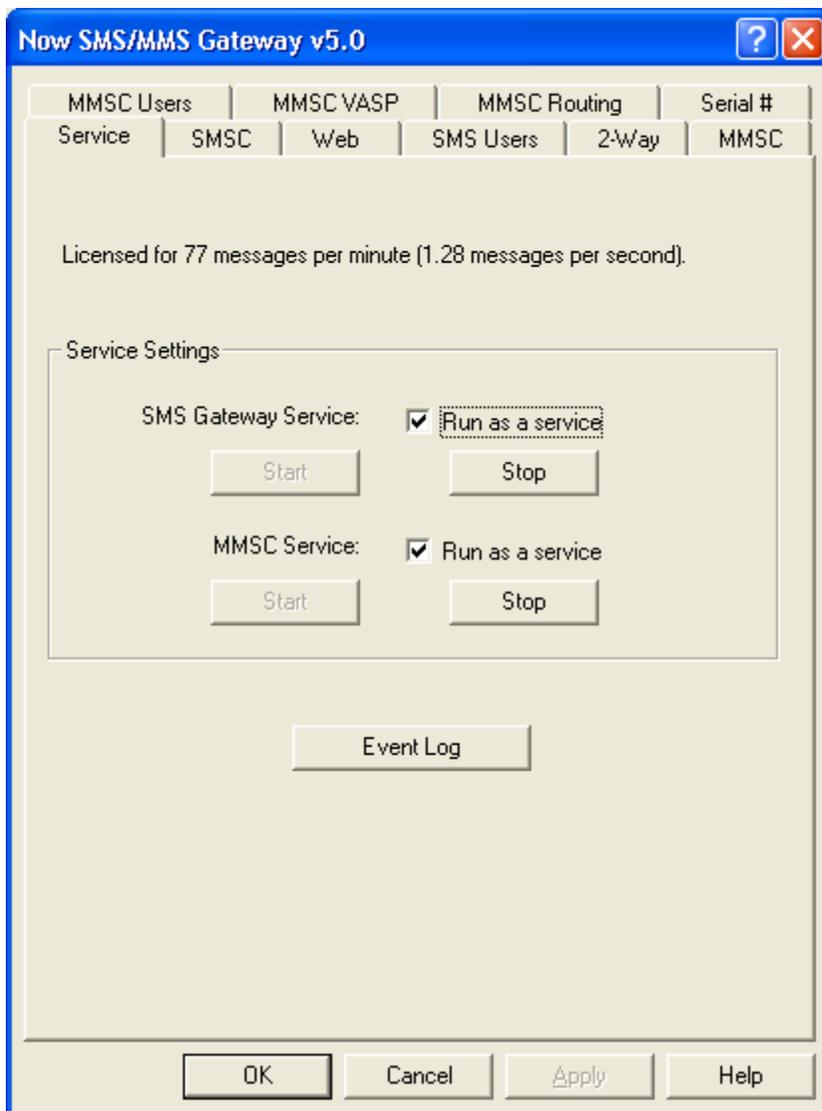
In the Windows environment, service processes are started automatically when the PC is started, so that it is not necessary for a user to logon to the computer to run a service program. The Now SMS/MMS gateway operates as a Windows service.

When you are configuring the gateway, and select “Ok” to close the configuration dialog, the gateway checks to see if the service process is active. If the service process is not active, the following dialog is displayed:



Select “Yes” to install and activate the service, or “No” to exit without activating the service.

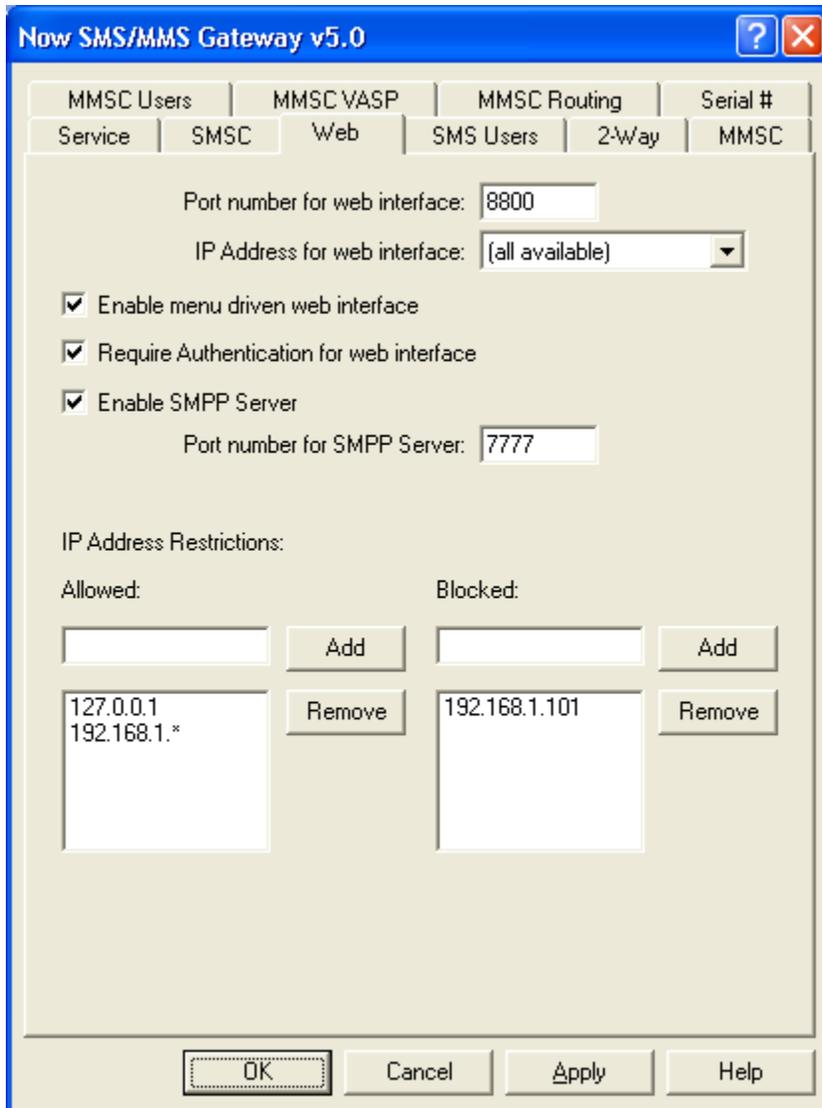
The “Service” tab in the configuration dialog can also be utilized to install or remove the gateway services.



The “Event Log” button displays a dialog that shows events related to the gateway which have been logged to the Windows Event Log. The gateway always logs when its services are started and stopped. Additionally errors are reported to the event log if a particular SMSC interface fails to initialize, and when a user account exceeds their defined message quota for sending messages.

Configuring the Web Interface

When you wish to submit any type of SMS message, including MMS notifications or WAP push messages, you must submit the request via the gateway's web interface. The "Web" tab of the configuration dialog contains settings relevant to this web interface.



You configure the gateway to listen for web/HTTP requests on a particular TCP/IP port number. The standard web port on the internet is 80, however you will most likely want to use a non-standard port for this service. The service defaults to the non-standard port 8800.

The PC that is running the gateway might have other web services installed. For this reason, the gateway allows you to specify which of the available IP addresses on the current PC should be used by the gateway. The **“IP Address for web interface”** prompt displays the available IP addresses on the current PC. To make the gateway service available via any address on the current PC, select **“(all available)”**, otherwise select a specific IP address.

“Enable menu driven web interface” specifies whether or not an HTML menu will be displayed when a user connects to the gateway via the web interface. This menu interface can be helpful when you are first exploring the features of the gateway. If this menu setting is disabled, then the gateway will require the appropriate URL parameters to perform any tasks. These URL parameters are defined elsewhere in this document.

Checking **“Enable SMPP Server”** enables the SMPP Server module of the gateway. While the gateway has the ability to act as an SMPP client, sending and receiving messages through an external SMPP server, it also has the ability to act as an SMPP server to provide message sending and receiving services to other SMPP clients. When you enable the SMPP server, you must specify a TCP/IP **“Port number for SMPP Server”**. This is a local port number on the gateway PC, which must not be in use by any other applications. The gateway will listen for SMPP clients to connect to the gateway on the port specified. User accounts for SMPP clients are defined on the **“[SMS Users](#)”** tab of the configuration dialog.

To prevent unauthorized access to the gateway, access should be limited. For most applications, it is recommended that this gateway be installed inside of your firewall, to help ensure that users on external computers cannot access the gateway. As further authorization measures, the gateway can also limit access by restricting address to a limited range of IP addresses, and by requiring a username and password for access.

To define that a username and password be required for access to the gateway, check **“Require Authentication for web interface”**. HTTP requests must include this username and password in order to issue requests to the gateway. (Note: A standard web browser will prompt for the username and password.) The **“[SMS Users](#)”** tab defines user accounts that can access the gateway.

To restrict access to the gateway to a limited range of IP addresses, **“Allowed”** and **“Blocked”** lists may be defined. When the gateway receives a new web request, it consults the **“Allowed”** and **“Blocked”** lists to determine if web access is allowed from the IP address of the machine that issued the request.

If an address is listed on the **“Blocked”** list, access will be denied, and the web interface cannot be used to submit an SMS message from that address.

If an address is not listed on the “Blocked” list, and an “Allowed” list is not defined, the web interface can be used to submit an SMS message from that address.

If an address is not listed on the “Blocked” list, and an “Allowed” list is defined, access will be denied if the address is not included in the “Allowed” list.

To add addresses to either list, enter an address in the appropriate text edit area and press the “Add” button. To remove an address from either list, highlight the appropriate address in the list, and press the “Remove” button. Wildcards can be used in a defined IP address to specify that any value in that portion of the IP address will be considered a match. (For example, 192.168.1.* would be considered a match with both 192.168.1.1 and 192.168.1.100.)

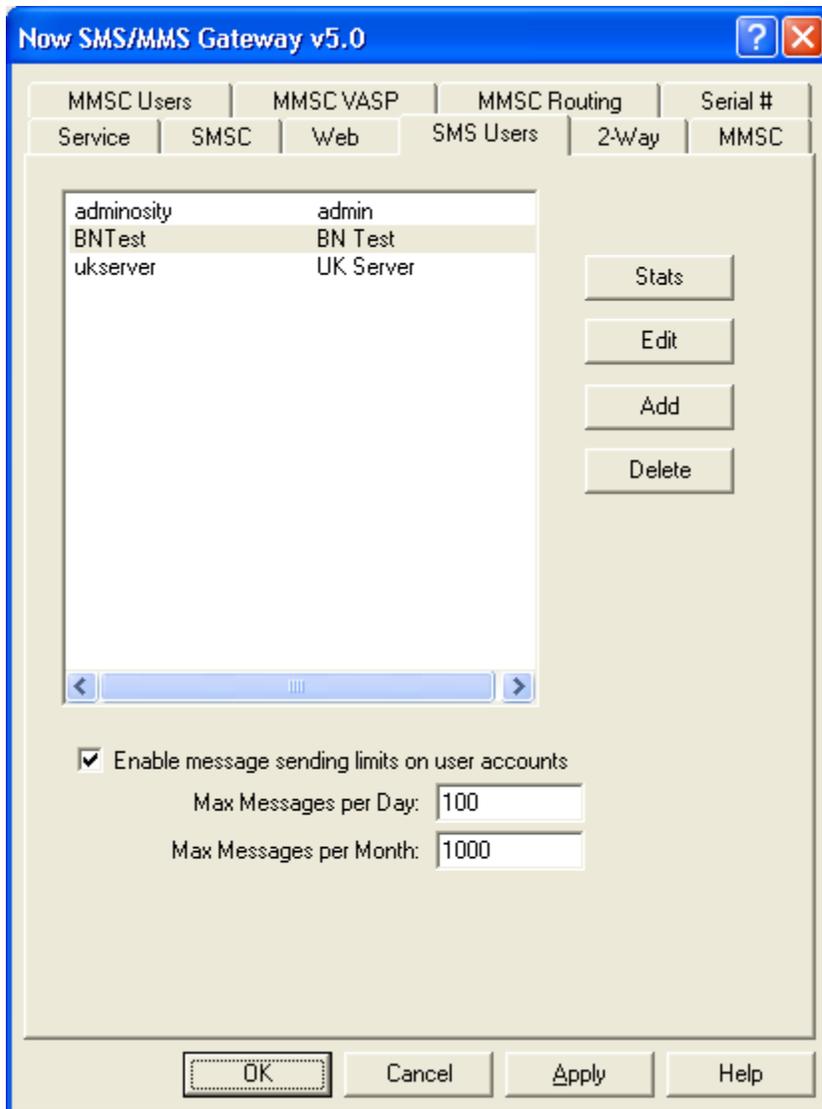
Defining SMS User Accounts

The “**SMS Users**” tab of the configuration dialog defines user accounts that are allowed to submit SMS and MMS messages through the gateway. Messages can be submitted to the gateway via HTTP (web interface), SMPP and/or SMTP (e-mail).

(Note: Mobile phone users that connect to the MMSC to send and receive MMS messages are configured under the “MMSC Users” dialog tab.)

When SMS user accounts are defined to the gateway, it is possible to define limits on the number of messages that an account is allowed to submit per day, and per month. It is also possible to define how the account is allowed to connect to the gateway, that is to say whether the account can login via HTTP, SMPP and/or SMTP.

The “**SMS Users**” dialog tab lists defined user accounts.



If you want to enable sending limits on your user accounts, you must check **“Enable message sending limits on user accounts”**, and specify a default limit for the maximum number of messages per day and per month, that users will be allowed to submit.

Individual user accounts can be allowed to have limits higher or lower than the default settings. However, you must enable the sending limits, and define default limits, in order to be able to define sending limits on any of the accounts defined to the gateway. After enabling message sending limits, click **“Apply”** to save the change before adding or editing any user accounts.

The **“Stats”** button displays information about the number of messages sent by the account that is selected in the list.

The image shows a dialog box titled "User Statistics" with a blue header bar containing a question mark icon and a close button. The main area is a table with a light beige background. The table lists dates from November 14, 2002, to November 20, 2002, along with the number of messages sent on each day. The last row, "TOTAL - November 2002", is highlighted in blue. Below the table is an "OK" button.

November 14, 2002	2 messages
November 15, 2002	2 messages
November 16, 2002	4 messages
November 18, 2002	18 messages
November 19, 2002	60 messages
November 20, 2002	22 messages
TOTAL - November 2002	108 messages

The **"Edit"** button is used to edit the settings for a defined user account. The **"Add"** button is used to define a new user account. The **"Delete"** button is used to delete a user account.

When adding or editing a user account, the following dialog will be displayed:

“User Name” and “Password” specify the user name and password that will be used to login to the account before sending any messages.

“Full Name” specifies a descriptive name for the account.

If the user account should be allowed to log into the web (HTTP) interface to submit messages, check “Enable Web Login for this user”.

If the user account should be allowed to connect as an SMPP client to the gateway’s SMPP server, check “Enable SMPP Login for this user”. In addition to allowing the SMPP client to send messages through the gateway, the gateway can also route received messages back to the SMPP client. To enable received messages to be routed to the SMPP client, check “Route received messages to user via SMPP”, and specify one or more phone numbers (separate multiple phone numbers with a comma), where if the recipient of a message received by the gateway matches this phone number, the message will be queued for delivery to this SMPP client.

If the user account should be allowed to log into the SMTP interface to submit messages using an e-mail client, check **“Enable SMTP Login for this user”**. This allows a user account to login via SMTP with an e-mail client to submit bulk delivery of SMS or MMS messages. The gateway uses the SMTP server from its built-in MMSC to provide this functionality. Therefore the [MMSC](#) must be configured and activated to enable this capability. When sending messages in this fashion, the sender must configure an e-mail client to connect to the gateway as an SMTP server, and to use SMTP authentication to login with the defined user name and password. The gateway does not provide an e-mail inbox, only outbound message sending via an SMTP interface. An authenticated SMTP user can send an MMS message by addressing the message to “`phonenumber@mms.domain.name`”, where “`mms.domain.name`” is the “Domain Name for MMS E-Mail” defined on the [MMSC configuration dialog](#). An authenticated SMTP user can send an SMS message by addressing the message to “`phonenumber@sms.domain.name`”, where “`sms.domain.name`” is the “Domain Name for SMS E-Mail” defined on the [MMS configuration dialog](#).

If message sending limits are enabled, check **“Use Default Message Sending Limits for this User”** to use the default limits defined for the system, or uncheck this setting to specify a maximum number of messages per day and per month that are allowed to be sent by this account.

It is also possible to define a **“Credit Balance”** for each account. This balance specifies a fixed number of messages that the account is allowed to send. Each time the account sends a message to a recipient, a credit is deducted from this balance. To add or remove credits, enter the number of credits in the **“Credits to add”** field and press **“Ok”**. (Prefix the number with a minus symbol, -, to remove credits from an account.)

If the account should only be allowed to connect from a limited set of IP addresses, a comma delimited list of IP addresses from which the account is allowed to login can be entered in the **“Restrict to IP Address(es)”** field. Addresses can include a **“*”** character as a wildcard to allow connections from all addresses within a subnet.

If all messages submitted by this account should have a particular sender address associated with the message, this sender address can be automatically applied to all messages submitted by the account by specifying a sender address (usually a phone number, but sometimes alphanumeric values are supported by SMSCs) in the **“Forced Sender Address”** field.

Web Menu Interface

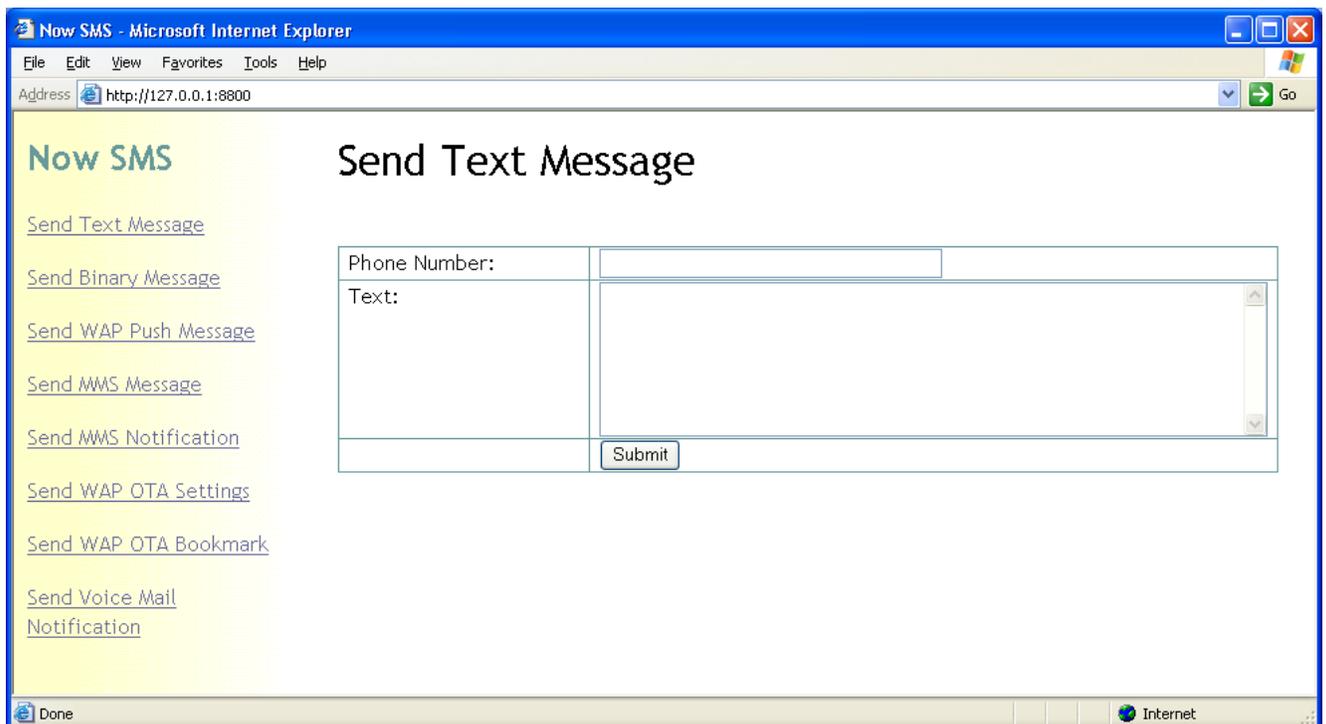
When the menu driven web interface is enabled, it is easy to test the ability of sending various types of SMS messages.

To enable the menu driven web interface of the gateway, you must check "Enable menu driven web interface" on the "Web" page of the configuration dialog. When that option is enabled, you can connect to the web interface with a web browser. On the "Web" page of the configuration dialog, there is a setting named "Port number for web interface". To connect to the web interface of the gateway, connect to `http://ip.address:port`, where "ip.address" is the IP address or host name of the PC running the gateway, and "port" is the port number specified for the web interface.

In a default configuration, the web menu interface can be accessed on the gateway PC by pointing a web browser to <http://127.0.0.1:8800>.

For more information on configuring the Web Menu Interface, see [Configuring the Web Interface](#).

With a web browser, connect to the web port configured for the SMS gateway, and an interface similar to the following will be displayed.



This web page provides a menu driven interface for sending various types of SMS and MMS messages.

For information on how to send specific types of messages, please refer to the appropriate section below:

- ❖ [Send Text Message](#)
- ❖ [Send Binary Message](#)
- ❖ [Send WAP Push Message](#)
- ❖ [Send MMS Message](#)
- ❖ [Send MMS Notification](#)
- ❖ [Send WAP Multimedia Message](#)
- ❖ [Send WAP OTA Settings](#)
- ❖ [Send WAP OTA Bookmark](#)
- ❖ [Send OMA OTA Settings](#)
- ❖ [Send SyncML OTA Settings](#)
- ❖ [Send XML Settings Document](#)
- ❖ [Send Voice Mail Notification](#)

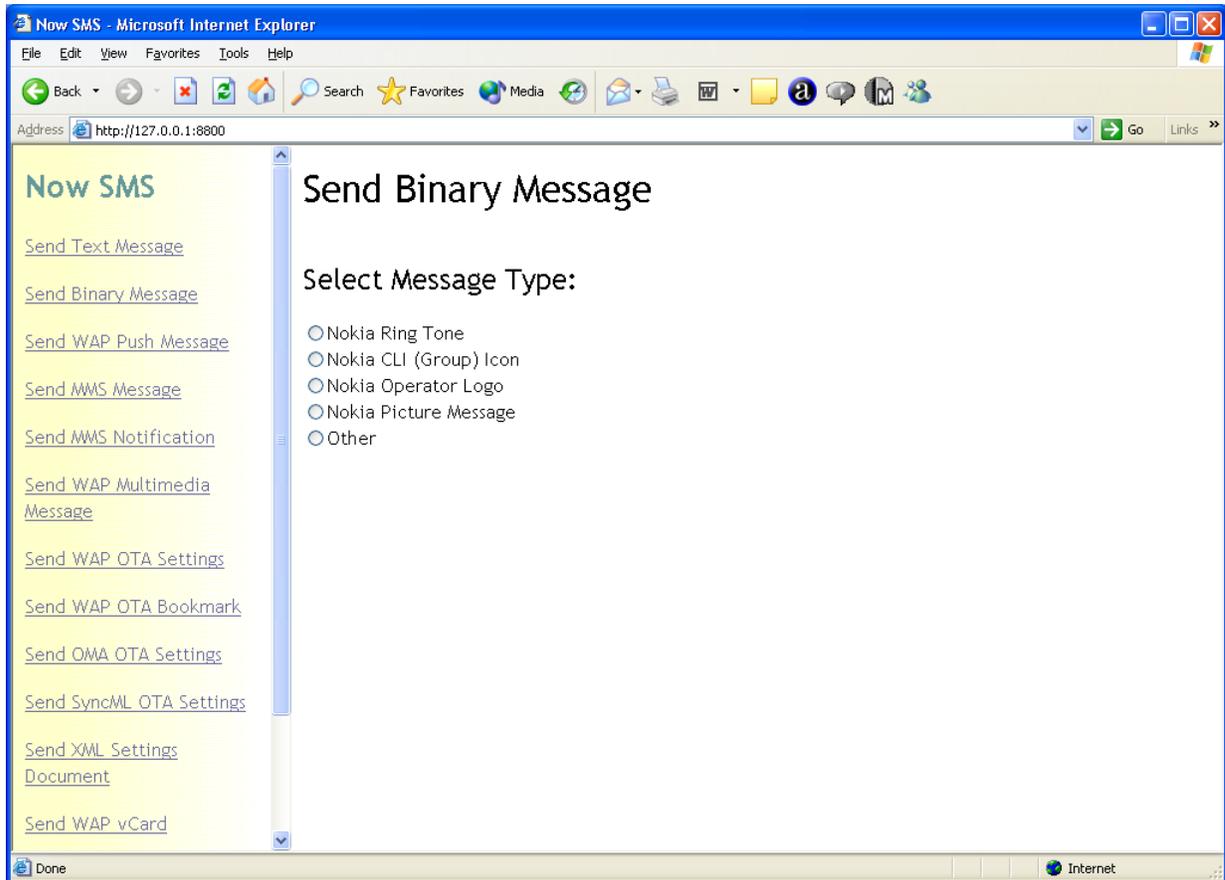
Send Text Message

Phone Number:	<input type="text"/>
Text:	<input type="text"/>
Message Type:	<input checked="" type="radio"/> Normal
	Replacement Type: <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7
	<input type="button" value="Submit"/>

To send a text message, simply enter a phone number and the text of your message. If the message is longer than 160 characters, the gateway will automatically use concatenated SMS (“long SMS”) message support to send the entire message.

