API Documentation

Sendega SMS/MMS Gateway

English version

sendega

CONTENTS

SENDEGA API DOCUMENTATION

API last updated April 11 2013

INTRODUCTION

Sendega AS and its product Sendega.com is a solution and technology for micro payments and communication using SMS, MMS and WAP messaging. Sendega AS acts as a transparent, white-label Content Acquirer and Transaction Router between Operators and Content Providers. Content Providers are provided with an easily implemented interface to connect to Sendega.com, which handles integration to the Operators.

This document provides a description of how to integrate the Content Provider's services for sending and receiving SMS, MMS and WAP messages through the Sendega.com system. The document is intended for technical architects and designers of the Content Provider's services.

OVERVIEW

Sendega API consists of a set of methods for sending and receiving SMS, MMS and WAP messages. All responses in HTTP are answered with XML or CSV for some functions, when requested. Username and password must always be used when using the methods.

All IP addresses for servers that will communicate via the API should be registered with us to prevent unauthorised persons sending SMS messages from your account. IP addresses can be registered on our customer pages at Sendega.com
All values are sent/received must be as described in this document. The date format is ISO-8601. YYYY-MM-DD HH:MM Return values comply with xml version 1.0 and use ISO-8859-1 encoding. All return xml are guaranteed "well formed", and follow the template settings in this API for each individual method.

Enquiries to Sendega.com system can either be SMPP, POST or GET. As GET has a limitation in the length of the address string, we recommend using POST for methods where a long message, or many telephone numbers, etc, are sent. All communication takes place via HTTP POST/GET. We recommend using SSL for all requests (https), for your own safety regarding username and password which is sent in every request.

IP AND SERVER SETTINGS

Outgoing traffic: https://smsc.sendega.com
Incoming traffic from Sendega (dlr, mo-messages, etc): 91.189.121.80 and 91,189,121.100
Newly added: 91.189.121.81, 91.189.121.104, 91.189.121.105 and 91.189.121.106

SEND CONTENT (SMS, WAP, MMS)

Parameter	Type	Default value	M/O*	Description
username	String		M	
password	String		M	
sender	String		M	Originating numeric or alpha numeric address for the outgoing message. Behavior may vary with Operator integrations. Maximum 16 numbers or 11 characters using alphanumeric sender.
destination	String		M	The MSISDN that the message should be sent to, starting with country code: Example 479232592. Multiple recipients are comma separated. *Maximum number of destinations per request: 100.*
pricegroup	Integer	0	O	Tariff class for sending premium messaging (premium rate). Consult Sendega for valid tariffs. See note below [1]
contentTypeID	Integer	1	O	Type of message to send. Default is 1 for bulk sms. Se own table with content type values description.
contentHeader	String	<NULL>	O	Message header. Must be hex-encoded. Only used when sending binary sms, wap or mms messages. See "valid parameter values" table for content header values and description.
content	String		M	The message content. The system automatically splits the message up into more messages if the text length exceeds 160 characters. Message will automatically be displayed as one single long message at the receiver, if the receiver's phone supports it. See "valid parameter values" table for more information
dlrUrl	String	<NULL>	O	URL used to receive delivery reports. Ignored when sending multiple recipients. See own chapter for delivery reports DO NOT SET THIS PARAMETER TO EXAMPLE VALUES! LEAVE EMPTY IF NOT USING DELIVERY REPORTS!
ageLimit	Integer	0	O	Used to indicate end-user age limit for premium or adult services.
extID	String	<NULL>	O	Your local unique ID reference. Will be returned if dlrUrl is used.
sendDate	String	<NULL>	O	The message can be delivered with delayed delivery time. Format: YYYY-MM-DD HH:MI:SS
refID	String	<NULL>	O	Only to be used when sending premium SMS/MMS to some countries.
priority	Integer	0	O	Low priority = -1 Normal priority = 0 High priority = 1 Se note [2]
gwID	Integer	0	O	Method for sending message to a specific gateway/supplier at Sendega. Contact Sendega for more information.
pid	Integer	0	O	Protocol ID of message. See note [3]
dcs	Integer	0	O	Data Coding Scheme. Always use 0 or 16 for flash sms. See miscellaneous chapter for more information about dcs values. See note [3]
output	String	xml	O	This parameter is only present in HTTP POST/GET API, not in SOAP API. Defines response format when making request to the API.

* M = Mandatory, O = Optional

[1] When sending premium messages, the Sender parameter must be the short code available for the preferred country. Ex Sender=2440 for Norway, or Sender=72721 for Sweden.

[2] High priority messages have a higher rate. (NOK 0.15 /€ 0.02 added to current unit price)

[3] Use of parameters PID and DCS at other values than 0 may affect the price of the sms, as this requires that the sms is sent through an advanced gateway. These parameters can cause sms to not be delivered, or to be handled in unexpected ways by the mobile handset. To read more about dcs-parameter, please consult 3GPP TS-23.028 (Technical Specification Group Core Network and Terminals; Alphabets and language-specific information)

Valid parameter values

Parameter	Values
contentTypeID	0 = WAP bookmark 1 = Sms Bulk 2 = Sms Premium / CPA 3 = MMS Bulk 4 = MMS Premium / CPA 5 = SMS Premium / GAS
contentHeader	contentTypeID 0 – URL contentTypeID 1, 2 and 5 – Binary message header contentTypeID 3 and 4 – Subject field Empty string for other content types
content	ContentTypeID 0 – Link text ContentTypeID 1, 2 and 5 – Message text or binary message body when sending binary ContentTypeID 3 and 4 – Base64 encoded .zip file containing MMS data (including .smil file and attachments)
priority	-1 = Low priority 0 = Normal priority 1 = High priority ***Note:** High priority messages have a higher rate. Messages to multiple recipients will always be set to priority -1. Please use priority -1 for large bulks of commercial messages*
sender	Senders must either be a number of up to 16 digits, or match the following regex for alphanumeric sender values: [A-Za-z0-9\.\-\ ]{1,11}. Although we will recommend matching alphanumeric sender values to this regex: [A-Za-z0-9]{1,11}, as "." (dot), "-" (hyphen) and " " (space) are discarded by quite a few newer phones, and some operators.
output	- xml - csv

RETURN VALUES

Both xml and csv is shown below. When using SOAP API, the response data will be an object with these listed parameters.