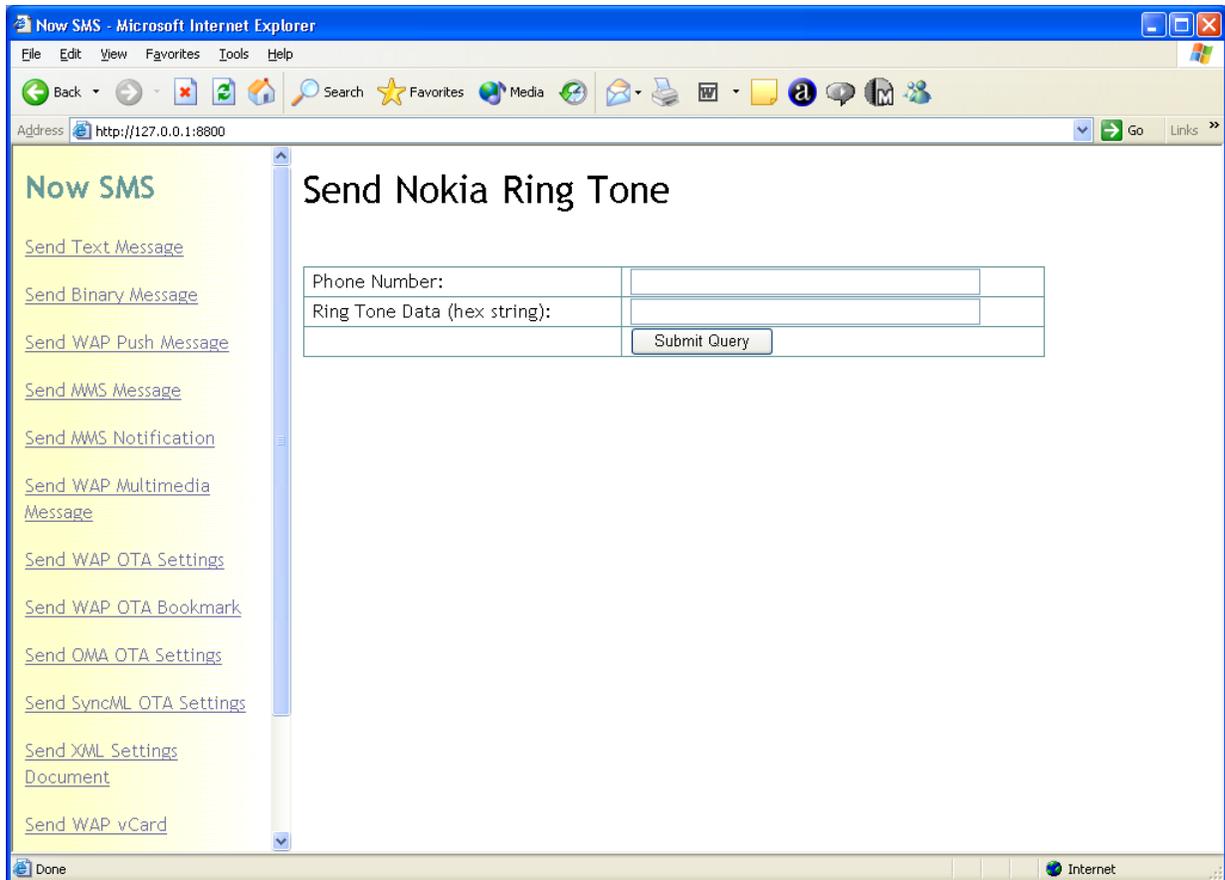
The “Message Type” would normally be set to “Normal”. Setting a “Replacement Type” value means that if the gateway sends a subsequent message with the same replacement type value, this will replace any previous messages that were sent by the same sender with the same replacement type value.

Send Binary Message



Sending a binary message through the web interface typically requires more knowledge of the binary SMS protocol that you are attempting to use. HTML forms are included for simplifying the process of sending Nokia Smart Messaging types, along with a general form for sending any binary message.

Send Nokia Ring Tone



To send a Nokia ring tone, you must have a hex string value for the ring tone data. The hex string format represents two characters for each binary byte of ring tone data. Documentation of the ring tone data format is beyond the scope of this document.

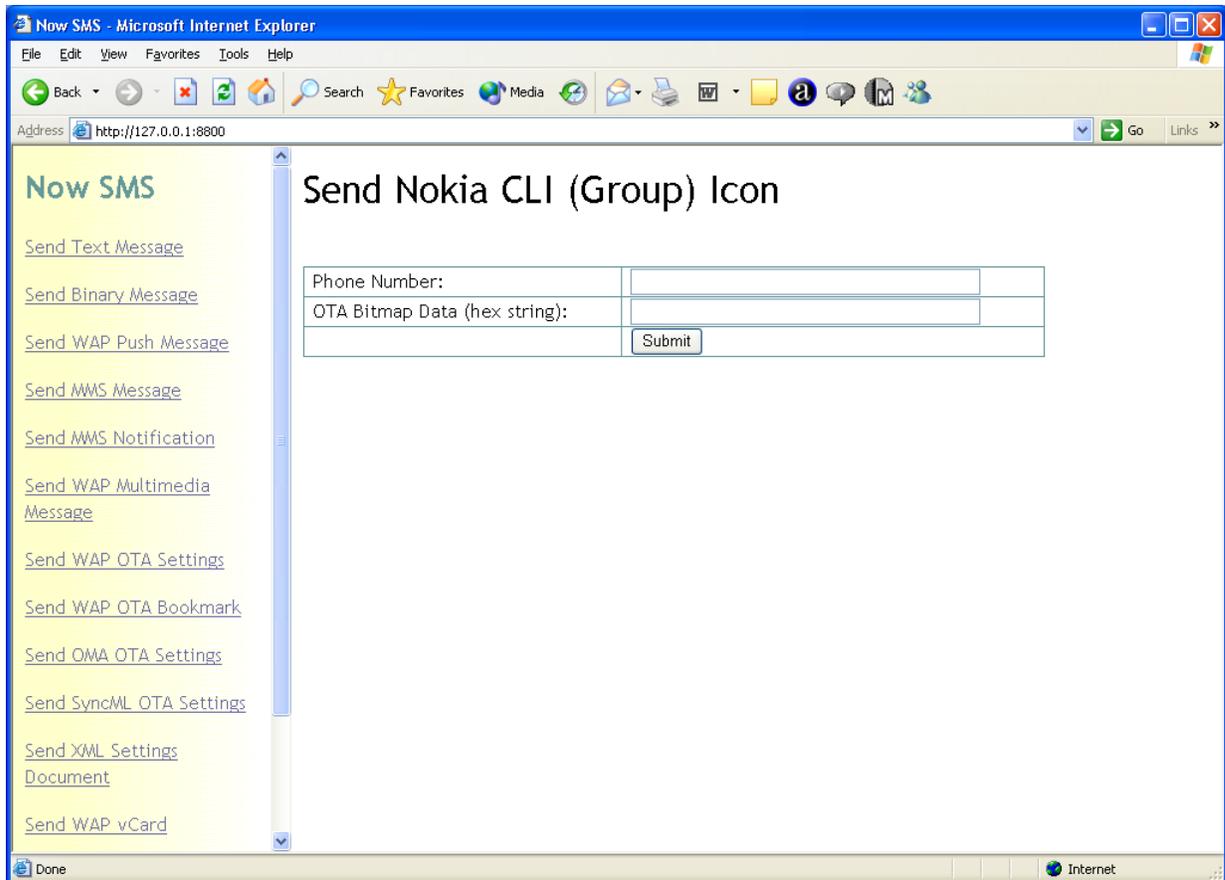
For those who wish to send ring tones programmatically via the Now SMS/MMS Gateway, note that this form includes the following hidden fields which are included as URL parameters when submitting the message to the server:

UDH = 06050415811581

PID = 0

DCS = F7

Send Nokia CLI (Group) Icon



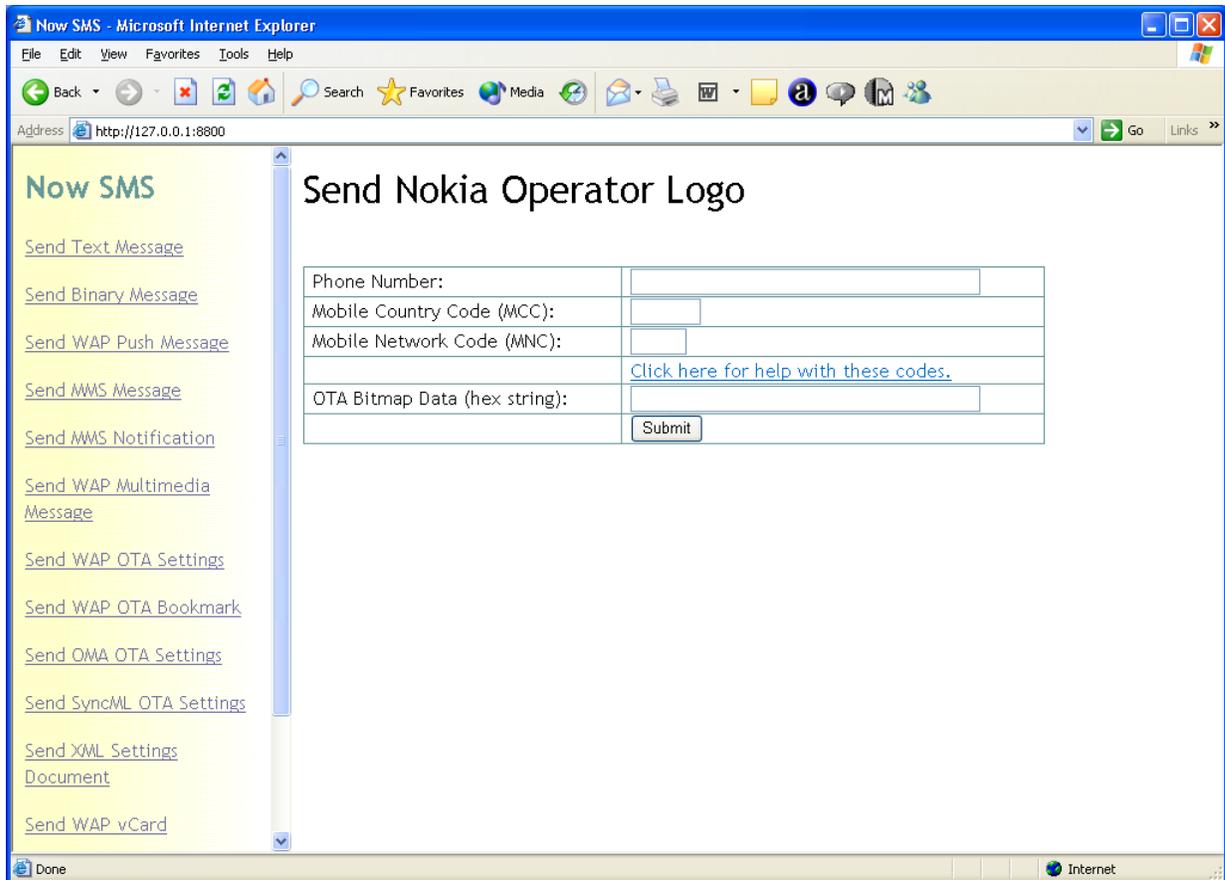
To send a Nokia Group Icon, you must have a hex string value for an OTA Bitmap, as defined by the Nokia Smart Messaging specification. The hex string format represents two characters for each binary byte of OTA Bitmap data. Documentation of the OTA Bitmap data format is beyond the scope of this document.

For those who wish to send Nokia Group icons programmatically via the Now SMS/MMS Gateway, note that this form includes the following hidden fields which are included as URL parameters when submitting the message to the server:

UDH = 06050415831583
PID = 0
DCS = F7

JavaScript in the HTML form adds the hex string "30" to the beginning of the OTA Bitmap string and submits it as the "Data" parameter in the URL.

Send Nokia Operator Logo



Nokia Operator logos are one of the more complicated of the Nokia Smart Messaging formats. To send a Nokia Operator logo, you must have a hex string value for an OTA Bitmap, as defined by the Nokia Smart Messaging specification. The hex string format represents two characters for each binary byte of OTA Bitmap data. Documentation of the OTA Bitmap data format is beyond the scope of this document. You must also know the Mobile Country Code (MCC) and Mobile Network Code (MNC) values of the network operator to which the recipient is subscribed. A link on the form provides more information on MCC and MNC codes, and a pointer to the URL <http://www.gsmworld.com/roaming/gsminfo/index.shtml>, from which you can look up the MCC and MNC codes of various network operators.

For those who wish to send Nokia Operator logos programmatically via the Now SMS/MMS Gateway, note that this form includes the following hidden fields which are included as URL parameters when submitting the message to the server:

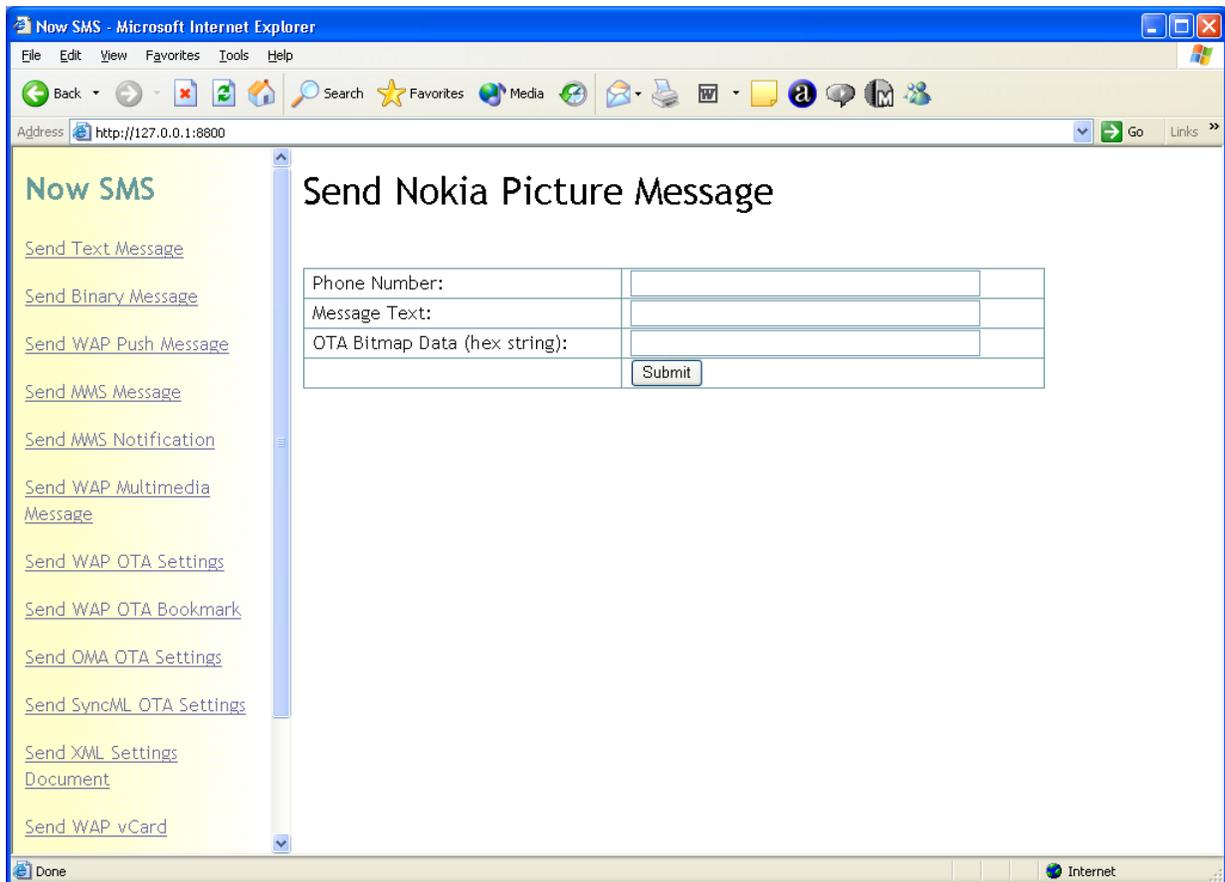
UDH = 06050415821582

PID = 0

DCS = F7

JavaScript in the HTML converts the MCC and MNC codes into the format required by the Nokia Smart Messaging specification, and combines them with the OTA Bitmap data to create a valid operator logo message in the URL "Data" parameter submitted by the form.

Send Nokia Picture Message



The screenshot shows a Microsoft Internet Explorer browser window titled "Now SMS - Microsoft Internet Explorer". The address bar displays "http://127.0.0.1:8800". The main content area is titled "Send Nokia Picture Message". On the left side, there is a vertical menu with the following links: [Send Text Message](#), [Send Binary Message](#), [Send WAP Push Message](#), [Send MMS Message](#), [Send MMS Notification](#), [Send WAP Multimedia Message](#), [Send WAP OTA Settings](#), [Send WAP OTA Bookmark](#), [Send OMA OTA Settings](#), [Send SyncML OTA Settings](#), [Send XML Settings Document](#), and [Send WAP vCard](#). The main form contains three input fields: "Phone Number:", "Message Text:", and "OTA Bitmap Data (hex string):". A "Submit" button is located at the bottom right of the form.

Nokia Picture Messaging should not be confused with MMS picture messaging. The Nokia picture messaging format typically only allows for the submission of small specially formatted black and white pictures, whereas MMS provides support for larger color images in a variety of different formats.

To send a Nokia Picture Message, you must have a hex string value for an OTA Bitmap, as defined by the Nokia Smart Messaging specification. The hex string format represents two characters for each binary byte of OTA Bitmap data. Documentation of the OTA Bitmap data format is beyond the scope of this document. A picture message also includes a short text message.

For those who wish to send Nokia Picture Messages programmatically via the Now SMS/MMS Gateway, note that this form includes the following hidden

fields which are included as URL parameters when submitting the message to the server:

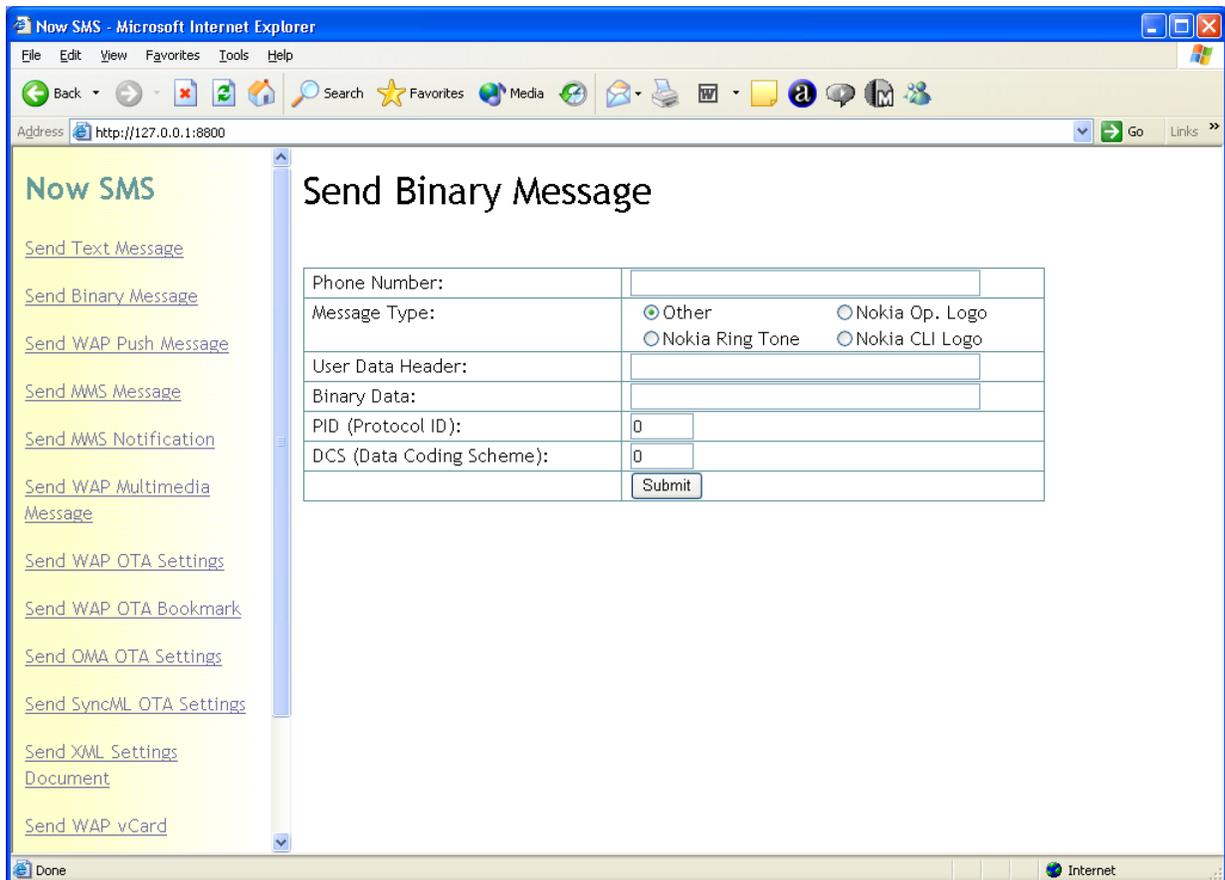
UDH = 060504158A158A

PID = 0

DCS = F7

JavaScript in the HTML form combines the message text and the OTA bitmap data to create a valid picture message in the URL "Data" parameter submitted by the form.

Send Binary Message Other



The screenshot shows a web browser window titled "Now SMS - Microsoft Internet Explorer" with the address bar showing "http://127.0.0.1:8800". The page content includes a sidebar with navigation links and a main form titled "Send Binary Message".

Now SMS

- [Send Text Message](#)
- [Send Binary Message](#)
- [Send WAP Push Message](#)
- [Send MMS Message](#)
- [Send MMS Notification](#)
- [Send WAP Multimedia Message](#)
- [Send WAP OTA Settings](#)
- [Send WAP OTA Bookmark](#)
- [Send OMA OTA Settings](#)
- [Send SyncML OTA Settings](#)
- [Send XML Settings Document](#)
- [Send WAP vCard](#)

Send Binary Message

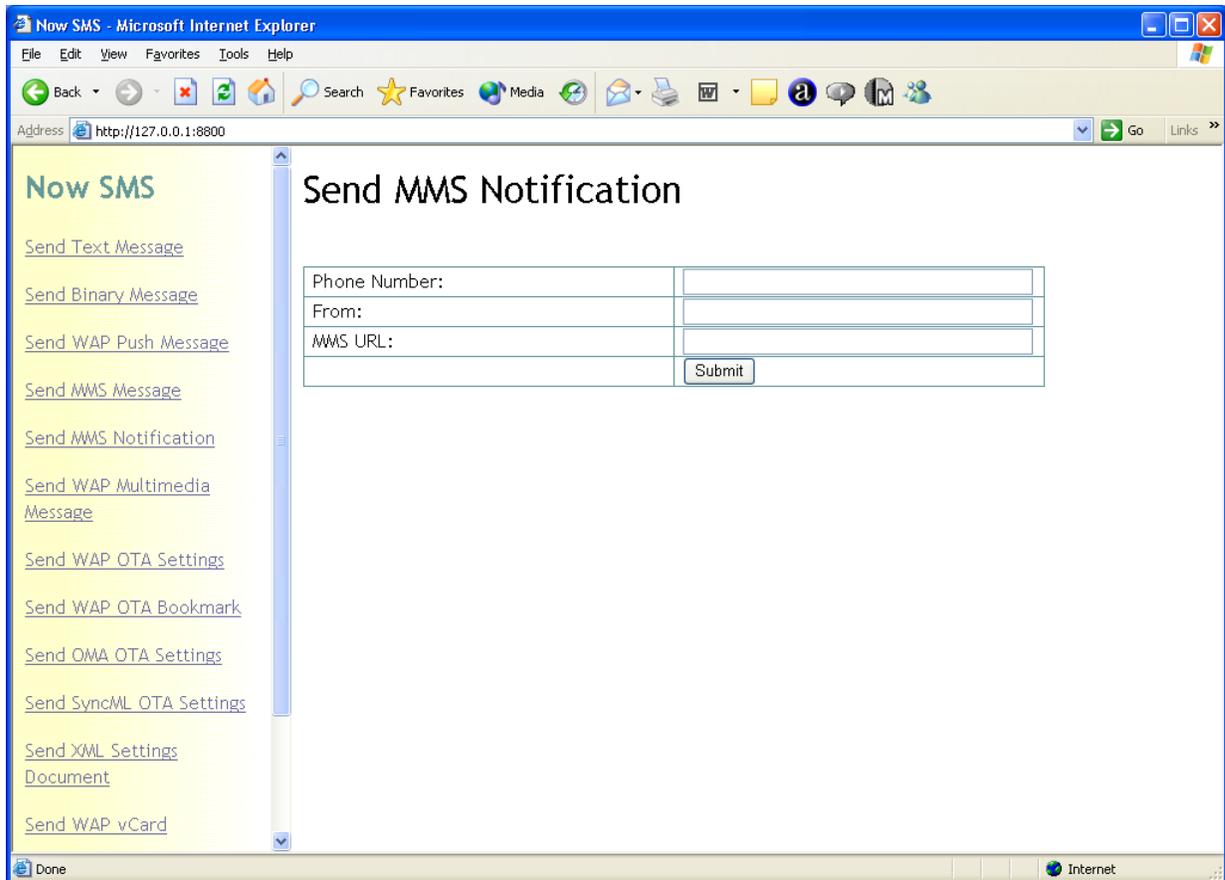
Phone Number:	<input type="text"/>
Message Type:	<input checked="" type="radio"/> Other <input type="radio"/> Nokia Op. Logo <input type="radio"/> Nokia Ring Tone <input type="radio"/> Nokia CLI Logo
User Data Header:	<input type="text"/>
Binary Data:	<input type="text"/>
PID (Protocol ID):	<input type="text" value="0"/>
DCS (Data Coding Scheme):	<input type="text" value="0"/>
	<input type="button" value="Submit"/>

The "Send Binary Message Other" form allows for the submission of other types of binary messages. This typically requires more knowledge of the binary SMS protocol that you are attempting to use, but this web form can be convenient for testing.

Send WAP Push Message

Connection Type:	<input checked="" type="radio"/> Service Indication (SI) <input type="radio"/> Service Load (SL)
Phone Number:	<input type="text"/>
WAP URL:	<input type="text"/>
Text:	<input type="text"/>
Signal Action:	<input type="radio"/> None <input type="radio"/> Low <input checked="" type="radio"/> Medium <input type="radio"/> High <input type="radio"/> Delete
SI ID (optional):	<input type="text"/>
SI Created (optional):	<input type="text"/>
SI Expires (optional):	<input type="text"/>
	<input type="button" value="Submit"/>

It has never been simpler to send a WAP Push message. Simply enter a phone number, a WAP URL (if the "http://" prefix is not included it will be added automatically), and some text to be included in the informational message displayed to the user. The gateway will automatically generate and send a WAP Push "Service Indication" (SI) message to the specified phone number. For information on other parameters, please refer to the WAP Push specifications for "ServiceIndication" and "ServiceLoad" push message content.



The alternative MMS interface, “Send MMS Notification” is intended for more advanced developers. After creating a binary MMS message file (don’t worry the gateway includes tools to help you do this!), and storing the message file on a web server with a MIME type of “application/vnd.wap.mms-message”, this dialog shows how the gateway can be used to send a notification to the message recipient that instructs the recipient’s phone to connect to the specified URL to retrieve the MMS message content.

Note: When the “Send MMS Notification” function is used, the MMS Notification is sent to the recipient independent of the MMSC built-in to the gateway. The message recipient will fetch the message directly from the URL specified. As the message is not routed through the MMSC, the MMSC cannot provide dynamic content adaptation and conversion services.

Send WAP OTA Settings

Connection Type:	<input checked="" type="radio"/> GPRS <input type="radio"/> GSM/CSD
Phone Number to Receive Settings:	<input type="text" value="+14435701011"/>
GPRS Access Point Name:	<input type="text" value="internet2.voicestream.com"/>
GPRS Login Parameters:	<input checked="" type="radio"/> Automatic <input type="radio"/> Prompted
GPRS User Name:	<input type="text"/>
GPRS Password:	<input type="text"/>
GPRS Login Type:	<input checked="" type="radio"/> Standard (PAP) <input type="radio"/> Secure (CHAP) <input type="radio"/> MS-CHAP
WAP Gateway IP Address:	<input type="text" value="213.48.20.10"/>
WAP Gateway Login Parameters:	<input checked="" type="radio"/> Automatic <input type="radio"/> Prompted
WAP Gateway User Name:	<input type="text"/>
WAP Gateway Password:	<input type="text"/>
WAP Gateway Connection Type:	<input checked="" type="radio"/> Connection-oriented <input type="radio"/> Connection-less <input type="radio"/> Connection-oriented, Secure (WTLS) <input type="radio"/> Connection-less, Secure (WTLS)
Settings Name:	<input type="text" value="Now MMS"/>
Home Page URL:	<input type="text" value="http://mms.now.co.uk/index.wml"/>
MMS Message Server URL:	<input type="text" value="http://mms.now.co.uk/username=password"/>
	<input type="button" value="Submit"/>

The gateway supports sending WAP OTA (Over-the-Air) configuration information to WAP compatible mobile phones. The "Send WAP OTA Settings" option allows a complete WAP configuration profile to be sent to a compatible mobile phone (the gateway supports the Nokia/Ericsson OTA Settings specification).

The web menu interface provides support for GPRS and GSM/CSD (GSM dial-up) configurations.

Now SMS - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Now SMS

[Send Text Message](#)
[Send Binary Message](#)
[Send WAP Push Message](#)
[Send MMS Message](#)
[Send MMS Notification](#)
[Send WAP OTA Settings](#)
[Send WAP OTA Bookmark](#)
[Send Voice Mail Notification](#)

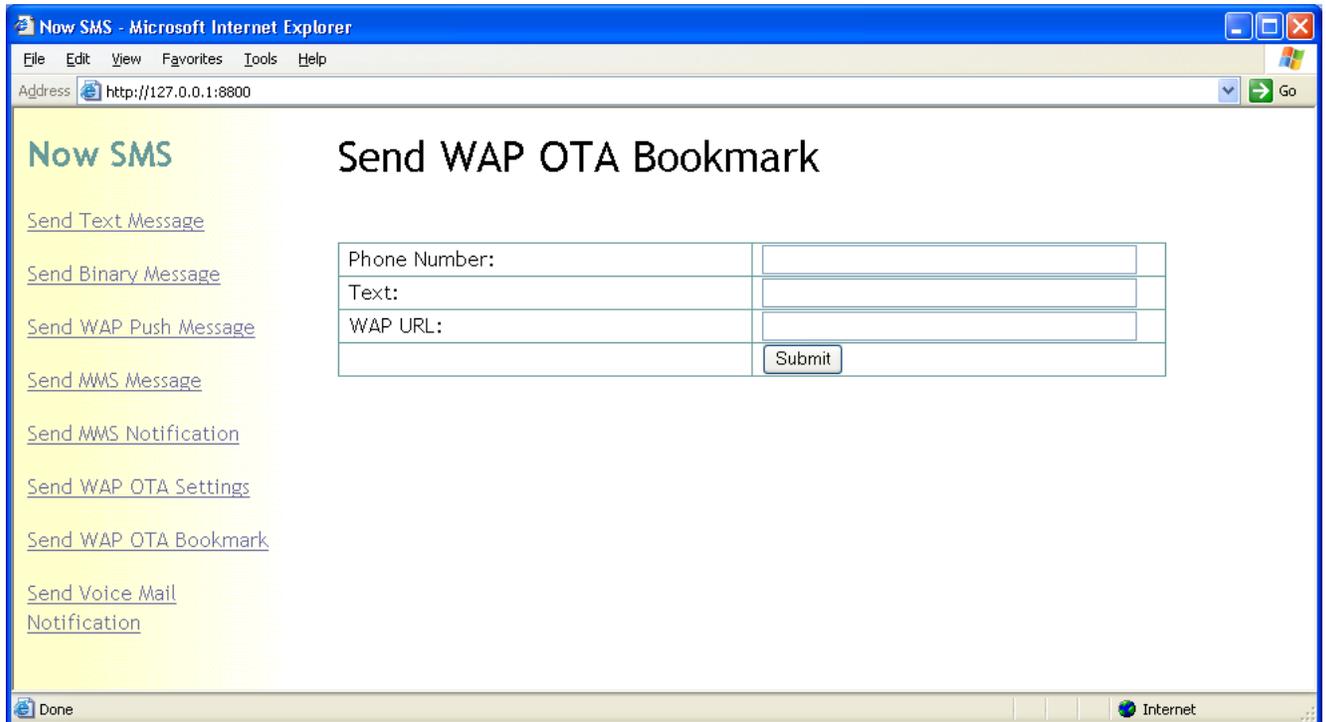
Send WAP OTA Settings

Connection Type:	<input type="radio"/> GPRS <input checked="" type="radio"/> GSM/CSD
Phone Number to Receive Settings:	+14435701011
ISP Phone Number:	+14435551212
ISP Call Type:	<input checked="" type="radio"/> Analogue <input type="radio"/> ISDN
ISP Call Speed:	AUTO
ISP Login Parameters:	<input checked="" type="radio"/> Automatic <input type="radio"/> Prompted
ISP User Name:	ispuser
ISP Password:	isppassword
ISP Login Type:	<input checked="" type="radio"/> Standard (PAP) <input type="radio"/> Secure (CHAP) <input type="radio"/> MS-CHAP
WAP Gateway IP Address:	213.48.20.10
WAP Gateway Login Parameters:	<input checked="" type="radio"/> Automatic <input type="radio"/> Prompted
WAP Gateway User Name:	
WAP Gateway Password:	
WAP Gateway Connection Type:	<input checked="" type="radio"/> Connection-oriented <input type="radio"/> Connection-less <input type="radio"/> Connection-oriented, Secure (WTLS) <input type="radio"/> Connection-less, Secure (WTLS)
Settings Name:	Dial-up
Home Page URL:	http://mms.now.co.uk/index.wml
MMS Message Server URL:	http://mms.now.co.uk/username=password

Done Internet

Settings are operator specific, so please refer to your operator for information on settings that are appropriate for your environment.

Send WAP OTA Bookmark



The gateway supports sending WAP OTA (Over-the-Air) configuration information to WAP compatible mobile phones. The "Send WAP OTA Bookmark" option allows bookmarks to be sent to compatible mobile phones. Simply specify the WAP URL, a title for the bookmark, and a phone number to which the bookmark should be sent. Be forewarned that many phones do not yet support this bookmark feature.

Send OMA OTA Settings

Now SMS

- [Send Text Message](#)
- [Send Binary Message](#)
- [Send WAP Push Message](#)
- [Send MMS Message](#)
- [Send MMS Notification](#)
- [Send WAP Multimedia Message](#)
- [Send WAP OTA Settings](#)
- [Send WAP OTA Bookmark](#)
- [Send OMA OTA Settings](#)
- [Send SyncML OTA Settings](#)
- [Send XML Settings Document](#)
- [Send WAP vCard](#)
- [Send Voice Mail Notification](#)
- [Distribution Lists](#)

Send OMA OTA Settings

Connection Type:	<input checked="" type="radio"/> GPRS <input type="radio"/> GSM/CSD
Phone Number to Receive Settings:	<input type="text"/>
GPRS Access Point Name:	<input type="text"/>
GPRS User Name:	<input type="text"/>
GPRS Password:	<input type="text"/>
GPRS Login Type:	<input checked="" type="radio"/> Standard (PAP) <input type="radio"/> Secure (CHAP) <input type="radio"/> MS-CHAP
WAP Gateway IP Address:	<input type="text"/>
WAP Gateway User Name:	<input type="text"/>
WAP Gateway Password:	<input type="text"/>
WAP Gateway Connection Type:	<input checked="" type="radio"/> Connection-oriented <input type="radio"/> Connection-less <input type="radio"/> Connection-oriented, Secure (WTLS) <input type="radio"/> Connection-less, Secure (WTLS) <input type="radio"/> Other <input type="text" value="9201"/>
Settings Name:	<input type="text"/>
Home Page URL:	<input type="text"/>
MMS Message Server URL:	<input type="text"/>
OTA PIN:	<input type="text"/>
OTA PIN Type:	<input type="radio"/> User PIN <input type="radio"/> Network PIN
<input type="button" value="Submit"/>	

The gateway supports sending OTA (Over-the-Air) configuration information to mobile phones that are compatible with the Open Mobile Alliance (OMA) Provisioning Content v1.1 Specification. The "Send OMA OTA Settings" option allows a complete WAP configuration profile to be sent to a compatible mobile phone. The "Send XML Settings" option provides greater flexibility, allowing any OMA Provisioning Content document to be binary encoding and sent by the gateway.

The gateway supports two types of PIN encoding for OMA Provisioning Content. A "User PIN" specifies a PIN that the recipient must enter to receive the settings. A "Network PIN" specifies the IMSI code for the SIM card in the recipient device.

Send SyncML OTA Settings

Now SMS

Send Text Message

Send Binary Message

Send WAP Push Message

Send MMS Message

Send MMS Notification

Send WAP Multimedia Message

Send WAP OTA Settings

Send WAP OTA Bookmark

Send OMA OTA Settings

Send SyncML OTA Settings

Send XML Settings Document

Send WAP vCard

Send SyncML OTA Settings

Phone Number to Receive Settings:	<input type="text"/>
Settings Name:	<input type="text"/>
Sync URL:	<input type="text"/>
Sync Port:	<input type="text"/>
User Authentication in SyncML can be performed on multiple levels. Unused authentication fields may be left blank.	
Sync Server User Name:	<input type="text"/>
Sync Server Password:	<input type="text"/>
Sync Server Auth Scheme:	<input checked="" type="radio"/> Basic <input type="radio"/> Digest MD5
Sync HTTP User Name:	<input type="text"/>
Sync HTTP Password:	<input type="text"/>
Sync HTTP Auth Scheme:	<input checked="" type="radio"/> Basic <input type="radio"/> Digest MD5
Contacts Database (URI):	<input type="text"/>
Contacts Content Type:	<input checked="" type="radio"/> text/x-vcard <input type="radio"/> text/vcard

SyncML OTA configuration information can be sent to SyncML compatible mobile phones. The “Send SyncML OTA Settings” option allows a complete SyncML configuration profile to be sent to a compatible mobile phone, using the Nokia/Ericsson OTA Settings specification.

To send SyncML OTA configuration information using the OMA Provisioning Content specification, use the “Send XML Settings Document” option.

Send XML Settings Document

The “Send XML Settings” Document allows any configuration document that is built using the OMA Provisioning Content v1.1 Specification, or the Nokia/Ericsson OTA Settings Specification, to be binary encoded and sent via the gateway.

Now SMS

[Send Text Message](#)

[Send Binary Message](#)

[Send WAP Push Message](#)

[Send MMS Message](#)

[Send MMS Notification](#)

[Send WAP Multimedia Message](#)

[Send WAP OTA Settings](#)

[Send WAP OTA Bookmark](#)

[Send OMA OTA Settings](#)

[Send SyncML OTA Settings](#)

[Send XML Settings Document](#)

[Send WAP vCard](#)

Send XML Settings Document

This form will accept the following types of XML settings documents:

- 1.) Nokia/Ericsson Over The Air Settings Specification (root XML element <CHARACTERISTIC-LIST>)
- 2.) SyncML Over The Air Settings (root XML element <SyncSettings>)
- 3.) OMA (Open Mobile Alliance) Provisioning Content (root XML element <wap-provisioningdoc>)

Phone Number:

XML Content of Settings Document:

OTA PIN:

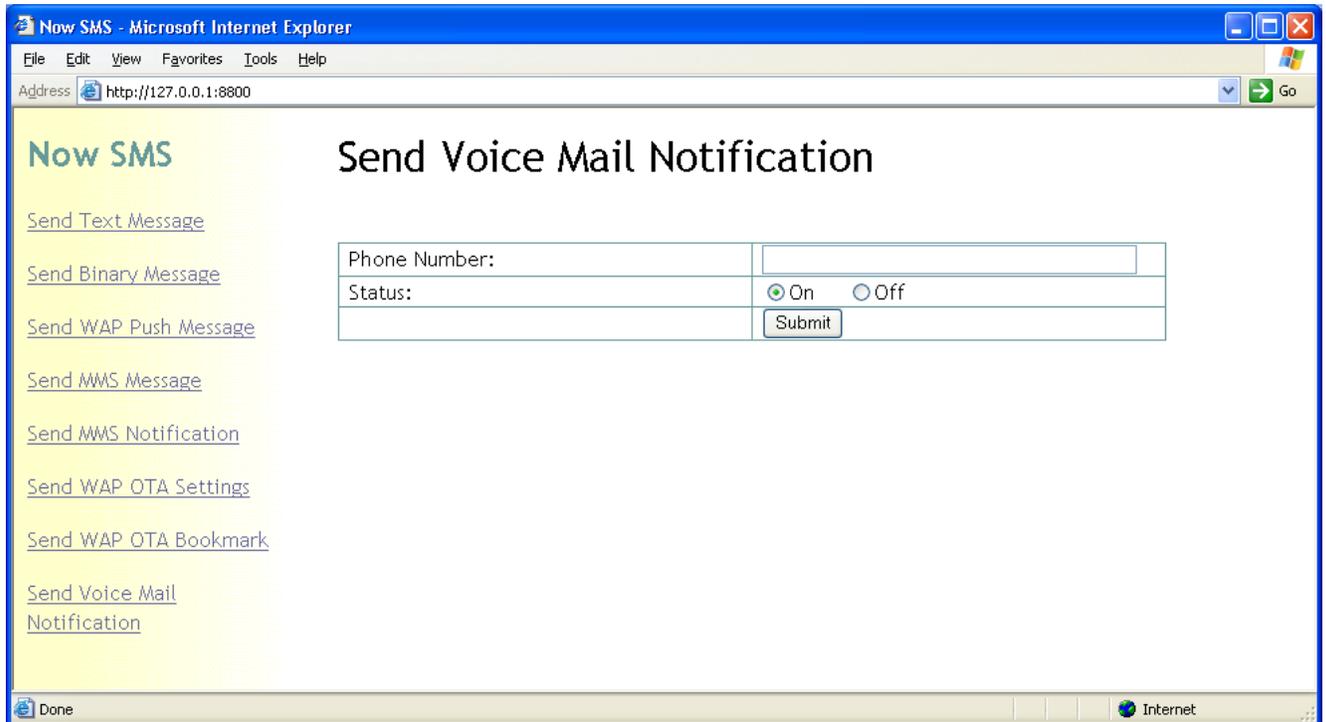
OTA PIN Type: User PIN Network PIN

The Nokia/Ericsson OTA Settings specification is available from either the Nokia or Ericsson web sites at <http://www.forum.nokia.com> or <http://www.ericsson.com/mobilityworld>.

The OMA Provisioning Content Specification is available for download from the Open Mobile Alliance web site at <http://www.openmobilealliance.org>.

The gateway supports two types of PIN encoding for OMA Provisioning Content. A “User PIN” specifies a PIN that the recipient must enter to receive the settings. A “Network PIN” specifies the IMSI code for the SIM card in the recipient device.

Send Voice Mail Notification



Voice Mail Notification Messages are special SMS messages that are used to tell the user that they have voice mail waiting. On most mobile phones, the phone displays a message prompt, and the user can press a single key to be transferred to voice mail. This voice mail phone number is configurable via the mobile phone settings.

This dialog supports sending special SMS messages to turn on and off the voice mail waiting status.

Sending Text Messages

To send a text SMS message via a menu driven interface, please see the help section titled "[Web Menu Interface](#)". This section describes how to send a text message programmatically via URL parameters.

To send a text message via SMS, use the following URL format:

<http://127.0.0.1:8800/?PhoneNumber=xxxxxxx&Text=abc+def+ghi>

For 127.0.0.1, please substitute the IP address or host name assigned to your gateway PC. (Note: 127.0.0.1 is a local loopback address that can be utilized when you are connecting to the gateway from the same computer.)

For 8800, please substitute the port number that the gateway is configured to use.

Substitute the phone number that you wish to send the SMS message to for the "xxxxxxx" in the "PhoneNumber" parameter. Use either the local phone number format, or the international phone number format (your network provider may or may not allow you to send to international phone numbers). If the international phone number format is used, note that you must substitute "%2B" for the "+" character, because of URL escaping restrictions. For example, to send an SMS to +447778001210, use the following URL format:

<http://127.0.0.1:8800/?PhoneNumber=%2B447778001210&Text=abc+def+ghi>

Substitute the text of the SMS message in the "Text" parameter. Note that due to URL escaping restrictions, space characters should be replaced with "+" characters. Also, certain special characters, such as "?", "&", ":" and "=" need to be replaced with an escape character. The gateway expects characters to be encoded in UTF-8 (Unicode-based) format, therefore some characters, including the Euro (€) may require multiple escaped characters. (Note: The Web Menu Interface automatically performs this escaping.) The following table shows common characters that must be escaped:

"	%22
<	%3C
>	%3E
&	%26
+	%2B
#	%23
%	%25

*	%2A
!	%21
,	%2C
'	%27
\	%5C
=	%3D
€	%E2%82%AC

Message text up to 160 characters in length can be sent in a single SMS message. The gateway automatically supports the sending of longer messages by utilizing "concatenated SMS" to send messages larger than 160 characters in length. Note that some older mobile phones will not support longer SMS messages. For longer SMS messages, one message is sent for every 153 characters of text in the message.

Additional supported URL parameters when sending text messages or any other type of message supported by the gateway via the web interface include:

"ContinueURL" - After the gateway submits the SMS message, the gateway will return a web page indicating that the message was submitted successfully. This page will contain a "Continue" link that points to the URL specified in this parameter. If "ContinueURL" is not specified, the link will default to "javascript:history.back()" which links to the previous page. Note that the "http://" portion of the URL is not necessary and is assumed. Also note that it may be necessary to escape some URL characters, please refer to the above table for more information.

"Sender" - Specifies the phone number to be included as the sender of the message. (Note: Depending on the configuration of the gateway and the SMSCs to which the gateway connects, this value may be ignored.)

For a complete list of URL parameters, please refer to the section ["URL Parameters for Sending Messages"](#).

Sending Binary Messages

To send a binary SMS message via a menu driven interface, please see the help section titled "[Web Menu Interface](#)". This section describes how to send a binary message programmatically via URL parameters.

To send a binary message via SMS, please refer to the specifications for the particular binary message format that you wish to send, and use the following URL format:

<http://127.0.0.1:8800/?PhoneNumber=xxxxxxx&Data=00112233445566&UDH=060504030201&pid=AA&dcs=AA&binary=1>

For 127.0.0.1, please substitute the IP address or host name assigned to your gateway PC. (Note: 127.0.0.1 is a local loopback address that can be utilized when you are connecting to the gateway from the same computer.)

For 8800, please substitute the port number that the gateway is configured to use.

Substitute the phone number that you wish to send the SMS message to for the "xxxxxxx" in the "PhoneNumber" parameter. Use either the local phone number format, or the international phone number format (your network provider may or may not allow you to send to international phone numbers). If the international phone number format is used, note that you must substitute "%2B" for the "+" character, because of URL escaping restrictions. For example, to send an SMS to +447778001210, use %2B447778001210 instead.

The "Data" parameter should include a string of hexadecimal digits that form the binary data content for the message.

The "UDH" parameter should include a string of hexadecimal digits that form the binary user data header for the message. Common UDH parameter settings include "06050415811581" for Nokia ring tones, "06050415821582" for Nokia operator logos, and "06050415831583" for Nokia CLI logos.

The "pid" parameter is a hexadecimal value between 0 and FF that specifies the GSM 03.40 TP-Protocol-Identifier.

The "dcs" parameter is a hexadecimal value between 0 and FF that specifies the GSM 03.38 SMS Data Coding Scheme. F7 is a common data coding scheme for most binary message formats.

The “binary” parameter should be set to “1” to tell the gateway that this is a binary message.

An example EMS message which includes a predefined EMS animation and a predefined EMS sound is shown below:

<http://127.0.0.1:8800/?phone=xxxxxxx&udh=080D0200040B020007&data=00&binary=1>

Refer to specifications such as “How to Create EMS Services” on the Ericsson developer site, and “Smart Messaging Services” on the Nokia developer site for more information on binary formats for SMS messages.

The gateway includes some HTML forms to simplify the process of creating Nokia smart messages. Those HTML message forms include JavaScript commands that build the binary message parameters for submitting smart messages. Refer to the JavaScript in the corresponding HTML forms and the Nokia “Smart Messaging Services” specification for additional information.

Please note that when a “user data header” is included, the data portion of the SMS message must be encoded in binary format. Text formats cannot be mixed with a user data header.

For a complete list of URL parameters, please refer to the section [“URL Parameters for Sending Messages”](#).

Sending WAP Push Messages

To send a WAP Push message via a menu driven interface, please see the help section titled "[Web Menu Interface](#)". This section describes how to send a WAP Push programmatically via URL parameters.

WAP Push messages are specially formatted SMS messages that display an alert message to the user, and give the user the option of connecting directly to a particular URL via the mobile phone's WAP browser.

For example, an e-mail application might send an alert that tells the user they have new e-mail, with a URL link to connect directly to a WAP e-mail application.

The WAP specifications define a format for applications to create XML-based "PAP" (Push Access Protocol) documents that can be posted to an operator's "PPG" (Push Proxy Gateway), in order to deliver a WAP push message to a mobile device.

Unfortunately, the complexity of this format, and the reluctance of operators to open their "PPG" to just anyone, has made it difficult for developers to deploy "WAP Push" in their applications.

The Now SMS/MMS Gateway makes it easy to generate and deliver "WAP Push" messages. While the gateway does not support all of the options available via the PAP-based PPG interface, it does implement "WAP Push" in an elegantly simple solution.

To send a WAP Push message, use the following URL format:

<http://127.0.0.1:8800/?PhoneNumber=xxxxxxx&WAPURL=name.domain/path&Text=abc+def+ghi>

For 127.0.0.1, please substitute the IP address or host name assigned to your gateway PC. (Note: 127.0.0.1 is a local loopback address that can be utilized when you are connecting to the gateway from the same computer.)

For 8800, please substitute the port number that the gateway is configured to use.

Substitute the phone number that you wish to send the SMS message to for the "xxxxxxx" in the "PhoneNumber" parameter. Use either the local phone number format, or the international phone number format (your network provider may or may not allow you to send to international phone numbers). If

the international phone number format is used, note that you must substitute "%2B" for the "+" character, because of URL escaping restrictions. For example, to send an SMS to +447778001210, use %2B447778001210 instead.

The alert text for the WAP Push message is contained in the "Text" parameter, and utilizes the same format as described in "Sending Text Messages".

Note that there are two types of "WAP Push" messages, "Service Indication (SI)" and "Service Load (SL)". The "SL" format can be selected by including "WAPSL=1" as a URL parameter, and does not support a "Text" parameter, while the "SI" format does. (By specification, the "SL" format was designed to tell the browser to connect to a URL without user intervention. However, for security reasons, mobile phones will always display a prompt before connecting to a URL. Therefore, the lack of a text parameter makes the "SL" format considerably less user-friendly than the "SI" format, and in practice, most users will exclusively use the "SI" format.)

The URL to be pushed to the mobile device is specified in the "WAPURL" parameter. Note that the "http://" portion of the URL is not necessary and is assumed. Also note that it may be necessary to escape some URL characters, please refer to the table in the "[Sending Text Messages](#)" section for common characters that must be escaped.

For a complete list of URL parameters, please refer to the section "[URL Parameters for Sending Messages](#)".

Sending WAP OTA Messages

To send a WAP OTA message via a menu driven interface, please see the help section titled "[Web Menu Interface](#)". This section describes how to send a WAP OTA message programmatically via URL parameters.

WAP OTA (Over The Air) Messages are special SMS messages that contain information used to configure the settings of a WAP browser in a mobile phone. There are two basic types of OTA messages, the most common type of OTA message contains a complete set of configuration parameters for the WAP browser, and a second type of OTA message contains a single bookmark.

WAP Bookmark OTA Messages

The WAP Bookmark OTA message is only supported by few mobile phones (at the time this document was written). To send a WAP Bookmark OTA message, use the following URL format:

<http://127.0.0.1:8800/?PhoneNumber=xxxxxxx&WAPBookmark=name.domain/path>

For 127.0.0.1, please substitute the IP address or host name assigned to your gateway PC. (Note: 127.0.0.1 is a local loopback address that can be utilized when you are connecting to the gateway from the same computer.)

For 8800, please substitute the port number that the gateway is configured to use.

Substitute the phone number that you wish to send the SMS message to for the "xxxxxxx" in the "PhoneNumber" parameter. Use either the local phone number format, or the international phone number format (your network provider may or may not allow you to send to international phone numbers). If the international phone number format is used, note that you must substitute "%2B" for the "+" character, because of URL escaping restrictions. For example, to send an SMS to +447778001210, use %2B447778001210 instead.

The URL to be pushed to the mobile device as a bookmark is specified in the "WAPBookmark" parameter. Note that the "http://" portion of the URL is not necessary and is assumed. Also note that it may be necessary to escape some URL characters, please refer to the table in the "[Sending Text Messages](#)" section for common characters that must be escaped.

WAP Configuration OTA Messages

The gateway supports OTA documents containing "Browser settings" or "Browser bookmarks", compatible with the Nokia/Ericsson ***Over The Air Settings Specification***, with support up to v7.0.

This specification can be downloaded from the developer area of either the [Nokia](#) or [Ericsson](#) web sites.

Three approaches are provided for sending WAP Configuration OTA messages:

- 1.) The "web menu" interface of the gateway provides a menu based interface for specifying WAP configuration settings.
- 2.) URL parameters can be passed to the gateway to dynamically define WAP configuration settings.
- 3.) Manually create an OTA document based on the Nokia/Ericsson specification, and store this document in the "OTA" subdirectory of the gateway installation, or POST the document to the gateway via the web interface.

The "web menu" interface is defined in the help section titled "[Web Menu Interface](#)". The other approaches are defined below.

WAP Configuration OTA - URL Parameters

Using the WAP Configuration OTA URL parameters requires a good understanding of the Nokia/Ericsson OTA Specification. This document specifies the mapping of gateway URL parameters to OTA setting parameters. The value provided for the gateway URL parameter is applied to the corresponding OTA setting parameter. Please refer to the Nokia/Ericsson specification for documentation of the OTA setting parameters.

OTA_BEARER - maps to TYPE=ADDRESS, PARM NAME=BEARER
OTA_PPP_AUTHTYPE - maps to TYPE=ADDRESS, PARM NAME=PPP_AUTHTYPE
OTA_PPP_AUTHNAME - maps to TYPE=ADDRESS, PARM NAME=PPP_AUTHNAME
OTA_PPP_AUTHSECRET - maps to TYPE=ADDRESS, PARM NAME=PPP_AUTHSECRET
OTA_PPP_LOGINTYPE - maps to TYPE=ADDRESS, PARM NAME=PPP_LOGINTYPE
OTA_PROXY - maps to TYPE=ADDRESS, PARM NAME=PROXY
OTA_PROXY_TYPE - maps to TYPE=ADDRESS, PARM NAME=PROXY_TYPE
OTA_PROXY_AUTHNAME - maps to TYPE=ADDRESS, PARM NAME=PROXY_AUTHNAME
OTA_PROXY_AUTHSECRET - maps to TYPE=ADDRESS, PARM NAME=PROXY_AUTHSECRET
OTA_PROXY_LOGINTYPE - maps to TYPE=ADDRESS, PARM NAME=PROXY_LOGINTYPE
OTA_PORT - maps to TYPE=ADDRESS, PARM NAME=PORT
OTA_CSD_DIALSTRING - maps to TYPE=ADDRESS, PARM NAME=CSD_DIALSTRING

OTA_CSD_CALLTYPE - maps to TYPE=ADDRESS, PARM NAME=CSD_CALLTYPE
OTA_CSD_CALLSPEED - maps to TYPE=ADDRESS, PARM NAME=CSD_CALLSPEED
OTA_ISP_NAME - maps to TYPE=ADDRESS, PARM NAME=ISP_NAME
OTA_SMS_SMSC_ADDRESS - maps to TYPE=ADDRESS, PARM NAME=SMS_SMSC_ADDRESS
OTA_USSD_SERVICE_TYPE - maps to TYPE=ADDRESS, PARM
NAME=USSD_SERVICE_TYPE
OTA_GPRS_ACCESSPOINTNAME - maps to TYPE=ADDRESS, PARM
NAME=GPRS_ACCESSPOINTNAME
OTA_URL - maps to TYPE=URL
OTA_MMSURL - maps to TYPE=MMSURL
OTA_NAME - maps to TYPE=NAME, PARM NAME=NAME
OTA_BOOKMARK_NAME - maps to TYPE=BOOKMARK, PARM NAME=NAME
OTA_BOOKMARK_URL - maps to TYPE=BOOKMARK, PARM NAME=URL
OTA_ID - maps to TYPE=ID, PARM NAME=NAME

Note that the "Send WAP OTA Settings" implementation in the gateway "web menu" interface uses this URL interface to submit OTA setting parameters. Viewing the source HTML for the corresponding "web menu" interface pages may provide an improved understanding of this URL interface.

WAP Configuration OTA - OTA Documents

It is also possible to provide OTA configurations by creating one or more OTA documents that contain settings compatible with the Nokia/Ericsson specification.

OTA documents should be created in the OTA subdirectory of the gateway installation, and given a file extension of ".OTA".

Once an OTA document has been created, to send an OTA "Browser settings" file to a mobile phone, use the following URL format:

<http://127.0.0.1:8800/?PhoneNumber=xxxxxxx&OTA=filename>

The "OTA" parameter specifies the name of a file located in the OTA subdirectory of the gateway with a file extension of ".OTA". For example, in the above sample URL, the gateway would attempt to locate a file named "filename.OTA" in the OTA gateway subdirectory.

To send an OTA "Browser bookmarks" file to a mobile phone, use the following URL format:

<http://127.0.0.1:8800/?PhoneNumber=xxxxxxx&OTABookmark=filename>

The "OTABookmark" parameter uses the same format as the "OTA" parameter when sending "Browser settings", except that it expects the browser bookmark settings file to have a file extension of ".BM".

An example OTA "Browser settings" file is shown below, for additional information, please refer to the Nokia/Ericsson "Over The Air Settings Specification".

GSM/CSD Settings Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<CHARACTERISTIC-LIST>
  <CHARACTERISTIC TYPE="ADDRESS">
    <PARAM NAME="BEARER" VALUE="GSM/CSD"/>
    <PARAM NAME="PROXY" VALUE="12.34.56.78"/>
    <PARAM NAME="CSD_DIALSTRING" VALUE="+12135551212"/>
    <PARAM NAME="PPP_AUTHTYPE" VALUE="PAP"/>
  </CHARACTERISTIC>
  <CHARACTERISTIC TYPE="URL"
VALUE="http://mobileinternet.ericsson.com"/>
  <CHARACTERISTIC TYPE="NAME">
    <PARAM NAME="NAME" VALUE="Mobile Internet"/>
  </CHARACTERISTIC>
  <CHARACTERISTIC TYPE="BOOKMARK">
    <PARAM NAME="NAME" VALUE="Mobile Internet"/>
    <PARAM NAME="URL" VALUE="http://mobileinternet.ericsson.com"/>
  </CHARACTERISTIC>
</CHARACTERISTIC-LIST>
```

It is also possible to send OTA messages without creating an OTA document on the gateway by submitting an HTTP POST request to the gateway with the content of the POST message being an OTA "Browser Settings" file. It is not possible to submit such a request via a standard web browser, instead this request must be submitted programmatically. Submit the POST to a URL of:

<http://127.0.0.1:8800/?PhoneNumber=xxxxxxx&OTA=POST>

When submitting an OTA request in this manner, the HTTP POST request **must** include a "Content-Length:" header.

For a complete list of URL parameters, please refer to the section ["URL Parameters for Sending Messages"](#).

Sending OMA Provisioning Content OTA Messages

To send an OMA Provisioning Content OTA message via a menu driven interface, please see the help section titled "[Web Menu Interface](#)". This section describes how to send a OMA Provisioning Content message programmatically via URL parameters.

OMA Provisioning Content Messages are special SMS messages that contain information used to configure certain settings of a mobile phone, such as settings for the browser, MMS client or SyncML client.

The gateway supports OMA Provisioning Content documents compatible with the Open Mobile Alliance "***Provisioning Content Specification v1.1***".

This specification can be downloaded from the Open Mobile Alliance web site at <http://www.openmobilealliance.org>.

Two approaches are provided for sending OMA Provisioning Content messages:

- 1.) The "web menu" interface of the gateway provides a menu based interface for specifying simple browser and MMS client configuration settings.
- 2.) Manually create an OTA document based on the OMA Provisioning Content specification, and store this document in the "OTA" subdirectory of the gateway installation, or POST the document to the gateway via the web interface.

The "web menu" interface is defined in the help section titled "[Web Menu Interface](#)". The other approach is defined below.

OMA Provisioning Content documents should be created in the OTA subdirectory of the gateway installation, and given a file extension of ".OTA".

Once a document has been created, to send the document to a mobile phone, use the following URL format:

<http://127.0.0.1:8800/?PhoneNumber=xxxxxxx&OMAOta=filename>

The "OMAOta" parameter specifies the name of a file located in the OTA subdirectory of the gateway with a file extension of ".OTA". For example, in

the above sample URL, the gateway would attempt to locate a file named "filename.OTA" in the OTA gateway subdirectory.

An example OMA Provisioning Content document for configuring browser settings on a mobile phone is shown below, for additional information, please refer to the OMA Provisioning Content Specification.

```
<wap-provisioningdoc>
  <characteristic type="BOOTSTRAP">
    <parm name="NAME" value="Movistar Spain"/>
  </characteristic>
  <characteristic type="NAPDEF">
    <parm name="NAME" value="Movistar Spain"/>
    <parm name="NAPID" value="Movistar_Spain_NAPID"/>
    <parm name="BEARER" value="GSM-GPRS"/>
    <parm name="NAP-ADDRESS" value="wap.movistar.es"/>
    <parm name="NAP-ADDRTYPE" value="APN"/>
    <characteristic type="NAPAUTHINFO">
      <parm name="AUTHTYPE" value="PAP"/>
      <parm name="AUTHNAME" value="WAPTM"/>
      <parm name="AUTHSECRET" value="WAPTM"/>
    </characteristic>
  </characteristic>
  <characteristic type="PXLOGICAL">
    <parm name="NAME" value="Movistar Spain"/>
    <parm name="PROXY-ID" value="Movistar Spain_Proxy"/>
    <parm name="STARTPAGE" value="http://wap.movistar.com"/>
    <characteristic type="PXPHYSICAL">
      <parm name="PHYSICAL-PROXY-ID" value="Movistar
Spain_PhProxy"/>
      <parm name="PXADDR" value="192.168.80.21"/>
      <parm name="PXADDRTYPE" value="IPV4"/>
      <parm name="TO-NAPID" value="Movistar_Spain_NAPID"/>
      <characteristic type="PORT">
        <parm name="PORTNBR" value="9201"/>
        <parm name="SERVICE" value="CO-WSP"/>
      </characteristic>
    </characteristic>
  </characteristic>
  <characteristic type="APPLICATION">
    <parm name="APPID" value="w2"/>
    <parm name="TO-PROXY" value="Movistar Spain_Proxy"/>
    <parm name="NAME" value="Browser"/>
    <characteristic type="RESOURCE">
      <parm name="URI" value="http://wap.movistar.com"/>
      <parm name="STARTPAGE"/>
    </characteristic>
  </characteristic>
</wap-provisioningdoc>
```

It is also possible to send Provisioning Content messages without creating a document on the gateway by submitting an HTTP POST request to the gateway with the content of the POST message being the Provisioning Content document. This document can either be sent programmatically, or it can be sent via the “Send XML Settings” option in the web menu interface. To submit the document via HTTP POST, it should be submitted to a URL of:

<http://127.0.0.1:8800/?PhoneNumber=xxxxxxx&OTAOMA=POST>

When submitting an OTA request in this manner, the HTTP POST request **must** include a “Content-Length:” header.

The URL request can also include an OTAPIN parameter specifying the PIN associated with the request, and an OTAPINTYPE parameter specifying the type of PIN associated with the request (USERPIN or NETWPIN).

For a complete list of URL parameters, please refer to the section [“URL Parameters for Sending Messages”](#).

Sending MMS Notifications and Content

To send an MMS message via a menu driven interface, please see the help section titled "[Web Menu Interface](#)". This section describes how to send an MMS message programmatically via URL parameters.

It is also possible to send an MMS message by sending MMS content to an e-mail address on the gateway MMSC, which will be routed to a mobile phone recipient. Please refer to the section titled "[MMSC Messaging Server](#)" for information on configuring the MMSC to send and receive e-mail messages.

MMS (Multimedia Messaging Service) messages are sent using a combination of SMS and WAP technologies. When an MMS message is sent, a mobile device receives an MMS notification message via SMS. When this MMS notification message is received by the mobile device, the mobile device automatically initiates a WAP gateway connection to download the content of the MMS message.

To send an MMS message, you must first create an MMS message file. The format of an MMS message file is documented in the MMS Encapsulation Protocol specification published by the Open Mobile Association (<http://www.openmobileassociation.org>) and/or the WAP Forum (<http://www.wapforum.org>). The MMS message file format consists of an MMS message binary header, followed by a multipart MIME message where the multipart message is encoded in a binary multipart format as defined by the WAP Wireless Session Protocol (WSP) specification. This binary MMS message file is stored on a web server using a MIME type of application/vnd.wap.mms-message and an MMS message type of m-retrieve-conf. A subset of the binary MMS header is sent as an MMS notification message (MMS message type m-notification-ind) via SMS to the mobile device together with a URL pointer to the location of the complete message.

The gateway includes an MMS message compiler to assist in the creation of the MMS message files, which will be described shortly. It is also possible to create MMS message files by uploading the individual MMS message components via the gateway [web menu interface](#), or by sending the individual MMS message components as e-mail attachments to the built-in [MMS e-mail gateway](#). This section of the document focuses more on the programmatic creation of MMS message files.

Once an MMS message file has been built and published via a web server, the MMS notification message can be sent by the gateway using the following URL format:

<http://127.0.0.1:8800/?PhoneNumber=xxxxxxx&MMSURL=name.domain/path/filename.mms>

For 127.0.0.1, please substitute the IP address or host name assigned to your gateway PC. (Note: 127.0.0.1 is a local loopback address that can be utilized when you are connecting to the gateway from the same computer.)

For 8800, please substitute the port number that the gateway is configured to use.

Substitute the phone number that you wish to send the SMS message to for the "xxxxxxx" in the "PhoneNumber" parameter. Use either the local phone number format, or the international phone number format (your network provider may or may not allow you to send to international phone numbers). If the international phone number format is used, note that you must substitute "%2B" for the "+" character, because of URL escaping restrictions. For example, to send an SMS to +447778001210, use %2B447778001210 instead.

The URL that contains the MMS message file is specified in the "MMSURL" parameter. Note that the "http://" portion of the URL is not necessary and is assumed. Also note that it may be necessary to escape some URL characters, please refer to the table in the "[Sending Text Messages](#)" section for common characters that must be escaped. Before sending the MMS notification message, the gateway will validate that the MMS message file is of the MIME type application/vnd.wap.mms-message, and is of the MMS message type m-retrieve-conf.

Additional parameters supported for the MMS notification message include "MMSFROM" and "MMSSUBJECT", which can be used to override the message sender and subject in the MMS message file.

For a complete list of URL parameters, please refer to the section "[URL Parameters for Sending Messages](#)".

Creating MMS Message Files - MMSCOMP

As the MMS message file format is a binary file format, special tools are required to create MMS message files. The MMSCOMP utility is provided to assist in the creation of MMS message files. The MMSCOMP utility accepts text input files to create a binary MMS Message file.

A standard format is not defined for a text version of an MMS message file, however a format can easily be derived based upon the MMS Encapsulation Protocol Specification. The MMSCOMP utility accepts as input a file that

contains text representations of the MMS header, and one or more files (image, sound, text, etc.) to comprise the multipart message content.

MMSCOMP is a command-line utility that accepts the following command-line format:

MMSCOMP [-charset] header.file [data1.file [data2.file [data3.file ...]]]

-charset is used to specify a character set for the text components of the input files. This parameter is not required. The default character set is iso-8859-1. Other supported character sets include big5, iso-10646-ucs-2, iso-8859-1, iso-8859-2, iso-8859-3, iso-8859-4, iso-8859-5, iso-8859-6, iso-8859-7, iso-8859-8, iso-8859-9, shift_JIS, us-ascii, and utf-8.

header.file is a text file that contains text representations of the MMS message header. Supported MMS message headers include:

```
X-Mms-Message-Type: m-retrieve-conf (required)
X-Mms-Transaction-Id: text-string
X-Mms-Version: 1.0
Message-Id: text-string (usually x@x format)
Date: HTTP-date-format
From: address@domain or +InternationalPhoneNumber/TYPE=PLMN (Address-present-token is assumed)
To: address@domain or +InternationalPhoneNumber/TYPE=PLMN (use multiple headers for multiple recipients)
Cc: (same format as To)
Bcc: (same format as To)
Subject: text-string
X-Mms-Message-Class: Personal, Advertisement, Informational or Auto (default is Personal)
X-Mms-Priority: Low, Normal or High (default is Normal)
X-Mms-Delivery-Report: Yes or No (default is No)
X-Mms-Read-Reply: Yes or No (default is No)
Content-type: MIME-Type (default is application/vnd.wap.multipart.related, override default with caution!)
X-NowMMS-Content-Location: filename;content-type (optional, use multiple headers for multiple files)
```

Only the X-Mms-Message-Type header is required, other headers are optional. It is recommended that From and Subject headers always be included.

Note that while the message may contain multiple recipients in the To, Cc and Bcc headers, the gateway itself will only send the MMS notification message to one recipient at a time, as specified in the PhoneNumber parameter passed in a URL request.

At least one data file must be specified to provide the content of the MMS message. This data file can be specified on the command line (e.g., data1.file, data2.file, data3.file, ...), or it may be specified in the MMS header file with one or more X-NowMMS-Content-Location headers.

If the first data file is a SMIL (Synchronized Multimedia Integration Language) file, then MMSCOMP will automatically parse all "src" references in the SMIL

file and include any referenced files in the MMS multipart message file automatically.

If a SMIL file is to be included for presentation of the MMS message, it is recommended that the SMIL file always be specified as the first data file to the MMSCOMP command.

MMSCOMP determines the MIME type of each file based on the file extension, or when using the "X-NowMMS-Content-Type" header, the content type can be specified following the file name. File extensions of .jpg, .jpeg (image/jpeg), .gif (image/gif), .txt (text/plain), .wbmp (image/vnd.wap.wbmp) and .smil (application/smil) are recognized automatically. Other file extensions are read from the MMSCTYPE.INI file, or the Windows registry, under the registry key HKEY_CLASSES_ROOT\extension, where ".extension" is the extension of the file. For best results, please ensure that any file types that you are using with MMSCOMP are defined in the MMSCTYPE.INI file.

The output of the MMSCOMP command will be stored in a file that matches the name of the input header file, but with ".MMS" as the file extension.

Example:

Assume that:

- 1.) You have created an MMS message header file named "test.hdr".
- 2.) You have created a SMIL file named "testfile.smil". The "testfile.smil" file references three external files through the following references in the SMIL file:

```

<audio src="sound.amr"/>
<text src="text.txt" region="region1_1"/>
```
- 3.) The "image.jpg", "sound.amr" and "text.txt" files referenced by the "testfile.smil" file are located in the same directory as the "testfile.smil" file.

To create a binary MMS file, run:

```
MMSCOMP test.hdr testfile.smil
```

or

```
MMSCOMP test.hdr testfile.smil image.jpg sound.amr text.txt
```

If you want to specify a character set for the text file, include the -cCHARSET parameter:

```
MMSCOMP -cUTF-8 test.hdr testfile.smil
```

The output of the MMSCOMP file will be "test.mms" (e.g., the same filename as "test.hdr", but with a ".mms" file extension).

To send the compiled MMS file, you can either:

- 1.) Submit it as a content file to the "[Send MMS Message](#)" option in the "[Web Menu Interface](#)".
- 2.) Store the file on a web server using a MIME content type of "application/vnd.wap.mms-message", and use the gateway to send an "[MMS Notification Message](#)".
- 3.) If the message recipient is defined to the MMSC built into the gateway, attach the file to an e-mail message and send the message to username@mmsdomainname, where "username" is the alias name defined for the user on the "MMSC Users" dialog, and "mmsdomainname" is the "Domain Name for MMS E-Mail" defined on the "MMSC" dialog. Please refer to the section titled "[MMSC Messaging Server](#)" for information on configuring the MMSC to send and receive e-mail messages.

The MMSCOMP utility works well in conjunction with tools such as the SonyEricsson MMS Composer. The SonyEricsson MMS Composer creates SMIL (Synchronized Multimedia Integration Language) files, which are often used in MMS messages, but it does not create the complete binary MMS message file. To use the output of the SonyEricsson MMS Composer, use File/Export to export your MMS message to a specified directory. The composer will output a SMIL file and any included MMS message components to the directory specified. Create an MMS message header file, and then run MMSCOMP passing the name of the MMS message header file and the name of the SMIL file output by the SonyEricsson MMS Composer.

MMSC Messaging Server

The gateway also includes a powerful MMSC Messaging Server. We used to refer to this messaging server as being "simple", and even went so far as calling it "MMSLite". However, as the product has evolved, the built-in MMSC messaging server has become a world-class MMSC with content adaptation and conversion services not found on many higher end products.

Normally an MMSC is only utilized when a mobile phone user sends/originates an MMS message. The mobile phone sends the message to the MMSC for delivery. However, phones may also assume that the MMSC is responsible for the delivery of all MMS messages, and this may cause phones to disregard or get confused over MMS messages that are sent via the gateway instead of the operator's MMSC.

The MMSC Messaging Server built into the Now SMS/MMS Gateway implements the MMS protocols, and allows you to bypass the operator MMSC.

The MMSC supports sending MMS messages between mobile phones, and also supports sending and receiving MMS messages between mobile phones and standard internet e-mail systems.

While extremely powerful, the MMSC integrated into the gateway is a lightweight version of the Now MMSC product which is designed for operator environments (and includes user provisioning systems to integrate into the operator network).

Configuration for the MMSC can be found under the “MMSC” and “MMSC Users” configuration tabs of the gateway.

The “MMSC” configuration tab specifies general configuration information for the MMSC:

Now SMS/MMS Gateway v5.0

MMSC Users	MMSC VASP	MMSC Routing	Serial #
Service	SMSC	Web	MMSC
		SMS Users	2-Way

Activate MMSC Service

HTTP Port Number:

SMTP Port Number: Require AUTH

IP Address:

Local Host Name or IP Address:

Domain Name for MMS E-Mail:

SMTP Relay Host:

Enable MMS Delivery Receipts

Enable Dynamic Image + Audio Conversion

Scale Images to: Screen Size Max Supported Size

Enable E-Mail WAV to AMR Conversion

Enable E-Mail BMP to JPEG Conversion

Enable E-Mail to SMS Support

Domain Name for SMS E-Mail:

Max SMS messages per e-mail:

Admin User:

Admin Password:

OK Cancel Apply Help

The MMSC runs as a separate service process from the gateway. To activate the MMSC service, check the box next to the prompt **“Activate MMSC Service”**.

When a mobile phone sends or receives an MMS message, it makes an HTTP connection to an MMSC (usually through a WAP gateway). The MMSC contains an integrated HTTP server to process these connections. Please specify an available **“HTTP Port Number”** on the local computer for the HTTP server to accept connections from mobile phones.

MMS messages can be sent to and received from, standard internet e-mail accounts. To support this functionality, the MMSC provides message format conversions between MMS and SMTP. To accept messages from internet e-mail accounts, the MMSC contains an integrated SMTP server. Please specify an available **“SMTP Port Number”** on the local computer for the SMTP server to accept e-mail messages from internet e-mail recipients. Note that the

standard SMTP port number is 25, and you will require special configuration of another SMTP mail server in your network to support relaying to a port other than 25.

The PC that is running the gateway might have other web and mail services installed. For this reason, the gateway allows you to specify which of the available IP addresses on the current PC should be used by the gateway. The **“IP Address”** prompt displays the available IP addresses on the current PC. To make the gateway service available via any address on the current PC, select **“(all available)”**, otherwise select a specific IP address.

“Local Host Name or IP Address” specifies the local host name or IP address of the computer that is running the MMSC service. The name or address specified here will be used to construct URLs when sending MMS messages to mobile phones. If a host name is used, this host name must be defined in DNS and resolve back to the computer running the MMSC service.

“Domain Name for MMS E-Mail” specifies the SMTP domain name that is associated with users defined to the MMSC. When MMS users exchange e-mail messages with internet recipients, this is the SMTP domain name associated with the MMSC users. Note that the MMSC acts as an e-mail server, and you will need to configure DNS for this domain name so that internet mail sent to this domain name is properly routed to the PC running the gateway.

When an MMS user sends a message to an SMTP recipient, the MMSC requires an **“SMTP Relay Host”** to transfer the message to the correct internet mail server for the intended recipient. Please define an appropriate SMTP mail server in your network that will perform this SMTP message relay capability.

Checking **“Enable Dynamic Image and Audio Conversion”** enables the dynamic content adaptation and conversion services of the MMSC. The MMSC uses WAP/MMS **“User Agent Profile”** capabilities to determine the MIME formats that a device supports, as well as the maximum size of images supported by the device. Where required, the MMSC converts between common image formats (including, but not limited to GIF, JPG, PNG, BMP and WBMP) to deliver an image supported by the device. For images larger than the maximum size supported by the device, the MMSC will automatically scale the image to fit the device, speeding up download times. For audio formats, conversion between WAV and AMR is provided in the e-mail gateway interface. MIME types not supported by the receiving device, which cannot be supported, will be removed prior to delivery to the receiving device to prevent compatibility issues and unnecessary download delays.

Checking **“Enable E-Mail WAV to AMR Conversion”** enables conversion between WAV (audio/wav) and AMR (audio/AMR) formats when messages are exchanged between e-mail and MMS recipients. For audio recordings, the WAV

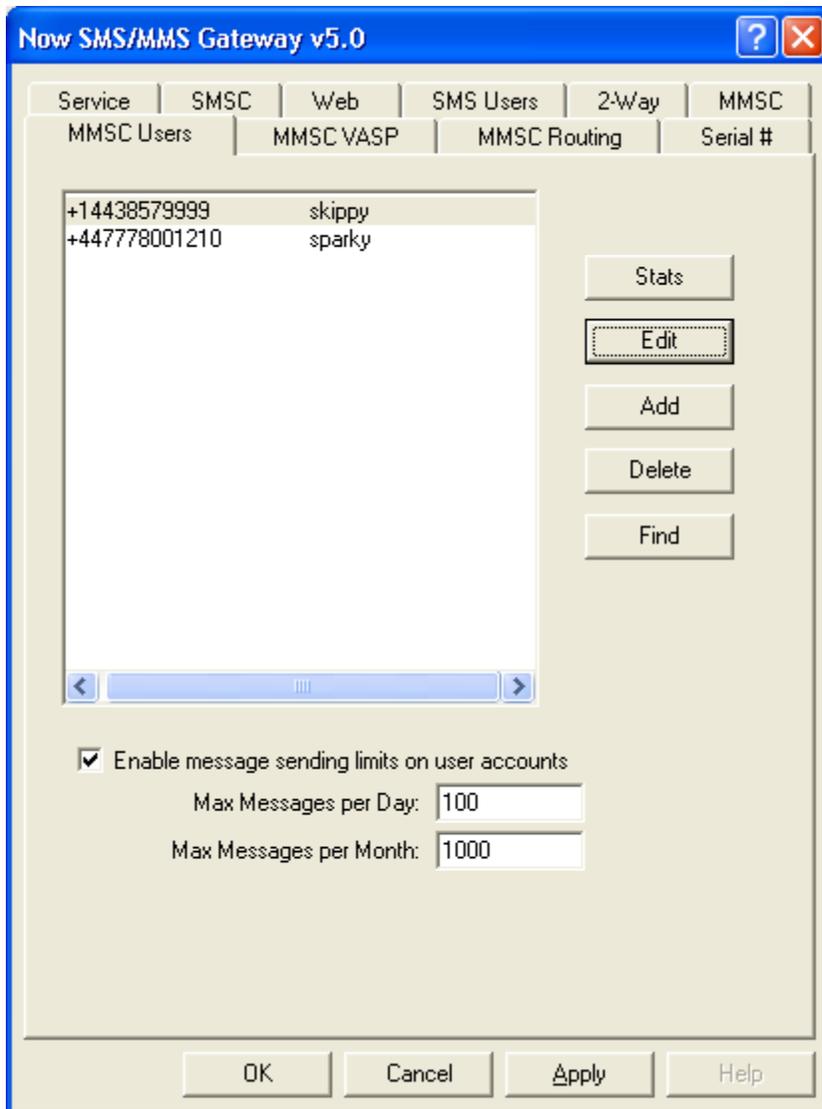
format is more commonly supported for e-mail recipients, while the AMR format is more commonly supported for MMS recipients. When this setting is enabled, AMR sound files going from MMS to e-mail are converted to WAV format, and WAV sound files going from e-mail to MMS are converted to AMR format.

Checking **“Enable E-Mail BMP to JPEG Conversion”** enables conversion of files from BMP (image/bmp) to JPEG (image/jpeg) format when messages are sent from an e-mail sender to an MMS recipient. The BMP format is somewhat common in PC environment, but is not frequently supported in MMS environments. Even when supported in MMS environments, BMP files are very large, which makes them awkward and inconvenient to transmit to MMS devices.

The **“Enable E-Mail to SMS Support”** checkbox enables the SMTP interface to be used to send SMS messages. This setting is used primarily in conjunction with bulk e-mail delivery via SMTP Authentication, as described in the [“SMS User Accounts”](#) section of this document. Gateway user accounts can be allowed to login via SMTP with an e-mail account to perform bulk delivery of SMS messages. If this configuration setting looks out of place, it is because this SMS gateway capability is provided through the MMSC’s SMTP server. When enabled, specify a **“Domain Name for SMS E-Mail”**, so that the gateway can identify by the domain name of the message recipient whether to route the message via SMS or MMS. Additionally, specify the **“Maximum number of SMS messages per e-mail”** to be used when converting from SMTP to SMS. When set to a value greater than 1, the gateway will use long (concatenated) SMS messages to support messages longer than 160 characters.

An administrative interface for adding, deleting and modifying users is available via the HTTP port of the MMSC if an **“Admin User”** is defined on the MMSC dialog. If an administrative user is defined, then the administrative interface is available by connecting via a web browser to **“/ADMIN”** on the MMSC HTTP port (e.g., <http://127.0.0.1/ADMIN>), and supplying the defined administrative user name and password. The administrative interface provides functionality similar to that described for the **“MMSC Users”** tab dialog, defined in the following section.

The **“MMSC Users”** tab defines users that are allowed to utilize the MMSC.



To define a user to use the MMSC, you must define a phone number using international format, and an alias name for the user account. (The alias name will be used as the user name when sending and receiving SMTP e-mail.)

Note that for a mobile phone user to use the simple MMSC integrated with this gateway, the mobile phone user must configure their MMSC (MMS Messaging Server) to point to the address of the MMSC, and include their user name and password in the MMSC URL. Example:

<http://x.x.x.x:81/username=password>

or

<http://host.domain:81/username=password>

The username can be either the user's alias name or phone number.

E-Mail - MMS Gateway

The MMSC built into the gateway supports the bi-directional exchange of MMS messages between mobile phone users and internet e-mail accounts.

To enable this capability, the MMSC must be able to send and receive SMTP internet e-mail. The configuration screens for the [built-in MMSC](#), define the required settings for sending and receiving SMTP internet e-mail.

When a mobile phone user sends a message to an e-mail recipient, the gateway will convert the message to SMTP e-mail format. Individual components of the MMS message will be sent as file attachments to the e-mail message.

To send an MMS message to a mobile phone from an e-mail client, address the message to `username@mmsdomainname`, where "username" is the alias name defined for the user on the "MMSC Users" dialog, and "mmsdomainname" is the "Domain Name for MMS E-Mail" defined on the "MMSC" dialog. E-mail attachments that are supported MMS content types (included in the `MMSCTYPE.INI` file) will be packaged and included in the MMS message sent to the mobile phone.

By default, the e-mail to MMS gateway will only accept inbound e-mail messages addressed to one of the users defined to the MMSC. It is also possible to use the SMTP interface for the bulk sending of MMS (and SMS) messages by logging into the SMTP server using an e-mail client that supports SMTP Authentication. When logged in via SMTP authentication, it is possible to send an MMS (or SMS) message to any recipient, by sending to addressing the message to `phonenumber@mms.domain.name`, where "mms.domain.name" is the "Domain Name for MMS E-Mail" defined on the [MMSC configuration dialog](#). An authenticated SMTP user can send an SMS message by addressing the message to `phonenumber@sms.domain.name`, where "sms.domain.name" is the "Domain Name for SMS E-Mail" defined on the [MMS configuration dialog](#). To define a user account that is allowed to login with SMTP Authentication, refer to the ["SMS Users"](#) configuration dialog.

Sending Voice Mail Notification Messages

To send an SMS voice mail notification message via a menu driven interface, please see the help section titled "[Web Menu Interface](#)". This section describes how to send a voice mail notification message programmatically via URL parameters.

Voice Mail Notification Messages are special SMS messages that are used to tell the user that they have voice mail waiting. On most mobile phones, the phone displays a message prompt, and the user can press a single key to be transferred to voice mail. This voice mail phone number is configurable via the mobile phone settings.

Voice Mail Notification Messages would be most often used in conjunction with voice mail systems. For example, a user may wish to combine their mobile phone voice mail with their office voice mail in a single voice mailbox. One way of accomplishing this is to configure the mobile phone to forward to an office phone number when the mobile phone is busy or unavailable, instead of the standard setting of forwarding to the mobile voice mail system (this setting can be configured via the mobile phone). If the user is unavailable, the office voice mail system assumes responsibility for accepting a voice mail message. The office voice mail system would be configured to make a request via the SMS gateway to turn on and off voice mail notifications for the mobile phone user.

<http://127.0.0.1:8800/?PhoneNumber=xxxxxxx&VoiceMail=On>

For 127.0.0.1, please substitute the IP address or host name assigned to your gateway PC. (Note: 127.0.0.1 is a local loopback address that can be utilized when you are connecting to the gateway from the same computer.)

For 8800, please substitute the port number that the gateway is configured to use.

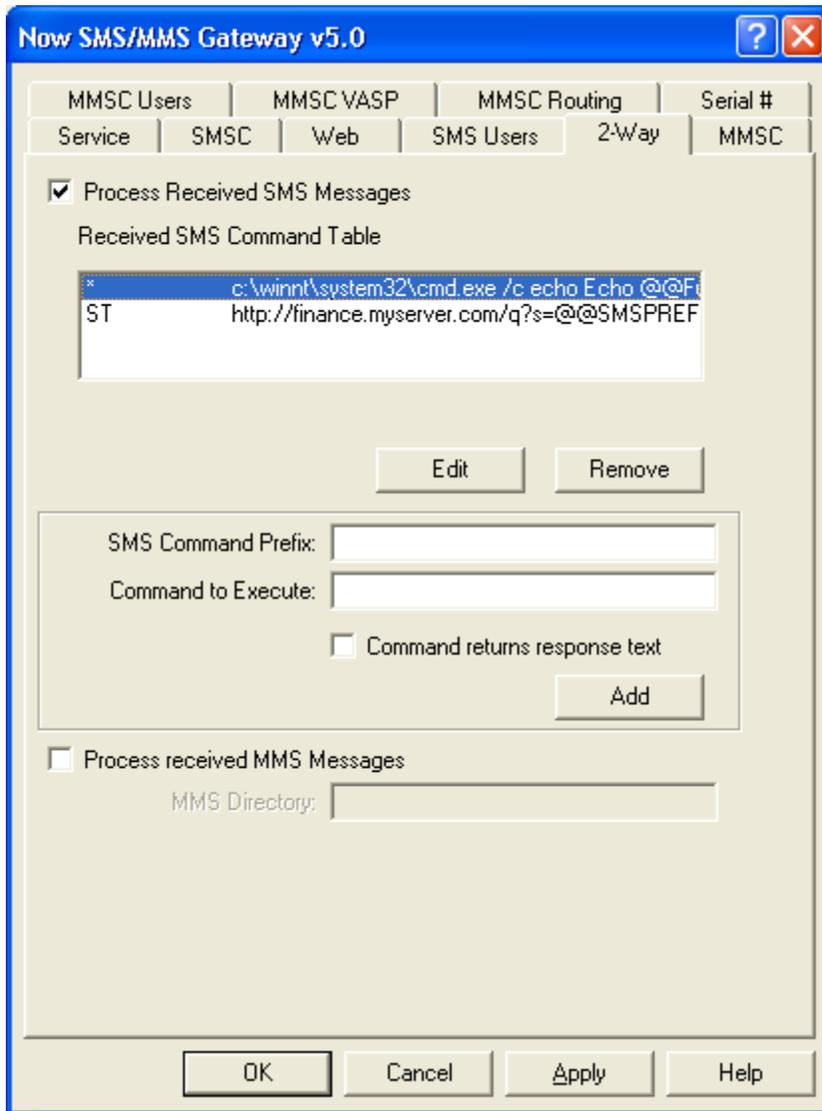
Substitute the phone number that you wish to send the SMS message to for the "xxxxxxx" in the "PhoneNumber" parameter. Use either the local phone number format, or the international phone number format (your network provider may or may not allow you to send to international phone numbers). If the international phone number format is used, note that you must substitute "%2B" for the "+" character, because of URL escaping restrictions. For example, to send an SMS to +447778001210, use %2B447778001210 instead.

The "VoiceMail" parameter can be either "On" or "Off", turning the notification on or off respectively.

For a complete list of URL parameters, please refer to the section ["URL Parameters for Sending Messages"](#).

2-Way SMS Support

The “2-Way” tab of the gateway configuration dialog contains settings relevant to the creation of 2-way applications that can receive SMS messages, and return a response based upon the content of the received SMS message.



The “**Process Received SMS Messages**” checkbox must be checked in order to enable the gateway to receive and process SMS messages.

When an SMS message is received, the gateway will evaluate the content of the message, and can either execute a program, or connect to an HTTP URL, based upon the content of the message. The decision of how to process a received message is based upon the first “word” of the received SMS message. In the

terminology of the gateway, this first word of the received message is called the **"SMS Command Prefix"**. Based upon this "SMS Command Prefix", the gateway will execute a command associated with the prefix. If the received prefix does not match any defined prefix, then it is considered to be a match for a special wildcard prefix, denoted as **"*"**.

When a command is executed based upon the receipt of an inbound message, the command line for the program or HTTP request can include replaceable parameters from the contents of the SMS message. The following replaceable parameters are supported:

@@SENDER@@	The phone number of the sender of the SMS Message.
@@SMSPREFIX@@	The first word of the SMS message.
@@SMS@@	The content of the SMS message, except the first word of the message.
@@FULLSMS@@	The complete content of the SMS message.
@@RECIP@@	If available, the phone number that is intended to receive (or did receive, if the message was received via a GSM modem connection) this message.

To return results back to the user, the command can either return a simple text response directly to the gateway (**"Command returns response text"** is checked), or the command can generate a more complex response to the gateway via a separate HTTP request to the gateway. An executable program returns a simple text response to the gateway by printing results to the screen, where the gateway captures the screen output, and generates an SMS response to send the screen output text back to the sender via SMS. An HTTP request returns a simple text response to the gateway by returning content of the MIME type "text/plain".

The example dialog above illustrates a simple command that can be used for testing this 2-way capability.

Define an **"SMS Command Prefix"** of **"*"** (wildcard), or any prefix of your choosing.

Define **"Command to Execute"** as **"c:\winnt\system32\cmd.exe /c echo Echo @@FULLSMS@@"**.

Check **"Command returns response text"**.

Press **"Add"** to add the command to the **"Received SMS Command Table"**.

When an SMS is received that matches this SMS command prefix (in the case of "*", any SMS that doesn't match another defined prefix), the gateway launches a command processor (CMD) that simply echoes the text back to the screen adding the word "Echo" to the beginning of the received text. In this example, the sender of the SMS message will receive an "Echo" back of the command that they sent in to the gateway. While not an extremely useful command, this is a useful way of testing to see that the gateway is alive and capable of receiving SMS messages.

In addition to supporting the launch of command line programs, the "**Command to Execute**" field can also be an HTTP command, causing the gateway to connect via HTTP to another application server to inform the application server of details regarding the received message. For example, `http://server:port/path?sender=@@SENDER@@&message=@@FULLSMS@@`.

When an HTTP command is used, if the command is to return a response to the gateway directly, the HTTP response must be of the MIME content type "text/plain".

It is also possible for any HTTP command to return an HTTP redirect response to the gateway, which instructs the gateway to fetch an alternative URL, even a URL command that contains parameters to tell the gateway to submit a message. This can be useful for creating a 2-way command script that responds with binary message content.

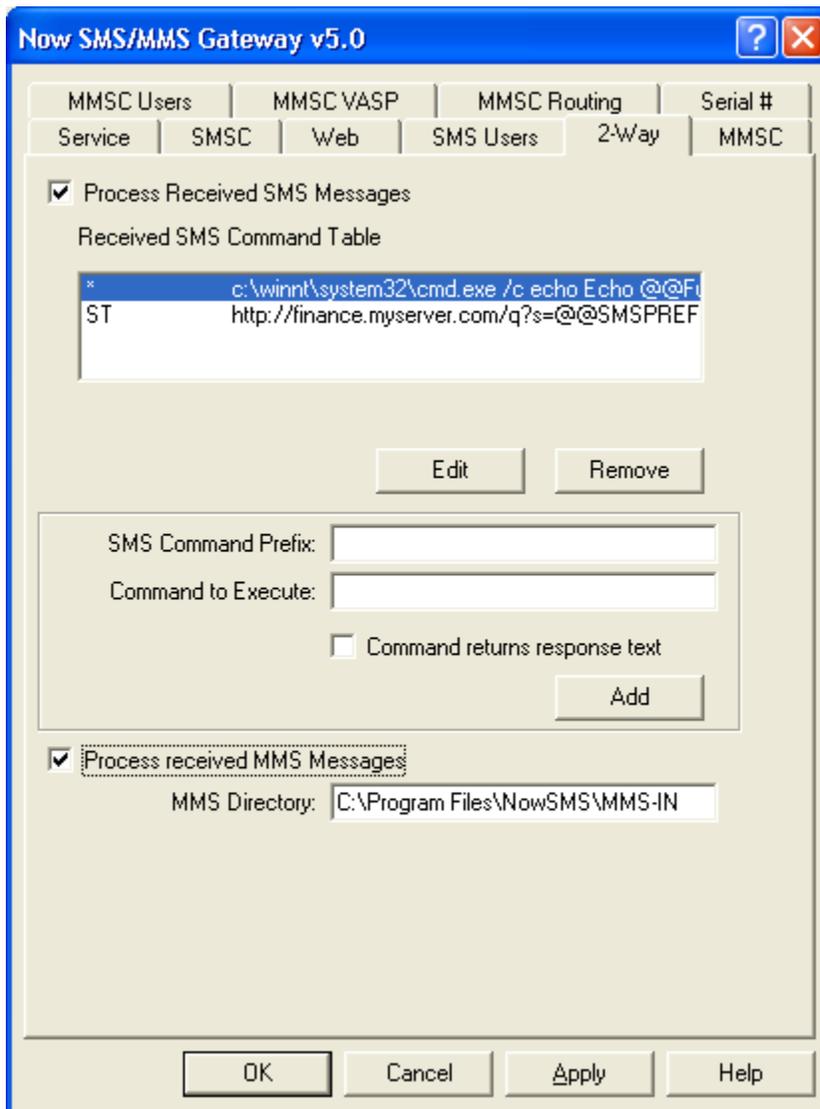
If an HTTP command requires HTTP authentication with a username and password, the URL format of "http://username:password@host.name/path" is supported. When a URL command is defined in this format, the gateway will connect to "http://host.name/path" using an authorization header of the specified username and password.

The "Command to Execute" field can also specify an e-mail address, in which case any received SMS messages that match the configured "SMS Command Prefix" will be forwarded to the specified e-mail address. To specify an e-mail address for the "Command to Execute", use the format "mailto:user@domain.com".

If the wildcard SMS command prefix is not associated with any command, any inbound SMS messages that do not match a prefix will be saved to the SMS-IN directory with a file extension of ".SMS", and they may be processed by another application independent of the gateway.

2-Way MMS Support

Support for processing received MMS messages is slightly more complex than received SMS messages, because the content of MMS messages is more complex, and the network configuration information to receive MMS messages is also considerably more complex. To enable received MMS messages to be processed, on the “2-Way” tab of the Configuration dialog, both “Process Received SMS Messages” and “Process Received MMS Messages” must be checked.



The Now SMS/MMS Gateway can receive MMS messages via two different types of connections.

- 1.) MMS messages can be received via a direct connection to an operator MMSC using one of the supported protocols, including MM7, MM4 or EAIF. When any of these protocols are used, the operator MMSC will automatically connect to your Now SMS/MMS Gateway to deliver messages. For more information on this type of connection, please refer to [Connecting to an Operator MMSC - Receiving MMS Messages](#).
- 2.) MMS messages can be received using a combination of SMS and WAP technologies. When an MMS message is sent to a mobile phone, the mobile phone first receives an MMS notification message over SMS. The Now SMS/MMS Gateway can receive this MMS notification over SMS, and then retrieve the content of the MMS message over WAP or the internet, just as it would be received by a mobile device. For more information on this type of connection, please refer to [Connecting to an Operator MMSC - Using a GPRS Modem](#).

There are different options for the processing of received MMS messages. Because MMS messages typically contain multiple objects, as opposed to the simple text string of an SMS message, they are routed to applications via different interfaces.

The routing choices for received MMS messages is configured separately for each potential connection through which the gateway can receive MMS messages.

When the gateway is receiving MMS messages via an operator MMSC, an account is defined in the "MMSC VASP" page of the configuration dialog, and the option for how to route MMS messages received over this connection is defined as part of the account definition.

When the gateway is receiving MMS messages via an SMSC or GSM modem connection, the "Properties" dialog for the SMSC configuration will include an "MMS Settings" dialog.

The choices for how received MMS messages are processed are:

"Receive to MMS-IN Directory" - In this case, the MMS message is parsed into text format, and individual components of the message are extracted into separate files. A ".hdr" file is written to the MMS-IN directory which contains a text version of the MMS header. This header file includes "X-NowMMS-Content-Location:" headers that point to the different file components that are included in the MMS message (text, images, etc.). These additional file components are stored in a subdirectory, and the location specified in the header is relative to the MMS-IN directory. The format of these ".hdr" files is consistent with the format used by the MMSCOMP utility, which is described in the section titled [Creating MMS Message Files - MMSCOMP](#).

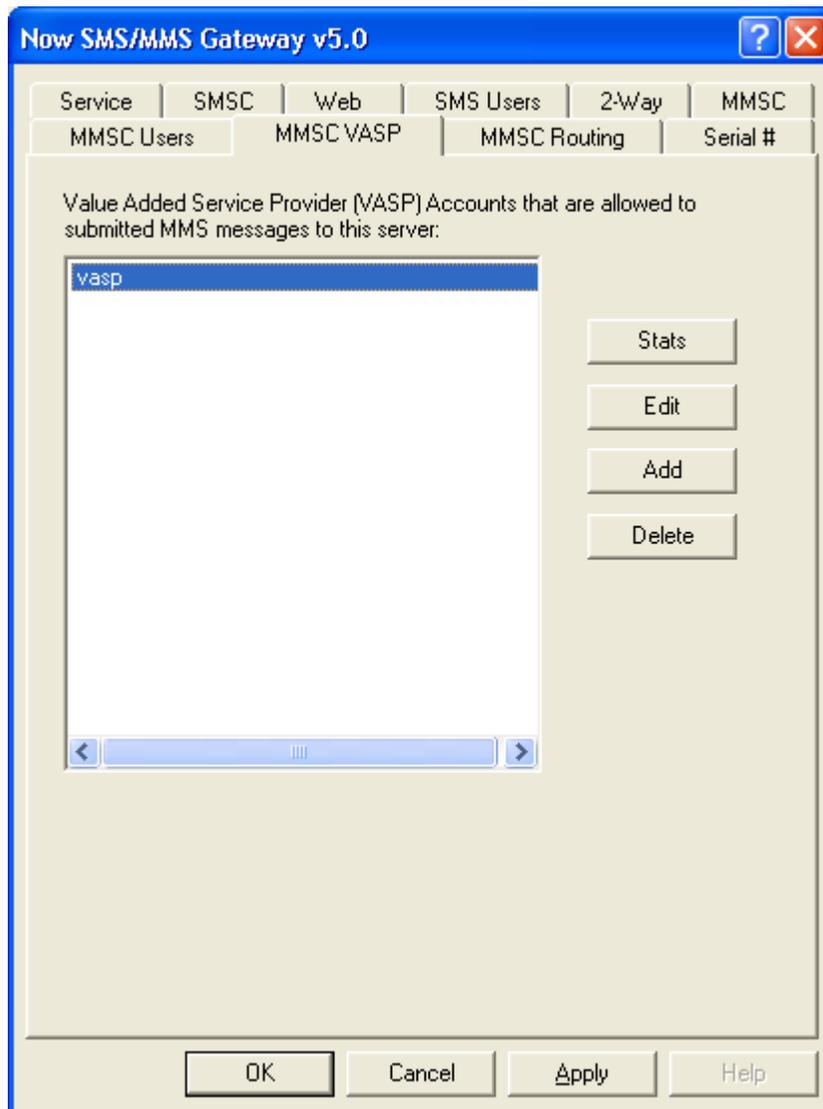
"Route via MM7" - MM7 is an XML/SOAP interface that is defined by the 3GPP as a format for applications to send/receive MMS messages. When this type of routing is defined for received MMS messages, NowSMS reformats the MMS message into MM7 format, and performs an HTTP POST of the content to the specified MM7 connection. You must first define an MM7 connection on the "MMSC Routing" page of the configuration dialog before you will be able to select this option for routing a received MMS message.

"Forward to E-Mail Address" - The MMS message is converted into an e-mail message (with components of the MMS message converted to attachments) and sent to a specified e-mail address. (Note: This option requires that a valid "SMTP Relay Host" be configured on the "MMSC" page of the configuration dialog. NowSMS will connect to the SMTP server specified as this relay host in order to send the message.)

Configuring MMS VASP Accounts

Value Added Service Providers (VASPs) are external application services that need to submit MMS messages to the Now SMS/MMS Gateway.

A VASP could be a customer that is submitting MMS messages through your gateway, or in some cases, a VASP account could be an operator MMSC delivering MMS messages to your gateway. VASP accounts are defined on the "MMSC VASP" page of the configuration dialog.



When you "Add" or "Edit" a VASP connection, the following screen is displayed:

VASP Account

Account Name:

(Note: Specify IP address as name if account does not require a login.)

Password:

Account Description:

IP Address Restrictions:

Accept Connections via: MM7 MM4 (SMTP)
 MM1 (EAlF)

VASP Sender Address:

Allow Sender Address Override

MMSC Routing for Received Messages:

Standard MMS Delivery

Receive to MMS-IN Directory

Route via MM7

Forward to E-Mail Address

Enable Credit Balance

Credits to add:

Enable Message Sending Limits for this account

Max messages per day:

Max messages per month:

OK Cancel

“Account Name” specifies a login name for the VASP to login to your Now SMS/MMS Gateway.

In some cases, where a VASP does not have configuration settings to login to the gateway, you should specify an IP address. When an IP address is specified as the “Account Name”, the Now SMS/MMS Gateway will treat any connections from the specified IP address as being from this defined VASP.

“Password” specifies a password for the VASP to login to your Now SMS/MMS Gateway.

“Account Description” is a descriptive field that can be associated with the VASP. It is not used for any purposes of the VASP making a connection to the gateway.

“IP Address Restrictions” specifies one or more IP addresses from which the account is allowed to login to the MMSC. If multiple IP addresses are specified, they should be separated by a comma only, with no white space characters. The “*” character is permitted as a wildcard to allow access from all IP addresses on a particular subnet.

VASPs can connect to the MMSC using any of the following protocols: MM7, MM4, MM1 or EAIF. The “Accept Connections via” setting allows you to select the protocols that this particular VASP is allowed to use when connecting to the MMSC. (Note: As EAIF and MM1 are very similar protocols, if MM1 is enabled, EAIF is automatically enabled, and vice versa.)

When a VASP connects to the MMSC using the [MM7](#) protocol, it should make connections to the configured “HTTP Port Number” on the “MMSC” page of the configuration dialog, using a URL path of “/mm7” (e.g., http://host.name:port/mm7). Unless an IP Address is specified as the “Account Name”, the application should authenticate to the MMSC using HTTP Basic Authentication using the configured “Account Name” and “Password” for the VASP account.

When a VASP connects to the MMSC using the [MM4](#) protocol, it should make connections to the configured “SMTP Port Number” on the “MMSC” page of the configuration dialog. Unless an IP address is specified as the “Account Name”, the application should use “SMTP AUTH” to authenticate to the MMSC.

When a VASP connects to the MMSC using the [EAIF](#) protocol, it should make connections to the configured “HTTP Port Number” on the “MMSC” page of the configuration dialog, using a URL path of “/eaif”. Unless an IP Address is specified as the “Account Name”, the application should authenticate to the MMSC using either HTTP Basic Authentication, or it can include the username and password in the URL using the following format (http://host.name:port/eaif/username=password).

When a VASP connects to the MMSC using the [MM1](#) protocol, it should make connections to the configured “HTTP Port Number” on the “MMSC” page of the configuration dialog, using a URL path of “/mm1”. Unless an IP Address is specified as the “Account Name”, the application should authenticate to the MMSC using either HTTP Basic Authentication, or it can include the username and password in the URL using the following format (http://host.name:port/mm1/username=password).

When the Now SMS/MMS Gateway receives a message from a VASP account, four options are available for processing the received message.

“Standard MMS Delivery” means the message will be queued for outbound delivery by the MMSC as would any other MMS message submitted to the gateway.

“Receive to MMS-IN Directory” means that the message will be received and stored to the “MMS-IN” subdirectory of the gateway as a received message file, and will not be delivered externally to the gateway.

“Route via MM7” means that the message will be received and routed to an MM7 connection that is defined in the “MMSC Routing” page of the configuration dialog.

“Forward to E-Mail Address” means that the message will be forwarded to a specified e-mail address.

Submitting MMS Messages to the Gateway

Applications can submit MMS messages to the Now SMS/MMS Gateway via any of the following protocols, which are described below:

- 1.) [Now SMS/MMS Proprietary URL Submission](#) via HTTP GET or POST
- 2.) [MM7](#) - The MMS standard for applications to submit MMS messages to an MMSC. MM7 is an XML/SOAP based API where the MMS message is formatted in a MIME encoded XML document and posted to the server using an HTTP POST.
- 3.) [MM4](#) - The MMS standard for connectivity between multiple MMSCs. MM4 is an SMTP based interface where the MMS message is posted to the server as a standard MIME encoded e-mail message. While the interface exists primarily for connectivity between multiple MMSCs, the Now SMS/MMS Gateway also makes this interface available to application developers that are more comfortable with the SMTP protocol.
- 4.) [MM1](#) - The MMS standard for phones to send and receive MMS messages from an MMSC. This is an HTTP based protocol where applications can submit binary encoded MMS messages encoding according to the MMS Encapsulation Specification (application/vnd.wap.mms-message MIME type) to the gateway using HTTP POST.
- 5.) [EAIF](#) - This is a Nokia proprietary variation on the MM1 protocol which was defined as an interface for submitting messages to a Nokia MMSC. The interface is functionally similar to MM1, with additional HTTP headers defined.

Now SMS/MMS Proprietary URL Submission

The Now SMS/MMS Proprietary format for submission of an MMS message is the interface that is used by the "Send MMS Message" form in the web menu interface of the gateway.

To submit a message via this interface, a user account must be specified on the "SMS Users" page of the configuration dialog.

To submit it the same way that the "Send MMS Message" form does in the gateway's web menu interface, you need to do an HTTP POST in the "multipart/form-data" MIME type.

Basically, when you POST, it would look something like this:

```
POST / HTTP/1.0
Accept: */*
Content-type: multipart/form-data; boundary="--boundary-border--"
Content-length: xxxxx (size of content part of post)
Authorization: username:password (base64 encoded)

----boundary-border--
Content-Disposition: form-data; name="PhoneNumber"

+448080148324
----boundary-border--
Content-Disposition: form-data; name="MMSFrom"

sender@domain (or +38484753009)
----boundary-border--
Content-Disposition: form-data; name="MMSSubject"

Message Subject
----boundary-border--
Content-Disposition: form-data; name="MMSText"

An optional text part of the message.
----boundary-border--
Content-Disposition: form-data; name="MMSFile"; filename="original-filename.ext"
Content-type: Mime/Type

File data goes here
----boundary-border----
```

The content-type for the overall message is "multipart/form-data". As with other multipart MIME encoding, you must include a boundary that separates the multiple parts of the message.

It is very important to set the Content-length: field to specify the length of the multipart content that follows (this is how the server knows your HTTP post is complete).

The Authorization header specifies the username and password used to login to the gateway. It is in the format "username:password" and is Base64 encoded.

Then, the content has a part for each form variable.

The "PhoneNumber" variable is required, it is the phone number of the message recipient.

The "MMSFrom" variable is optional. It is the "From:" address to be used in the MMS message. (If sending a pre-compiled MMS message, this is used for the notification only.)

The "MMSSubject" variable is optional. It is the "Subject" line used in the MMS message. (If sending a pre-compiled MMS message, this is used for the notification only.)

The "MMSText" variable is optional. It contains a text part of the message.

The "MMSFile" variable is optional, and can be repeated multiple times. It contains binary file content for an uploaded file. If you're sending a pre-compiled MMS file, you'd only include the "MMSFile" variable once. If you're sending a message for the gateway to compile, you could include each of the individual message parts as a separate instance of the "MMSFile" variable.

As an alternative to using the HTTP POST, if the content of the MMS message already exists on a web server, the "MMSFile" variable can be specified as a URL instead of the actual file content. In this case, the message can be submitted to the gateway with an HTTP GET request, instead of requiring HTTP POST. For example:

<http://127.0.0.1:8800/?PhoneNumber=xxxxxx&MMSFrom=sender@domain&MMSSubject=Message+Subject&MMSText=An+optional+text+part+of+the+message&MMSFile=http://www.nowSMS.com/media/logo.gif>

The variables are the same as described above, except that in a GET request, the "MMSFile" variable must point to a URL. As with the POST request, the "MMSFile" variable can be repeated to specify multiple content files.

MM7

Before submitting a message via MM7, a VASP (Value Added Service Provider) account must be defined to the Now SMS/MMS Gateway. This account is defined on the "MMSC VASP" page of the configuration dialog.

To post to the Now SMS/MMS Gateway via MM7, you must connect to the HTTP port configured for the MMSC on the "MMSC" page of the configuration dialog. And you must perform an HTTP POST of the MM7 content to a URI of "/mm7", which is how the gateway knows that the VASP intends to submit in the MM7 format.

The HTTP headers of your POST must include a "Content-length:" header, and it needs to either include an "Authorization:" header for Basic authentication using the username and password configured for the account, or it must originate from an IP address that matches the name configured for the VASP account. (If your software cannot generate an "Authorization:" header, it is possible to configure the account name for the VASP as an IP address, and in this case, the MMSC will recognise any connections from that IP address as being for this VASP account.)

The "Content-type:" header in the POST should be one of the "multipart" types (usually "multipart/related"), and should include a "boundary=" parameter that delimits the different parts of the message.

The first part of the multipart message is expected to be the XML for the MM7 request, and we're going to expect to see a <Recipients> section with at least one <To>, <Cc> or <Bcc> recipient specified.

The second part of the multipart message is expected to be the content for the MMS message, and this in turn will usually be another MIME multipart structure.

The following example is adapted from the official MM7 specification that is included in the 3GPP TS 23.140 specification:

Note that this example does not include a SMIL file, and as part of the MMS content, you would probably want to include a SMIL file (application/smil), which this example does not include.

Also note that the MM7 XML portion of the document (the first part of the main multipart content) should not use any Content-Transfer-Encoding, it should always be expressed without any encoding. For the portion of the document that includes the MMS content itself, you can use Content-Transfer-Encoding of either quoted-printable or base64, or no encoding can be specified in which case it is assumed that the binary data is to be included as is.

```

POST /mm7 HTTP/1.1
Host: mms.omms.com
Content-Type: multipart/related; boundary="NextPart_000_0028_01C19839.84698430";
type=text/xml; start="</tnn-200102/mm7-submit>"
Content-Length: nnnn
SOAPAction: ""

--NextPart_000_0028_01C19839.84698430
Content-Type:text/xml; charset="utf-8"
Content-ID: </tnn-200102/mm7-submit>

<?xml version="1.0" ?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
<env:Header>
<mm7:TransactionID
xmlns:mm7="http://www.3gpp.org/ftp/Specs/archive/23_series/23.140/schema/REL-5-M M7-1-3"
env:mustUnderstand="1">
vas00001-sub
</mm7:TransactionID>
</env:Header>
<env:Body>
<SubmitReq xmlns="http://www.3gpp.org/ftp/Specs/archive/23_series/23.140/schema/REL-5-
MM7-1 -3">
<MM7Version>5.6.0</MM7Version>
<SenderIdentification>
<VASPID>TNN</VASPID>
<VASID>News</VASID>
</SenderIdentification>
<Recipients>
<To>
<Number>7255441234</Number>
</To>
<Cc>
<Number>7255443333</Number>
</Cc>
<Bcc>
<RFC2822Address>7255444444@OMMS.com</RFC2822Address>
</Bcc>
</Recipients>
<ServiceCode>gold-sp33-im42</ServiceCode>
<LinkedID>mms00016666</LinkedID>
<MessageClass>Informational</MessageClass>
<TimeStamp>2002-01-02T09:30:47-05:00</TimeStamp>
<EarliestDeliveryTime>2002-01-02T09:30:47-05:00</EarliestDeliveryTime>
<ExpiryDate>P90D</ExpiryDate>
<DeliveryReport>>true</DeliveryReport>
<Priority>Normal</Priority>
<Subject>News for today</Subject>
<ChargedParty>Sender</ChargedParty>
<DistributionIndicator>>true</DistributionIndicator>
<Content href="cid:SaturnPics-01020930@news.tnn.com" allowAdaptations="true"/>
</SubmitReq>
</env:Body>
</env:Envelope>

--NextPart_000_0028_01C19839.84698430
Content-Type: multipart/mixed; boundary="StoryParts-74526-8432-2002-77645"
Content-ID:<SaturnPics-01020930@news.tnn.com>

--StoryParts-74526-8432-2002-77645
Content-Type: text/plain; charset="us-ascii"

Science news, new Saturn pictures...

--StoryParts-74526-8432-2002-77645
Content-Type: image/gif
Content-ID:<saturn.gif>
Content-Transfer-Encoding: base64

```

R01GODdhZAAwAOMAAAAAIGJjGltcDE00OfWo6Ochbi1n1pmcbGojpKbnP/lpW54fBMTE1RYXEFO

...

--StoryParts 74526-8432-2002-77645--
--NextPart_000_0028_01C19839.84698430--

MM4

Before submitting a message via MM4, a VASP (Value Added Service Provider) account must be defined to the Now SMS/MMS Gateway. This account is defined on the "MMSC VASP" page of the configuration dialog.

To post to the NowSMS MMSC via MM4, your application is making an SMTP connection to the MMSC, and you would connect to the "SMTP Port Number" that is configured on the "MMSC" page of the configuration dialog.

To submit a message via MM4, the VASP must authenticate via SMTP using the "AUTH LOGIN" approach, or similar to the way it works for MM7, the VASP account can be created with an IP address as the "Account Name", and in that case any connection from that IP will be accepted as being a connection from that VASP account.

"AUTH LOGIN" is rather simple. Basically, as part of the SMTP dialog, after the initial HELO or EHLO command, the SMTP client needs to issue the command AUTH LOGIN. The SMTP server responds with a "300" series code and prompts for the account name (the prompt after the code is a BASE64 encoded string). The client sends the account name as a BASE64 encoded string. The SMTP server responds with another "300" series code prompting for the password, and the client responds with the password as a BASE64 encoded string. A "200" series response indicates that the authentication was accepted, a "500" series response indicates that it was not.

The SMTP dialog then continues as normal, generally with a "MAIL FROM:" command from the client indicating the sending address of the message, followed by one or more "RCPT TO:" commands to indicate the recipients for the message. Note that NowSMS expects the "RCPT TO:" addresses to be in a format of `phonenumber@domain.name` or `phonenumber/TYPE=PLMN@domain.name`, where "domain.name" is the "Domain Name for MMS E-Mail" configured on the "MMSC" page of the configuration dialog. (The "Local Host Name or IP Address" value is also acceptable here.) If the domain name is not present, the MMSC will reject the recipient. As special support for Multimedia WAP Push, the following address formats are also supported to specify Multimedia WAP Push to be used for sending to the phone number: `phonenumber.wappush@domain.name` or `phonenumber/TYPE=WAPP@domain.name`.

The actual MMS message is then transmitted via the "DATA" command. Normally an MMS message would be a multipart MIME message with multiple content parts, although NowSMS will also accept a message that includes only a single part.

The MM4/SMTP dialog looks something like this:

The dialog looks something like this (IN means from server, OUT means from client):

```
IN: 220 SMTP Ready
OUT: HELO client.name (or EHLO client.name)
IN: 250 OK (or a multiline response if EHLO was used)
OUT: AUTH LOGIN
IN: 334 VXNlcm5hbWU6
("Username:" BASE64 encoded)
OUT: dGVzdA==
("test" BASE64 encoded)
IN: 334 UGFzc3dvcmQ6
("Password:" BASE64 encoded)
OUT: dGVzdA==
("test" BASE64 encoded)
IN: 235 Ok
OUT: MAIL FROM: <username@domain.com>
IN: 250 Ok
OUT: RCPT TO: <+447778889999/TYPE=PLMN@mms.domain.com>
IN: 250 Ok
OUT: DATA
IN: 354 Ok, end with "." on a new line...
OUT: (Transmit MIME encoded message, then end with a line with only the . character)
IN: 250 Message Accepted
OUT: QUIT
```

Using the example data from the MM7 message above, the MIME encoded message would look something like this:

```
To: +447778889999/TYPE=PLMN@mms.domain.com
From: username@domain.com
Subject: News for today
Content-Type: multipart/mixed; boundary="StoryParts-74526-8432-2002-77645"
Content-ID: <SaturnPics-01020930@news.tnn.com>

--StoryParts-74526-8432-2002-77645
Content-Type: text/plain; charset="us-ascii"

Science news, new Saturn pictures...

--StoryParts-74526-8432-2002-77645
Content-Type: image/gif
Content-ID: <saturn.gif>
Content-Transfer-Encoding: base64

R01GODdhZAAwAOMAAAAAIGJjGltcDE00fWo6Ochbi1n1pmcbGojpKbnP/lpW54fBMTE1RYXEFO
...

--StoryParts-74526-8432-2002-77645--
```

MM1

Before submitting a message via MM1, a VASP (Value Added Service Provider) account must be defined to the Now SMS/MMS Gateway. This account is defined on the "MMSC VASP" page of the configuration dialog.

To post to the Now SMS/MMS Gateway via MM1, you must connect to the HTTP port configured for the MMSC on the "MMSC" page of the configuration dialog. And you must perform an HTTP POST of the MM1 content to a URI of "/mm1", which is how the gateway knows that the VASP intends to submit in the MM1 format.

The HTTP headers of your POST must include a "Content-length:" header. To specify the username and password of the VASP account, you must either include an "Authorization:" header for Basic authentication using the username and password configured for the account, or the username and password can be specified on the request URI (e.g., /mm1/username=password). Alternatively, the request must originate from an IP address that matches the name configured for the VASP account. (If your software cannot generate an "Authorization:" header, it is possible to configure the account name for the VASP as an IP address, and in this case, the MMSC will recognise any connections from that IP address as being for this VASP account.)

The "Content-type:" header in the POST should be "application/vnd.wap.mms.message", and the message should be a binary MMS message of the m-send-req format, formatted according to the MMS Encapsulation Specification, published by the Open Mobile Alliance.

Consistent with HTTP specifications, a "400" or "500" series HTTP response would be considered an error condition. The MMSC might return a "200" series response even if an error did occur, in which case information would be included in the "X-Mms-response-status" field of the MM1 response (MIME type "application/vnd.wap.mms-message", X-Mms-Message-Type of "m-send-conf") would contain a response-status value other than Ok.

EAIF

Before submitting a message via EAIF, a VASP (Value Added Service Provider) account must be defined to the Now SMS/MMS Gateway. This account is defined on the "MMSC VASP" page of the configuration dialog.

To post to the Now SMS/MMS Gateway via EAIF, you must connect to the HTTP port configured for the MMSC on the "MMSC" page of the configuration dialog. And you must perform an HTTP POST of the EAIF content to a URI of "/eaif", which is how the gateway knows that the VASP intends to submit in the EAIF format.

The HTTP headers of your POST must include a "Content-length:" header. To specify the username and password of the VASP account, you must either include an "Authorization:" header for Basic authentication using the username and password configured for the account, or the username and password can be specified on the request URI (e.g., /eaif/username=password). Alternatively, the request must originate from an IP address that matches the name configured for the VASP account. (If your software cannot generate an "Authorization:" header, it is possible to configure the account name for the VASP as an IP address, and in this case, the MMSC will recognise any connections from that IP address as being for this VASP account.)

The "Content-type:" header in the POST should be "application/vnd.wap.mms.message", and the message should be a binary MMS message of the m-send-req format, formatted according to the MMS Encapsulation Specification, published by the Open Mobile Alliance.

MMS message recipients can be specified either in the MMS message content itself, or in the "X-Nokia-MMSC-To:" header.

Consistent with HTTP and EAIF specifications, a "400" or "500" series HTTP response would be considered an error condition. The expected response for a valid message submission would be an HTTP "204" response that includes an "X-Nokia-MMSC-Message-Id:" header. (In beta releases of the v5.0 Now SMS/MMS Gateway, the MMSC might return a "200" series response even if an error did occur, in which case information would be included in the "X-Mms-response-status" field of an MM1 response with the "X-Mms-Message-Type" field containing a response-status value other than Ok. We expect this to have been corrected prior to the v5.0 release.)

Connecting to an Operator MMSC

The Now SMS/MMS Gateway includes an MMSC which can function independent of an operator MMSC. However, in many scenarios it is useful to configure the gateway to send and/or receive MMS messages via an operator MMSC.

Connecting to an Operator MMSC - Using a GPRS Modem

The Now SMS/MMS Gateway can use a GSM/GPRS modem to send and receive MMS messages via an operator MMSC. In this type of configuration, there is no special setup requirement required by the mobile operator. The Now SMS/MMS Gateway sends and receives MMS messages using the same protocol that is used by the MMS client in a mobile phone, so it simply requires that the SIM card in your mobile phone be provisioned by your mobile operator for MMS support.

It is also helpful to have an MMS compatible mobile phone that is already configured to send and receive MMS messages via the operator MMSC. In this way, you can check the settings on the mobile phone to determine the correct settings to configure in the Now SMS/MMS Gateway to allow the gateway to connect to your operator MMSC.

In this scenario, when sending an MMS message, the Now SMS/MMS Gateway initiates a GPRS connection to the mobile operator, it then makes a connection to the operator WAP gateway, and submits the MMS message over this WAP and GPRS connection to the mobile operator's MMSC.

When receiving an MMS message, the gateway first receives an MMS notification message via SMS. When this special notification message is received, the gateway initiates a GPRS connection to the mobile operator, and a connection to the operator WAP gateway over GPRS, in order to retrieve the content of the MMS message from the mobile operator's MMSC.

There are two areas where MMS settings specific to using a GSM/GPRS modem are specified, one is specific to sending messages, and the other is specific to receiving messages.

To send MMS messages to an operator MMSC via a GPRS modem connection, please see the description below for [Connecting to an Operator MMSC - Sending MMS Messages](#). This section focuses on setting for receiving MMS messages over a GPRS modem connection.

When a GSM phone or GSM modem receives an MMS message, what it actually receives is an MMS notification message. The MMS notification message arrives via an SMS message (usually two concatenated SMS messages). The MMS notification message contains header information about the MMS message, plus a URL pointer (e.g., <http://host.name/path/file.mms>) to the actual MMS content. In order to receive the MMS message, the receiving device must initiate an HTTP connection to retrieve the URL with the MMS message content (this URL generally points to content residing on the operator MMSC).

In earlier releases of the Now SMS/MMS Gateway, when the gateway received an MMS notification message, it would always attempt to retrieve the MMS message content via HTTP over the default internet connection of the PC running the gateway software. While this worked for some installations, this did not work for retrieving MMS message content from many operator MMSCs. The reason that it would not work in many installations is that the operator MMSCs were either firewalled off so that they were inaccessible from the internet, or they were on private IP addresses within the operator network.

The v5.0 release of the Now SMS/MMS Gateway includes configuration options that allow the gateway to automatically initiate a GPRS connection when an MMS notification message is received. Rather than retrieving the MMS message via HTTP, the gateway can make a connection to the operator WAP gateway to retrieve the MMS message content from the MMSC through the WAP gateway using WAP protocols. As almost all MMS compatible phones issue their MMS requests through a WAP gateway proxy, this allows the gateway software to appear just like a standard MMS compatible phone.

Note that you can still not use an MMS compatible phone as a GSM modem if you want to be able to receive MMS messages through the gateway. This is because an MMS compatible phone intercepts the MMS notification and tries to process it automatically, never forwarding it to the gateway. You must use a GSM/GPRS modem device to take advantage of this feature.

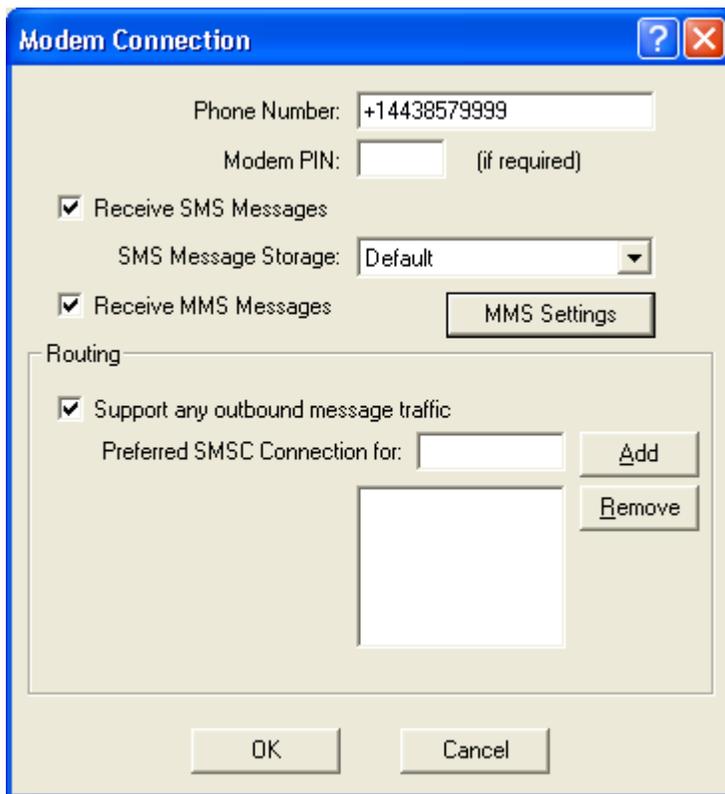
Before continuing, you will need to determine the GPRS APN (Access Point Name), the IP address of the WAP gateway, and the MMSC Message Server URL that are used for sending/receiving MMS via your operator network. These settings are operator dependent, and it may be advisable to check the MMS configuration settings on a working mobile phone to determine the correct settings. Note that your mobile operator possibly has multiple GPRS APNs and multiple WAP gateways, and you need the settings that are appropriate for MMS, not for WAP browsing or general internet connectivity.

With most GSM/GPRS modems (except for some PC card modems that allow GPRS access through a network connection without using dial-up networking), NowSMS cannot simultaneously use the modem for both sending/receiving SMS messages and accessing GPRS data.

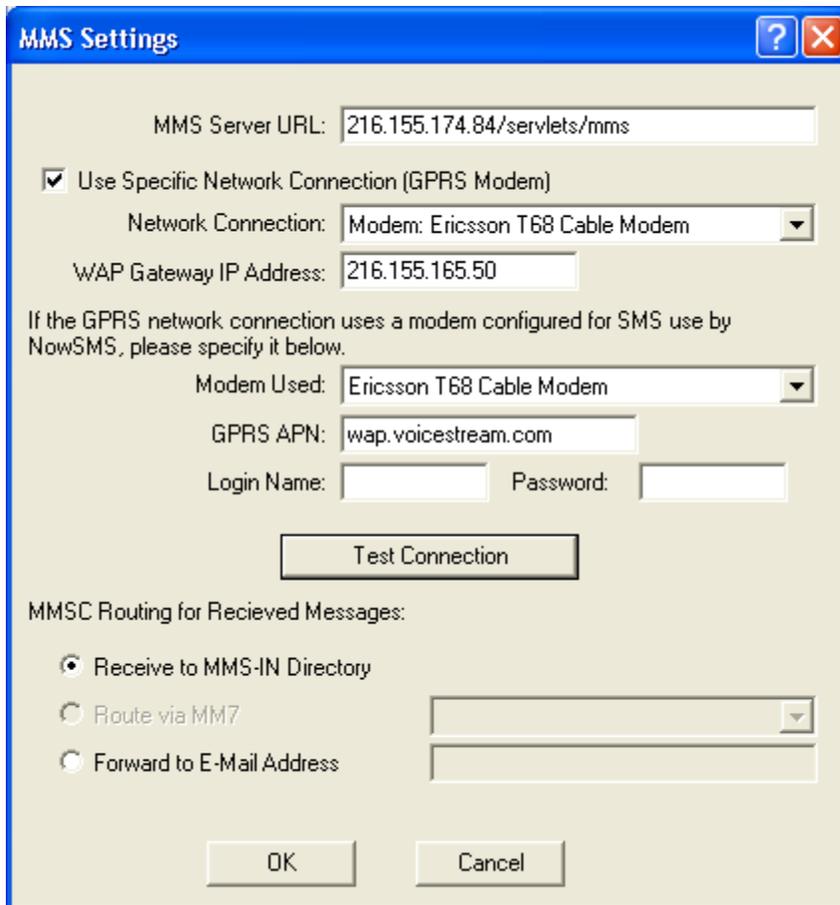
The gateway allows you to define separate settings for receiving MMS messages for each SMSC connection that is defined to your gateway. (Recall that the first stage of receiving MMS messages is that an MMS notification message is received via SMS.)

From the “SMSC” page of the Now SMS/MMS Gateway configuration dialog, highlight a connection for which you want to configure settings for receiving MMS messages, and select “Properties”.

The screen that is displayed will vary depending on the type of SMSC connection selected. But for any of the types of SMSC connections that can receive SMS messages, there will be a check box for whether or not the gateway should attempt to “Receive MMS Messages” for MMS notifications received via that SMSC connection.



To configure the “MMS Settings”, ensure that “Receive MMS Messages” is checked, and then select the “MMS Settings” button.



The “MMS Settings” dialog specifies how to receive MMS messages associated with any MMS notifications received over the SMSC connection.

The “MMS Server URL” is the URL address for the operator MMSC. While this setting is primarily used when sending MMS messages, the gateway will acknowledge MMS message receipt to this URL.

If the MMS message content needs to be received over a specific network connection, such as a GPRS modem, check the “Use Specific Network Connection (GPRS Modem)” checkbox.

In the “Network Connection”, select the name of the network connection that is to be used. NowSMS can use any of three different types of connections to make a GPRS connection. The different available connections are listed in the drop-down field associated with this configuration field. The different types of connections are prefixed with the text “Modem:”, “Dial-up:”, or “Network:”, and are described below:

a.) “Modem:” refers to a standard GPRS connection to be initiated over a GPRS modem. Select the modem that should be used for this connection. (Note that only modems that have a Windows modem driver defined for the modem can

be used. If your modem does not have a modem driver supplied by the manufacturer, you can use one of the "Standard" or "Generic" modem drivers available when defining a modem in the Windows Control Panel.)

b.) **"Dial-up:"** refers to a dial-up networking connection defined on the current PC. This setting can be used if you have advanced requirements and wish to create a custom dial-up networking profile.

c.) **"Network:"** refers to a particular network interface card installed in the PC. Some PC card GPRS modems, such as the Sierra Wireless Aircard 750 provide GPRS access via a network driver interface. To tell NowSMS to use that specific network driver for connecting to the MMSC, select the named "Network:" driver.

The **"WAP Gateway IP Address"** field should contain the IP address of the operator WAP gateway which will act as a proxy for connections to the MMSC.

For most GSM/GPRS modems, it is not possible for the Now SMS/MMS Gateway to connect to the modem to send/receive SMS messages as the same time as a GPRS connection is active (Network Connection type = "Modem:" or "Dial-up:"). For these situations, the "Modem Used" setting should specify the name of a modem that is used by the Dial-up networking profile. If the Now SMS/MMS Gateway is using the modem to send/receive SMS messages, it will automatically release the modem when it needs to initiate a GPRS connection, otherwise the GPRS connection will not be able to be properly established.

The **"GPRS APN"** field specifies the GPRS Access Point Name (APN) to be accessed for connecting to the MMSC. This setting is operator dependent, and it may be advisable to check the MMS configuration settings on a working mobile phone to determine the correct settings. Note that your mobile operator possibly has multiple GPRS APNs and multiple WAP gateways, and you need the settings that are appropriate for MMS, not for WAP browsing or general internet connectivity. Note that this setting is only available when using a Network Connection of type "Modem:". For other connection types, the GPRS APN must be configured for the connection using some means external to NowSMS.

The **"Login Name"** and **"Password"** parameters specify a username and password to be used for connecting to the GPRS network.

The **"Test Connection"** dialog verifies that the Now SMS/MMS Gateway can initiate a network connection to the specified profile, and that it can make a connection to the specified WAP gateway over the connection. (The "MMS Message Server URL" is not tested at this time.)

When the Now SMS/MMS Gateway receives an MMS message, three options are available for processing the received message.

“Receive to MMS-IN Directory” means that the message will be received and stored to the “MMS-IN” subdirectory of the gateway as a received message file, and will not be delivered externally to the gateway.

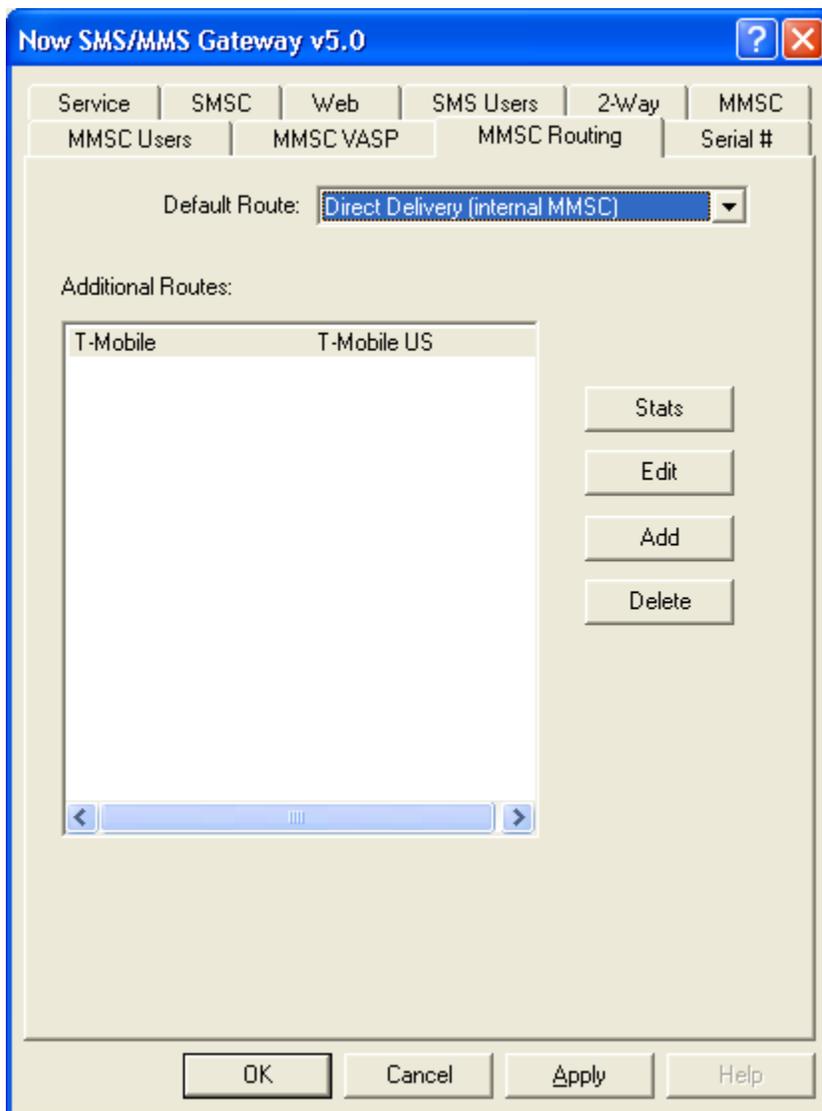
“Route via MM7” means that the message will be received and routed to an MM7 connection that is defined in the “MMSC Routing” page of the configuration dialog.

“Forward to E-Mail Address” means that the message will be forwarded to a specified e-mail address.

Connecting to an Operator MMSC - Sending MMS Messages

Similar to how the Now SMS/MMS Gateway allows recipient phone number masks to be defined to route SMS messages to different SMSC connections, this version of the gateway allows recipient phone number masks to be defined to route MMS messages to different MMSC connections.

The “MMSC Routing” page of the gateway configuration dialog specifies outbound routes for MMS messages.



By default, the Now SMS/MMS Gateway will act as an MMSC and perform direct delivery of MMS messages to recipients using a combination of SMS and WAP technologies.

However, there are certain closed operator configurations where it can be desirable to route MMS messages via an operator MMSC, or to reformat MMS message content to be delivered as a multimedia WAP push.

The “**Default Route**” setting specifies the default route to be used for delivering MMS messages. This setting can specify “Direct Delivery” utilizing the internal MMSC, “Convert MMS Message to Multimedia WAP Push”, or it can specify any of the other routes that are defined in the “Additional Routes” page.

For example, if you wanted to setup the gateway so that it routed all outbound MMS messages to an operator MMSC over a GPRS modem connection, you would first define this routing in the “**Additional Routes**” list, and then return to the “**Default Route**” prompt to select that routing as the default route for the system.

MMSC Routings can use any of the following MMSC connectivity protocols:

MM7 - An XML/SOAP based format for MMS messages to be transmitted using HTTP POST. This standard is defined by the 3GPP.

MM4 - An SMTP based format for MMS messages to be transmitted between MMSCs.

MM1 - A binary format for transmitting MMS messages using HTTP POST. This is the protocol that is used for phone to phone MMS, so if you are routing messages to an operator MMSC over a GPRS connection, this is the protocol that is used.

EAIF - This is a Nokia proprietary variation on the MM1 format that is used for sending messages to a Nokia MMSC.

MMS Outbound Routing

Account Name:

Account Description:

Default Sender Address:

Allow Sender Address Override

Route messages to this account for recipient phone number(s):

Route messages to VASP via:

- MM7
- MM4 (SMTP)
- MM1
- EAIF
- Direct Delivery (internal MMSC)
- Convert to Multimedia WAP Push

Server Address:

Login Name: Password:

E-Mail Domain:

Use Specific Network Connection (GPRS Modem)

Network Connection:

WAP Gateway IP Address:

If the GPRS network connection uses a modem configured for SMS use by NowSMS, please specify it below.

Modem Used:

GPRS APN:

Login Name: Password:

“Account name” and **“Account Description”** are settings that identify the connection to the Now SMS/MMS Gateway only. These settings are not transmitted externally.

“Default Sender Address” specifies the default sender address to applied to any MMS messages that are transmitted over this connection. The **“Default Sender Address”** is used only if the message being transmitted does not include a sender address. This address can either be a standard e-mail address, or a telephone number. If a telephone number is specified, it must be specified in MMS messaging format (+phonenumber/TYP=PLMN). If **“Allow Sender Address Override”** is checked, then submitted messages can include their own sender address. If this setting is not checked, the sender address for all messages

transmitted via the MMSC connection will be changed to the “Default Sender Address”.

When sending to an operator MMSC over GPRS, the operator MMSC will usually not allow a sender address other than the MSISDN of the sending phone. In these cases, it is usually best to leave the “Sender Address” field blank, and allow the MMSC to assign it automatically. To do this, leave the “**Default Sender Address**” field blank, and uncheck “**Allow Sender Address Override**”.

The “**Route messages to this account for recipient phone number(s)**” field is an address mask for defining which recipient phone numbers should be routed to this account. For example, “+44*” would route all messages for the UK country code (44) to this connection. Multiple address masks can be defined. When multiple address masks are defined, they should be separated by a comma only (,) and no white space characters.

If this field is left blank, no messages will be routed to this connection, except under special circumstances.

The “**Route Messages Via**” option defines the protocol that will be used for routing the MMS message. Messages can be routed to an external host using the MM7, MM4, MM1 or EAIF protocols. Other options for message routing include “**Direct Delivery**” via the gateway’s internal MMSC, or an option for the MMS to be converted into a Multimedia WAP Push message to be received by the WAP client on the recipient phone instead of the MMS client.

For MM7 connections, the “**Server Address**” field should contain a URL for posting to the MMSC. (The “http://” portion of the address is optional.) For example, http://192.168.1.1:8080/mm7. The “**Login Name**” and “**Password**” specify an optional login name and password that will be used to login to the MMSC using HTTP Basic Authentication. Other MMSC connectivity fields do not apply for MM7 connections.

For MM4 connections, the “**Server Address**” field should contain an IP address or host name of the MMSC. This field can also include a port number in the format, hostname:port, where “25” is the default port if a port number is not explicitly specified. The “**E-Mail Domain**” field specifies an e-mail domain that should automatically be appended to phone numbers when routing via this MMSC connection.

For EAIF connections, the “**Server Address**” field should contain a URL for posting to the MMSC. (The “http://” portion of the address is optional.) For example, http://192.168.1.1:8080/eaif. The “**Login Name**” and “**Password**” specify an optional login name and password that will be used to login to the MMSC using HTTP Basic Authentication. Other MMSC connectivity fields do not apply for EAIF connections.

For MM1 connections, the **“Server Address”** field should contain a URL for posting to the MMSC. (The **“http://”** portion of the address is optional.) For example, **http://192.168.1.1:8080/mm1**. Note that when you are using MM1 to connect with an operator MMSC over GPRS, this **“Server address”** is the **“MMS Server URL”** that would normally be configured on a mobile phone to connect with the operator MMSC. The **“Login Name”** and **“Password”** specify an optional login name and password that will be used to login to the MMSC using HTTP Basic Authentication, which is typically not used for a connection to an operator MMSC. As MM1 connections are frequently used to connect with an operator MMSC over a GPRS connection, additional parameters can be specified to have the gateway automatically initiate a GPRS connection and post the MMS message to an MMSC over GPRS through a WAP gateway.

If the MMS message content needs to be sent over a specific network connection, such as a GPRS modem, check the **“Use Specific Network Connection (GPRS Modem)”** checkbox.

In the **“Network Connection”**, select the name of the network connection that is to be used. NowSMS can use any of three different types of connections to make a GPRS connection. The different available connections are listed in the drop-down field associated with this configuration field. The different types of connections are prefixed with the text **“Modem:”**, **“Dial-up:”**, or **“Network:”**, and are described below:

a.) **“Modem:”** refers to a standard GPRS connection to be initiated over a GPRS modem. Select the modem that should be used for this connection. (Note that only modems that have a Windows modem driver defined for the modem can be used. If your modem does not have a modem driver supplied by the manufacturer, you can use one of the **“Standard”** or **“Generic”** modem drivers available when defining a modem in the Windows Control Panel.)

b.) **“Dial-up:”** refers to a dial-up networking connection defined on the current PC. This setting can be used if you have advanced requirements and wish to create a custom dial-up networking profile.

c.) **“Network:”** refers to a particular network interface card installed in the PC. Some PC card GPRS modems, such as the Sierra Wireless Aircard 750 provide GPRS access via a network driver interface. To tell NowSMS to use that specific network driver for connecting to the MMSC, select the named **“Network:”** driver.

The **“WAP Gateway IP Address”** field should contain the IP address of the operator WAP gateway which will act as a proxy for connections to the MMSC.

For most GSM/GPRS modems, it is not possible for the Now SMS/MMS Gateway to connect to the modem to send/receive SMS messages at the same time as a GPRS connection is active (Network Connection type = "Modem:" or "Dial-up:"). For these situations, the "**Modem Used**" setting should specify the name of a modem that is used by the Dial-up networking profile. If the Now SMS/MMS Gateway is using the modem to send/receive SMS messages, it will automatically release the modem when it needs to initiate a GPRS connection, otherwise the GPRS connection will not be able to be properly established.

The "**GPRS APN**" field specifies the GPRS Access Point Name (APN) to be accessed for connecting to the MMSC. This setting is operator dependent, and it may be advisable to check the MMS configuration settings on a working mobile phone to determine the correct settings. Note that your mobile operator possibly has multiple GPRS APNs and multiple WAP gateways, and you need the settings that are appropriate for MMS, not for WAP browsing or general internet connectivity. Note that this setting is only available when using a Network Connection of type "Modem:". For other connection types, the GPRS APN must be configured for the connection using some configuration means external to NowSMS.

The "**Login Name**" and "**Password**" parameters specify a username and password to be used for connecting to the GPRS network.

The "**Test Connection**" dialog verifies that the Now SMS/MMS Gateway can make a TCP/IP connection to the specified "Server Address". Or, when an MM1 connection is used with a dial-up connection ("Use Specific Network Connection (GPRS Modem)"), the gateway will initiate a network connection to the specified profile, and that it can make a connection to the specified WAP gateway over the connection, and the "Server Address" is not tested at this time.

Connecting to an Operator MMSC - Receiving MMS Messages

This section describes configuration information specific to receiving MMS messages from an operator MMSC using MM7, MM4 or EAI. Receiving MMS messages over an MM1 connection using a GPRS modem is described in the section titled [Connecting to an Operator MMSC - Using a GPRS Modem](#).

In order to receive messages from an operator MMSC using MM7, MM4 or EAI, the mobile operator must configure their MMSC to initiate outbound connections to your MMSC for message delivery.

The process of configuring the Now SMS/MMS Gateway to send messages to an operator MMSC is separate from the process of configuring the gateway to receive messages from an operator MMSC.

While an operator MMSC is technically not a VASP (Value Added Service Provider), in order to receive MMS messages from an operator MMSC, you must configure a VASP account for the mobile operator on the "MMSC VASP" page of the configuration dialog. Please refer to the section titled [Configuring MMS VASP Accounts](#) for more information on these configuration settings.

When an operator MMSC connects to your system to deliver received MMS messages, the process is the same as a VASP connecting to your system to submit messages for external delivery. The primary difference is that the **"MMSC Routing for Received Messages"** setting is different for a VASP compared to an operator MMSC connection.

VASP Account

Account Name:

(Note: Specify IP address as name if account does not require a login.)

Password:

Account Description:

IP Address Restrictions:

Accept Connections via: MM7 MM4 (SMTP)
 MM1 (EAIF)

VASP Sender Address:

Allow Sender Address Override

MMSC Routing for Received Messages:

Standard MMS Delivery

Receive to MMS-IN Directory

Route via MM7

Forward to E-Mail Address

Enable Credit Balance
Credits to add:

Enable Message Sending Limits for this account
Max messages per day:
Max messages per month:

When defining a VASP account that is used to receive messages from a mobile operator, it is important that the “**MMSC Routing for Received Messages**” be set to **any value other than “Standard MMS Delivery”**. This is because the “Standard MMS Delivery” option would specify that the messages should be routed for external delivery. However, when receiving messages from an operator MMSC, you would want to perform receive processing against the messages.

URL Parameters for Sending Messages

To send an SMS message via a menu driven interface, please see the help section titled "[Web Menu Interface](#)". This section describes how to send a text message programmatically via URL parameters.

To send a message via SMS, use the following URL format:

<http://127.0.0.1:8800/?PhoneNumber=xxxxxxx&...>

For 127.0.0.1, please substitute the IP address or host name assigned to your gateway PC. (Note: 127.0.0.1 is a local loopback address that can be utilized when you are connecting to the gateway from the same computer.)

For 8800, please substitute the port number that the gateway is configured to use.

Substitute the phone number that you wish to send the SMS message to for the "xxxxxxx" in the "PhoneNumber" parameter. Use either the local phone number format, or the international phone number format (your network provider may or may not allow you to send to international phone numbers). If the international phone number format is used, note that you must substitute "%2B" for the "+" character, because of URL escaping restrictions. For example, to send an SMS to +447778001210, use the following URL format:

<http://127.0.0.1:8800/?PhoneNumber=%2B447778001210&...>

Parameters after the "..." are dependent on the type of message being sent. The following table summarizes available URL parameters:

URL Parameter	Message Type	Description
PhoneNumber	All	Recipient phone number
Text	Text SMS, Binary SMS, WAP Push, WAP OTA Bookmark	For SMS text messages, this specifies the text of the message. For binary SMS messages, this is a string of hexadecimal characters representing the data being sent in the binary message. For WAP Push messages, this is the text associated with a Service Indication (SI) Push. For a WAP OTA Bookmark, this is the text

		name of the bookmark.
Data	Same as "Text" parameter	Interchangeable with the "Text" parameter. Officially documented only for binary SMS messages.
UDH	Binary SMS	A text string of hexadecimal characters representing the User Data Header (UDH) of the binary SMS message.
DCS	Binary SMS	A hex value representing the value of the SMS Data Coding Scheme (DCS) for this message. F7 is a common value for most binary SMS message types.
PID	Binary SMS	A hex value representing the SMS Protocol ID (PID) of this SMS message.
Binary	Binary SMS	Set to 1 for binary message submission
Sender	All	Sender phone number for this SMS message.
ContinueURL	All	URL to continue to after SMS message submission.
VoiceMail	Voice Mail Notification	Set to "On" to turn on voice mail waiting indication, "Off" to turn off voice mail waiting indication.
WAPURL	WAP Push	URL to be referenced in the WAP Push message.
WAPSL	WAP Push	Set to any value to send the WAP Push as a "Service Load" (SL) message, instead of the default "Service Indication" (SI) message.
MMSText	MMS Message	Text to be included when sending an MMS Message.
MMSFile	MMS Message	Contains the contents of an uploaded file posted via a form with a MIME encoding of "multipart/form-data", or specifies a HTTP URL pointing to the file content when specified in a GET request.
MMSSubject	MMS Message,	Subject for the MMS Message or MMS Notification Message.

	MMS Notification	
MMSFrom	MMS Message, MMS Notification	Sender for the MMS Message or MMS Notification Message
MMSURL	MMS Notification	URL that contains the MMS Message content for which an MMS Notification Message should be generated.
WAPBookmark	WAP Bookmark	URL to be sent as a bookmark.
OTA	WAP OTA Config	Name of an ".ota" file in the OTA subdirectory which contains WAP OTA configuration information, or value "POST" when OTA content is being submitted via HTTP POST.
OTAOMA	OMA Provisioning Content OTA	Name of an ".ota" file in the OTA subdirectory which contains an OMA Provisioning Content document, or value "POST" when provisioning content is being submitted via HTTP POST.
OTAPIN	OMA Provisioning Content OTA	PIN associated with the OMA Provisioning Content Request. (PIN is the SIM card IMSI when OTAPINTYPE=NETWPIN)
OTAPINTYPE	OMA Provisioning Content OTA	Specifies the type of PIN specified in the OTAPIN variable. Can either be a value of USERPIN or NETWPIN.
OTABookmark	WAP OTA Config	Name of a ".bm" file in the OTA subdirectory which contains WAP OTA bookmark configuration information.
OTA_BEARER	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=BEARER
OTA_PPP_AUTHTYPE	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=PPP_AUTHTYPE
OTA_PPP_AUTHNAME	WAP OTA Config	Maps to TYPE=ADDRESS PARM NAME=PPP_AUTHNAME
OTA_PPP_AUTHSECRET	WAP OTA Config	Maps to TYPE=ADDRESS PARM NAME=PPP_AUTHSECRET
OTA_PPP_LOGINTYPE	WAP OTA	Maps to TYPE=ADDRESS, PARM

	Config	NAME=PPP_LOGINTYPE
OTA_PROXY	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=PROXY
OTA_PROXY_TYPE	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=PROXY_TYPE
OTA_PROXY_AUTHNAME	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=PROXY_AUTHNAME
OTA_PROXY_AUTHSECRET	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=PROXY_AUTHSECRET
OTA_PROXY_LOGINTYPE	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=PROXY_LOGINTYPE
OTA_PORT	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=PORT
OTA_CSD_DIALSTRING	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=CSD_DIALSTRING
OTA_CSD_CALLTYPE	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=CSD_CALLTYPE
OTA_CSD_CALLSPEED	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=CSD_CALLSPEED
OTA_ISP_NAME	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=ISP_NAME
OTA_SMS_SMSC_ADDRESS	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=SMS_SMSC_ADDRESS
OTA_USSD_SERVICE_TYPE	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=USSD_SERVICE_TYPE
OTA_GPRS_ACCESSPOINTNAME	WAP OTA Config	Maps to TYPE=ADDRESS, PARM NAME=GPRS_ACCESSPOINTNAME
OTA_URL	WAP OTA Config	Maps to TYPE=URL
OTA_MMSURL	WAP OTA Config	Maps to TYPE=MMSURL
OTA_NAME	WAP OTA Config	Maps to TYPE=NAME PARM NAME=NAME
OTA_BOOKMARK_NAME	WAP OTA Config	Maps to TYPE=BOOKMARK, PARM NAME=NAME
OTA_BOOKMARK_URL	WAP OTA Config	Maps to TYPE=BOOKMARK, PARM NAME=URL
OTA_ID	WAP OTA Config	Maps to TYPE=ID, PARM NAME=NAME

When a text message is being submitted via the "Text" parameter, note that due to URL escaping restrictions, space characters should be replaced with "+" characters. Also, certain special characters, such as "?", "&", ":" and "=" need to be replaced with an escape character. The gateway expects characters to be encoded in UTF-8 (Unicode-based) format, therefore some

characters, including the Euro (€) may require multiple escaped characters. (Note: The Web Menu Interface automatically performs this escaping.) The following table shows common characters that must be escaped:

"	%22
<	%3C
>	%3E
&	%26
+	%2B
#	%23
%	%25
*	%2A
!	%21
,	%2C
'	%27
\	%5C
=	%3D
€	%E2%82%AC

Message text up to 160 characters in length can be sent in a single SMS message. The gateway automatically supports the sending of longer messages by utilizing "concatenated SMS" to send messages larger than 160 characters in length. Note that some older mobile phones will not support longer SMS messages. For longer SMS messages, one message is sent for every 153 characters of text in the message.

Technical Bulletins

This section contains technical bulletins that provide supplemental information to the standard Now SMS/MMS Gateway documentation.

Now SMS/MMS Accounting Callbacks

Document ID: TB-NOWSMS-001, Last Update: February 5, 2003

Accounting callbacks have been added to the Now SMS/MMS Gateway effective with v4.11 and later.

Before the gateway accepts an SMS or MMS message for delivery, the gateway will connect to a configurable customer-provided URL, providing information about the message to be sent, and who is sending the message. The customer provided URL can either tell the gateway to either accept or reject the message. This is a "pre-authorisation" request, and does not mean that the message will actually be accepted for delivery. If the gateway cannot successfully connect to the URL, or the URL returns a response other than a standard "HTTP 200 OK", the user request to send a message will be blocked. A "PreAuth" request to send a message will also be blocked if the response includes the text "PreAuth=Deny".

After an SMS or MMS message is accepted for delivery, the gateway will connect to a configurable customer provided URL, providing accounting information about the message being sent, allowing the customer to maintain external accounting information on messages processed by the Now SMS/MMS Gateway. The gateway ignores HTTP responses to the accounting callbacks.

MMS Accounting Callbacks

For MMS messages, the MMS callback URL is defined in the MMSC.INI file, under the [MMSC] section header:

```
MMSAccountingURL=http://server.name/path
```

The variables listed in the sections below will be added to the MMSAccountingURL when the URL is executed by the gateway as HTTP GET (CGI-style) parameters.

MMSSend PreAuth Callback

This callback is executed when an MMS user is requesting to send a message. The following variables will be set:

```
PreAuth=Yes (indicates that the message is a Pre-Authorisation Request)  
Type=MMSSend  
From=SenderPhoneNumber
```

MsgCount=#### (number of recipients user is requesting to send the message to)

MMSSend Accounting Callback

This callback is executed after an MMS message that has been sent by an MMS user has been accepted by the gateway for delivery.

Type=MMSSend
From=SenderPhoneNumber
To=RecipientPhoneNumberOrEmailAddress
MessageID=SystemSpecificMessageID
Size=#### (size of message in bytes, may vary slightly with header changes in-transit)

MMSEMail Pre-Auth Callback

This callback is executed when an e-mail message is being presented for delivery to an MMS recipient. The following variables will be set:

PreAuth=Yes
Type=MMSEMail
From=SenderPhoneNumberOrEmailAddress (almost always an e-mail address)
To=RecipientPhoneNumber
MsgCount=1 (always 1 in current implementations)

MMSEMail Accounting Callback

This callback is executed after an MMS message has been received via e-mail, and has been queued for delivery to an MMS recipient. The following variables will be set:

Type=MMSEMail
From=SenderPhoneNumberOrEmailAddress (almost always an e-mail address)
To=RecipientPhoneNumber
MessageID=SystemSpecificMessageID
Size=#### (size of message in bytes, may vary slightly with header changes in-transit)

MMSRetrieve Accounting Callback

This callback is executed with a client connects in to retrieve an MMS message. The following variables will be set:

Type=MMSRetrieve
From=SenderPhoneNumberOrEmailAddress
To=RecipientPhoneNumber
MessageID=SystemSpecificMessageID
Size=#### (size of message in bytes)

SMS Accounting Callbacks

For SMS messages, the SMS callback URL is defined in the SMSGW.INI file, under the [SMSGW] section header:

SMSAccountingURL=http://server.name/path

The variables listed in the sections below will be added to the SMSAccountingURL when the URL is executed by the gateway as HTTP GET (CGI-style) parameters.

SMSSend PreAuth Callback

This callback is executed when an SMS (web, SMPP, SMTP) user is requesting to send a message. The following variables will be set:

PreAuth=Yes (indicates that the message is a Pre-Authorisation Request)
Type=SMSSend
From=DefinedSMSUserAccount
MsgCount=#### (number of recipients user is requesting to send the message to)

SMSSend Accounting Callback

This callback is executed after an SMS message that has been sent by an SMS user has been accepted by the gateway for delivery.

Type=SMSSend
From=DefinedSMSUserAccount

To=RecipientPhoneNumber
MessageID=SystemSpecificMessageID
Size=##### (size of message in bytes, may vary slightly with header changes
in-transit)

Now MMSC Operator Considerations

Document ID: TB-NOWSMS-002, Last Update: May 5, 2003

Effective with v4.11 and later of the Now SMS/MMS Gateway, the operator functions of the Now MMSC are now included as base features of the MMSC built into the Now SMS/MMS Gateway.

The primary considerations for the Now MMSC in an operator environment pertain to user provisioning and user identification.

In the standard configuration for the MMSC, it is a requirement that each user be configured with a special MMS Message Server URL. The URL contains the username and password of the user account on the MMSC. In an operator environment, it is not practical to provision phones with unique MMS Message Server URLs, and it is also not practical to manually configure user accounts on the MMSC.

The reason that the standard configuration of the MMSC requires each user to be configured with a special MMS Message Server URL, is because the connection that the phone makes to the MMSC is IP-based, and contains no information about the MSISDN (phone number) of the user that is making the request. In fact, the MMSC only sees an HTTP request coming from the IP address of the WAP gateway that is proxying the request on behalf of the mobile device. Without further hooks into the operator network to determine the identity of the device making the request, the MMSC relies on the username and password in the URL request to identify the user.

To overcome this limitation, it is possible to configure the MMSC to receive user identity information directly from the operator network, without requiring the username and password on the URL.

To integrate into the operator network, the Now MMSC expects to receive user information from the operator WAP gateway. As all requests to the MMSC are being proxied through the operator WAP gateway, it is best that the WAP gateway take responsibility for user identification. Additional information on configuring the WAP3GX gateway to provide this information is provided later in this document.

The WAP gateway must be configured to forward the MSISDN of the requesting user to the MMSC via a configurable HTTP header. For example, the WAP3GX gateway uses the "X-MSISDN:" header to forward the MSISDN to HTTP content servers such as the MMSC.

To configure the Now MMSC to read the MSISDN from an HTTP header, configuration settings must be applied to the MMSC.INI file. The following configuration settings may be applied within the [MMSC] section of the MMSC.INI file:

`MSISDNHeader=header-name`

The `MSISDNHeader` setting specifies the name of the HTTP header that will contain the MSISDN. For example, with WAP3GX, the appropriate setting would be:

`MSISDNHeader=X-MSISDN`

`MSISDNHeaderGateways=1.2.3.4,5.6.7.8,9.10.11.*`

The `MSISDNHeaderGateways` setting specifies a list of one or more IP addresses from which the MMSC will accept the `MSISDNHeader`. This is to prevent forged requests, where another gateway or application inserts an MSISDN header to attempt to fool the MMSC. One or more IP addresses can be listed in this configuration setting. Each address must be separated by a comma. Wildcard addresses are supported by placing a "*" in place of a position within the IP address. For example, a setting of 9.10.11.* would mean that the `MSISDNHeader` would be accepted from any request originating from an IP address in the range of 9.10.11.1 thru 9.10.11.255.

`MSISDNHeaderDefaultCountryCode=##`

The `MSISDNHeaderDefaultCountryCode` setting specifies a default country code to be applied to MSISDN numbers presented in the `MSISDNHeader`, so that the gateway can convert the MSISDN address to international format automatically.

`MSISDNHeaderLocalPrefix=#`

The `MSISDNHeaderLocalPrefix` setting specifies the default prefix that is used for phone numbers in the MSISDN header that are in local (national) format. For example, in the UK, the `MSISDNHeaderDefaultCountryCode` would be 44, and the `MSISDNHeaderLocalPrefix` would be 0. With these settings, and MSISDN header of "07778001210" would automatically be converted to "+447778001210" by the MMSC.

`MSISDNHeaderAutoProvision=Yes/No`

The `MSISDNHeaderAutoProvision` setting specifies whether or not user accounts should be automatically provisioned on the MMSC the first time

that a user sends a message through the MMSC. This removes the requirement to automatically provision accounts. Any user that makes a request through the appropriate WAP gateway with the MSISDNHeader set will be automatically provisioned on the MMSC.

Configuring the WAP3GX WAP Gateway to Forward MSISDN

When a mobile device connects to a WAP Gateway, the connection requests are made over the IP protocol, and there is no part of the WAP standard that specifies how the MSISDN is presented to the WAP gateway.

The mobile device is generally assigned an IP address when it connects to the GGSN (or network access server for dial-up connections, collectively we'll refer to them as this point on as an "access server"). This IP address is usually dynamically assigned and changes between sessions.

The WAP gateway needs to receive information from the "access server" in order to identify the MSISDN of a device associated with the IP address making a request of the WAP gateway.

The WAP3GX gateway integrates with access servers using the Radius accounting protocol, which is one of the Radius (Remote Authentication Dial In User Service) protocols. The Radius authentication protocol is defined in internet RFC 2865, and the Radius accounting protocol is defined in internet RFC 2866. The two protocols are frequently used together.

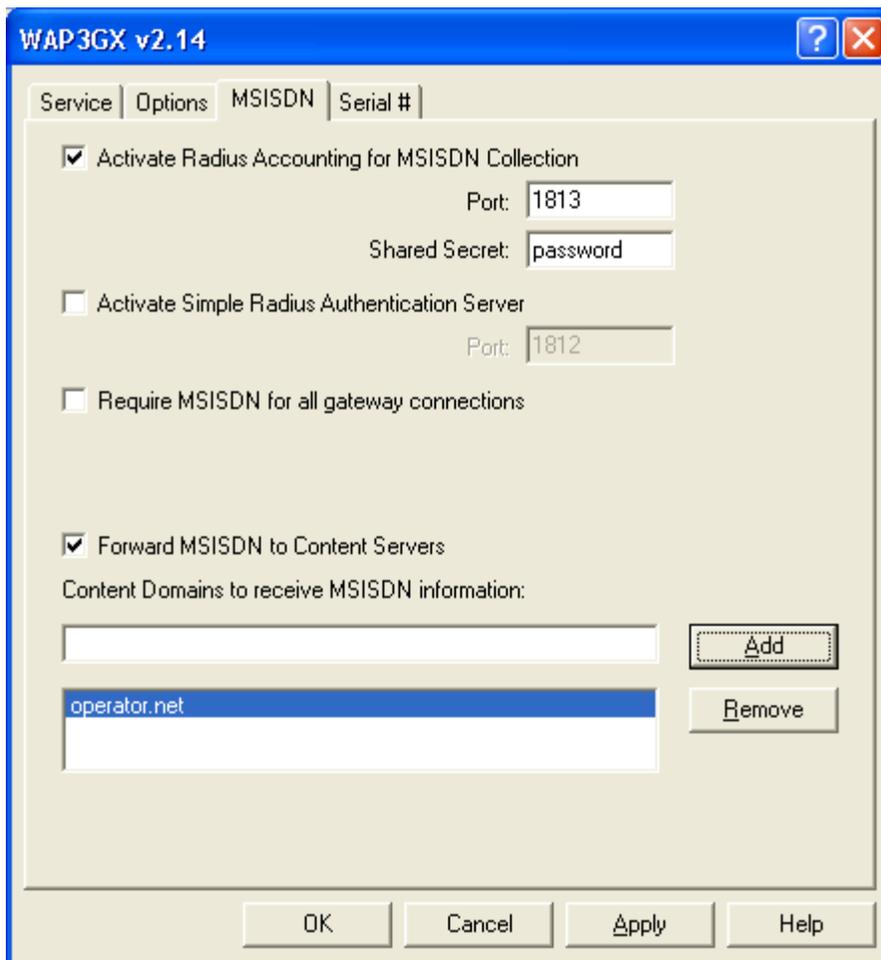
The Radius authentication protocol is used to authenticate a user. When a new connection is received by an access server (including a GGSN), the access server can be configured to use Radius authentication to determine whether or not to accept the connection. The access server sends a Radius authentication request over UDP to a Radius server. This request includes the username presented and a hash of the password against a shared secret, and usually the CLI (caller line identification) of the station making the request. The Radius server responds back over UDP to the access server telling the access server whether or not to allow the connection, and in some configurations, the Radius server can also tell the access server what IP address to assign to the client.

After a connection has been authenticated, the access server can be configured to use the Radius accounting protocol to inform an accounting server of a new connection. Similarly, the access server notifies of a dropped connection. The Radius accounting protocol is also UDP-based, but it is an informational protocol. After a new connection is authenticated, the access server sends a Radius accounting packet over UDP to the configured accounting server(s). This packet includes a hash of a shared secret for packet validation, and

typically includes the username that connected, the IP address that was assigned, and the CLI of the station that connected.

In an operator environment, users are typically not assigned their own username for the purposes of Radius authentication. Instead CLI is the determining factor.

To provide MSISDN support, WAP3GX needs to be configured as a Radius accounting server, so that it receives Radius accounting packets from the access server. This support is activated in WAP3GX on the MSISDN page of the configuration dialog.



In the configuration dialog, you must specify a port (1813 is the default according to the specification, but some access servers default to 1646 which is an incorrect default from an older specification), and the appropriate shared secret.

This setting enables the WAP3GX gateway to receive Radius accounting packets from the access server.

Check "Forward MSISDN to Content Servers", and add your local domain name as a content domain. The gateway will then include the MSISDN in any requests to content servers within any listed domains. For example, if "operator.net" is listed as a supported content domain, then the MSISDN would be forwarded to a content server named "mms.operator.net", or a content server simply named "operator.net". The gateway will forward the MSISDN to the content server in all HTTP requests sent to the content server, using the HTTP "X-MSISDN:" header.

As long as the Radius accounting feed from the access server is reliable, it may be desirable to require an MSISDN for any connections to the WAP gateway. This can be enabled by checking "Require MSISDN for all gateway connections". Connections from IP addresses where the WAP gateway has not received an MSISDN assignment from the access server will be rejected.

Getting the Radius accounting feed setup in the access server correctly, so that it is sent to the WAP gateway is the hardest part of this process. Unfortunately, WAP3GX doesn't give you much indication as to whether or not it is receiving a Radius accounting feed. When testing, it is best to enable the debug log in WAP3GX (edit WAPGW.INI, add Debug=Yes under [WAPGW] section header), and monitor the WAPDEBUG.LOG for a keyword of "Radius", where it will log all Radius packets that the gateway receives. If you have problems getting the Radius configuration working, then it is best to contact WAP3GX technical support for further advice.

WAP3GX can also be configured to be a Radius authentication server, but it is a VERY simple Radius authentication server. When WAP3GX is a Radius authentication server, it will accept any authentication request. We do not recommend the use of this setting, except for testing purposes. (This setting was added originally for a particular customer that needed it.)

2-Way SMS Returning a Non-Text Response

Document ID: TB-NOWSMS-003, Last Update: February 7, 2003

It is a relatively simple process to define a 2-way command that returns a text response to the message sender. The "2-Way" page of the NowSMS configuration dialog allows you to define commands that are executed when an SMS message is received. The receipt of an inbound SMS message by the gateway can cause an executable command to be executed, or the gateway can connect to an HTTP URL to run a script on a separate web server. If "Command returns response text" is set for the command, it is simple for the command to return a text response back to the sender of the SMS. In the case of an HTTP URL, the HTTP URL can return a simple text response with a MIME type for the response set as "text/plain", and the gateway will automatically send this text as a response back to the sender of the SMS.

The screenshot shows the "Now SMS/MMS Gateway v4.11 BETA" configuration window. The "2-Way" tab is selected, and the "Process Received SMS Messages" checkbox is checked. The "Received SMS Command Table" contains one entry with a prefix "*" and a command to execute: "http://192.168.1.102/cgi-bin/sms.cgi?Sender=@". The "Command returns response text" checkbox is also checked. The "Process received MMS Messages" checkbox is unchecked.

MMSC	MMSC Users	Serial #		
Service	SMSC	Web	SMS Users	2-Way

Process Received SMS Messages

Received SMS Command Table

*	http://192.168.1.102/cgi-bin/sms.cgi?Sender=@
---	---

SMS Command Prefix: *

Command to Execute: http://192.168.1.102/cgi-bin/sms.cgi?S

Command returns response text

Process received MMS Messages

MMS Directory:

In the example shown in the screen shot above, the receipt of a text SMS message causes the gateway to connect to the HTTP URL:

```
http://192.168.1.102/cgi-  
bin/sms.cgi?Sender=@@Sender@@&Text=@@FullSMS@@
```

When a message is processed, the “@@Sender@@” value is replaced with the phone number of the sender of the received message, and @@FullSMS@@ contains the complete text of the received message.

What if an HTTP script wants to return a response back to the sender that is a non-text SMS format, such as WAP OTA settings, a Nokia smart messaging message, or even an MMS message?

The HTTP script could always include its own processing logic to spawn a command that issues a request back to the NowSMS gateway, but this can be difficult. An alternative approach is for the HTTP response to return a “redirect” response, where the redirect URL is a URL pointer back to the NowSMS gateway, for example
`http://127.0.0.1:8800/?PhoneNumber=%2b12125551212&...` (If you want to be able to test your HTTP script by executing it from a web browser, you will need to substitute the real IP address of your NowSMS server for the 127.0.0.1 address.)

If your NowSMS server requires authentication, a username and password can be included in the redirect response using the URL format
`http://user:password@server.name/path.`

Any URL that would work when sent directly to the NowSMS server can be returned as a “redirect” response. If a script will be returning a “redirect” response, it is important that “Command returns response text” is NOT checked for the command on the “2-Way” configuration dialog. This is because a redirect response simply instructs the HTTP client to retrieve a different URL. The NowSMS gateway will not return the actual response from that URL request to the client.

An HTTP redirect response is an HTTP 302 response, with the redirect URL included in the “Location:” header, for example:

```
HTTP/1.0 302 Redirect  
Location: http://127.0.0.1:8800/?PhoneNumber=...  
<blank line>
```

Most scripting languages have built-in support for returning a redirect response. For example, in an ASP script, `Response.Redirect`

("http://127.0.0.1:8800/?PhoneNumber=...") is used to send a redirect response. Other scripting languages have similar support.

Using Port 80 on a PC running IIS

Document ID: TB-NOWSMS-004, Last Update: February 7, 2003

As mentioned on the NowSMS documentation, the MMSC works best with operator WAP gateways when it is configured to use HTTP port 80. HTTP Port 80 is the standard port used by web servers. If an IIS web server is running on the same PC as the NowSMS gateway, it is likely that port 80 is already in use by the IIS web server.

If you can bind multiple IP addresses to the PC running IIS and the NowSMS gateway, it is possible for both IIS and NowSMS to use port 80, by assigning one IP address to IIS, and another IIS to the NowSMS gateway.

To assign a specific IP address to the NowSMS gateway, it is a simple matter of selecting the appropriate IP Address from the "IP Address" list on the MMSC page of the configuration dialog. This setting defaults to "(all available)", meaning that the NowSMS gateway would bind to the specified ports on all IP addresses bound to the PC. Note that when you configure NowSMS to use a specific IP address, it will no longer be accessible on the local machine using the loopback address 127.0.0.1, you must always use the specific IP address, even when connecting from the local machine. Also note that the "IP Address" setting on the MMSC page of the configuration dialog applies only for the IP address used by the HTTP and SMTP ports of the MMSC. The Now SMS gateway itself also has an HTTP web port. The IP address setting for that interface is found on the "Web" page of the configuration dialog.

It is not as simple to configure IIS to bind to only a specific IP address. IIS has a feature called "socket pooling", which causes it to grab the web ports on all IP addresses configured on the machine, even if web services have only been configured for a specific IP address.

To configure IIS to bind only to a specific IP address or a limited number of specific IP addresses, check the configuration of each host defined in the Internet Information Services Manager, and ensure that the host is configured for a specific address instead of all unassigned addresses. You must then disable IIS socket pooling.

For information on disabling socket pooling on IIS, and configuring IIS to connect to only a specific IP address, please consult the following Microsoft references:

IIS Binds To All Available IP Addresses When It Starts

<http://support.microsoft.com/support/kb/articles/Q259/3/49.ASP>

How to Disable Socket Pooling

<http://support.microsoft.com/support/kb/articles/Q238/1/31.ASP>

XCON: How to Disable Internet Information Services Socket Pooling on the SMTP Service

<http://support.microsoft.com/support/kb/articles/Q281/7/60.ASP>

Provisioning MMSC User Accounts via HTTP

Document ID: TB-NOWSMS-005, Last Update: February 7, 2003

With the v4.11 release of the Now SMS/MMS Gateway, an additional HTTP interface has been added for provisioning MMSC User Accounts, to compliment the existing menu driven "MMSC Admin" interface.

The interface is HTTP "GET" based, and requires that the HTTP request include an Authorization header. The Authorization header is set with the BASE64 encoded value of "username:password", where username and password are the values configured for the MMSC Admin interface.

HTTP

```
/provision?AdminAction=Action&PhoneNumber=number&Name=AliasName&Password=password&FullName=Full+Name HTTP/1.0  
Authorization: username:password (BASE64 encoded)  
<blank line>
```

Valid values for "AdminAction" are "Add", "Modify" and "Delete".

All actions require that a "PhoneNumber" be specified. (If you are configuring phone numbers in international format, be sure to remember that HTTP GET URL parameters must be URL encoded, so that "%2B" is used to represent the "+" character.) The other parameters (Name, Password and FullName) are optional.

Assuming that the user is authorized to issue a provisioning command (valid username/password, and supported IP address for originating the request), the server will return a normal HTTP response code of 200 with a MIME content type of "text/plain". The response will start with "OK" if the request was successful, or "ERROR" if the request failed.

Support for restricting access to the MMSC Admin and provisioning web functions by IP address was added in v4.11 of the Now SMS/MMS Gateway. Under the [MMSC] section header, use the AdminIPAddressList setting, which is a comma delimited list of IP addresses that are allowed access to the gateway. Do not include any spaces between addresses. To specify a wildcard address for a subnet, use the "*" character (for example, 192.168.1.* to allow all addresses in the 192.168.1 subnet).

Faster GSM Modem Speeds with SMS over GPRS

Document ID: TB-NOWSMS-006, Last Update: February 10, 2003

With the v4.11 release of the Now SMS/MMS Gateway, the GSM modem interface has been optimised to support faster message transmission, particularly when used on devices and networks that support SMS over GPRS. With traditional SMS over GSM links, it was only possible to achieve an SMS transmission speed of 6 to 10 messages per minute, depending on network factors. However, with SMS over GPRS, and the v4.11 release of the Now SMS/MMS Gateway, we've tested GSM modem speeds of 30 messages per minute, or 1 message every 2 seconds, a speed that is 3 to 5 times faster than previously available with GSM modems.

While SMS over GPRS is still not nearly as fast as SMS speeds available with direct SMSC connections, SMS over GPRS does make the use of GSM/GPRS modems fast enough for a larger variety of applications.

To support SMS over GPRS, you require a mobile phone that can be configured to support SMS over GPRS, and you will need a subscription with a mobile operator that supports SMS over GPRS. (Many mobile operators are in the process of adding support for SMS over GPRS, as it can be more cost effective for them to implement on their networks.)

Our tests of 30 messages per minute over a GSM/GPRS modem were conducted with a SonyEricsson T68i connected to the NowSMS gateway with a serial cable. The mobile operator was T-Mobile in the USA. We configured the T68i to support SMS over GPRS by selecting Connect->Data. comm.->SMS access->GPRS.

It is anticipated that many more devices and modems will be introduced that support SMS over GPRS in the coming months. Below is a list of devices of which we are currently aware that support SMS over GPRS, with instructions for configuring the device to support SMS over GPRS. This should not be considered a comprehensive list of devices supporting SMS over GPRS. Consult the documentation for your device for additional information, and inquire with your network operator to determine if they support SMS over GPRS.

Ericsson T39/T65:

Settings->Data comm.->SMS access->GPRS

Ericsson T68m/SonyEricsson T68i:

Connect->Data comm.->SMS access->GPRS

Nokia 8310:

Menu->Services->Settings->Edit active service settings->GPRS Connection->Select "Always Online"

Menu->Messages->Message Settings->Default Profile->Use GPRS->Select "Yes"

Siemens S45/ME45:

Menu->Messages->Message Setup->SMS. Select Options. Select Change Settings. Select SMS via GPRS.

Routing MMS Notifications via a WAP Push Proxy Gateway

Document ID: TB-NOWSMS-007, Last Update: April 4, 2003

With releases v4.20 and later of the Now SMS/MMS Gateway, a configuration option has been added to allow MMS Notification messages to be routed via a WAP Push Proxy Gateway, instead of direct delivery over SMS. This configuration option can allow for improved integration of the MMSC with some existing WAP environments.

Note that the WAP Push Proxy Gateway (PPG) to which the MMSC connects must support the following characteristics:

1. IF AUTHENTICATION IS REQUIRED (USERNAME/PASSWORD), THE PPG MUST SUPPORT HTTP BASIC AUTHENTICATION
2. THE ABILITY TO RECEIVE PUSH DATA IN 8-BIT FORMAT, WITH SUPPORT FOR THE MIME TYPE “APPLICATION/VND.WAP.MMS-MESSAGE” (NOTE THAT WHEN THIS SETTING IS ENABLED, THE WAP PUSH OPTION SUPPORTED BY THE NOWSMS WEB MENU INTERFACE WILL ALSO GENERATE PUSH MESSAGES VIA THE CONFIGURED PPG, AND IN THESE CASES, THE MIME TYPES “APPLICATION/VND.WAP.SIC” AND “APPLICATION/VND.WAP.SLC” WILL BE USED).
3. A PUSH RECIPIENT ADDRESS FORMAT OF “WAPPUSH=PHONENUMBER/TYPE=PLMN@DOMAIN”, WHERE “PHONENUMBER” WILL CONTAIN THE PHONE NUMBER OF THE PUSH RECIPIENT, AND “DOMAIN” IS A CONFIGURABLE SETTING.

To configure the Now SMS/MMS Gateway to generate MMS notifications via a WAP Push Proxy Gateway, you must manually edit the MMSC.INI, and create a section of this file with the header “[WAPPPG]”. The following settings are supported under the “[WAPPPG]” section header:

URL=http://host.name:port/path

This setting is required. It should point to a URL that is valid for submitting messages to the PPG. The host name may be entered as either a DNS host name or IP address, and the port setting defaults to “80” if not otherwise specified.

User=username

This setting is optional. It specifies a username for HTTP Basic authentication to be used to connect to the PPG.

Password=password

This setting is optional. It specifies a password for HTTP Basic authentication to be used to connect to the PPG. (Note: This setting is ignored if the "User" setting is not specified.

PushAddressDomain=domain

This setting is optional. It specifies the "domain" portion of the WAP Push recipient address (e.g., "WAPPUSH=PhoneNumber/TYPE=PLMN@domain"). If this setting is left blank or not specified, the "@domain" portion will not be included in the WAP Push recipient address specification.