Note: This only indicates if message is successfully accepted by Sendega Mobile Communication Platform or not. For delivery and billing results, please see chapter “delivery reports”

Response parameters:

Parameter	Type	Description
Success	Boolean	Indicates if request was successful or not
MessageID	Guid	Guid/uniqueidentifier – uniquely identifies this request.
ErrorNumber	Integer	Error code according to Error code list – 0 if success.
ErrorMessage	String	Error description according to Error code list. Empty if success

RETURNVALUES WHEN MESSAGE IS SENT SUCCESSFULLY

```
<?xml version="1.0" encoding="iso-8859-1"?>
<SendResult xmlns="http://smsc.sendega.com/Content">
  <MessageID>7fa588e7-6e24-4b32-931e-01136ed02f8a</MessageID>
  <ErrorMessage />
  <ErrorNumber>0</ErrorNumber>
  <Success>True</Success>
</SendResult>
```

```
True;7fa588e7-6e24-4b32-931e-01136ed02f8a;0;
```

RETURNVALUE WHEN MESSAGE FAILED.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<SendResult xmlns="http://smsc.sendega.com/Content">
  <MessageID>00000000-0000-0000-0000-000000000000</MessageID>
  <ErrorMessage>Recipient not a valid MSISDN </ErrorMessage>
  <ErrorNumber>1012</ErrorNumber>
  <Success>False</Success>
</SendResult>
```

```
False;00000000-0000-0000-0000-000000000000;1012;Recipient is not a valid MSISDN
```

EXAMPLES

Web service URL: https://smsc.sendega.com/Content.asmx (Function: Send)
HTTP POST/GET URL: https://smsc.sendega.com/SendSms.ashx

When using SOAP API, all parameters must be submitted, even if empty. When using HTTP POST/GET API, optional parameters may be omitted.

EXAMPLE – SEND SMS

```
?username=xxx
&password=xxxx
&sender=Sendega.com
&destination=4790001100
&content=Hello world!
```

EXAMPLE – SEND BINARY SMS

```
?username=xxx
&password=xxxx
&sender=Sendega.com
&destination=4790001100
&contentTypeID=1
&contentHeader=0B05040B84C0020003100301
&content=6C497900436F6E746163747300746578742F782D76636172640043616C656E6 [..]
```

EXAMPLE – SEND WAP BOOKMARK

```
?username=xxx
&password=xxxx
&sender=Sendega.com
&destination=4790001100
&contentTypeID=0
&contentHeader=http://wap.mydomain.com
&content=My link to open wap-site!
```

EXAMPLE – SEND MMS

```
?username=xxx
&password=xxxx
&sender=26026
&destination=4790001100
&contentTypeID=3
&ContentHeader=MMS message subject
&Content= /9j/4AAQSkZJRgABAQEASABIAAD/7QAcUGhvd […]
```

ERROR CODES

The API returns the following error codes if the message is rejected. Always check API status codes when sending messages.

Error no	Error message	Description
1001	Not validated	Wrong CID
1003	Wrong format: pid/dcs	Wrong pid or dcs values.
1004	Erroneous typeid	Parameter contentTypeID incorrect.
1020	Fromalpha too long	Alphanumeric sender value to long. Max 11 chars
1021	Fromnumber too long	Numeric sender value to long. Max 16 numbers.
1022	Erroneous recipient, integer overflow	Integer overflow value used as recipient.
1023	No message content submitted	Content parameter has no value.
1024	Premium sms must have abbreviated number as sender	Please include short code such as 2440 for Norway, 72721 for Sweden
1025	The message sender is not allowed	The sender value is barred.
1026	Balance to low	Only for prepaid customers where the balance is to low to send messages.
1027	Message too long	Content value is too long. Bulk sms max 459 characters.
1028	Alphanumeric sender is not valid	Invalid alphanumeric value or barred.
1029	Unknown MSISDN. Remember to add country prefix	
1099	Internal error	
9001	Username and password does not match	Incorrect username or password.
9002	Account is closed	Contact Sendega for more information.
9004	Http not enabled	No access or API interface not available
9005	Smpp not enabled	No access or SMPP service not available.
9006	Ip not allowed	IP address not valid for this account. IP addresses can be managed using Sendega controlpanel
9007	Demo account empty	Only used for demo accounts with limited number of test messages

RECEIVE CONTENT (SMS, MMS)

To be able to receive messages you must order a keyword using Sendega control panel (http://controlpanel.sendega.com). By default, all messages received is only stored at Sendega.com. You can add several rules to a keyword, like automatic response and forwarding to external URL address using HTTP POST method as described here. Your server has to respond with a HTTP-200 Success. Sendega will then store the 50 first characters of the text in Your receive script. If you return any other HTTP status than 200 the message will be re-posted with increasing intervals for the next 24 hours.

Parameter	Type	Description
msgid	String	Sendega unique message id
msisdn	Double	The subscribers MSISDN starting with country code
msg	String	Message content
mms	Boolean	Set to 1 if the message is MMS
mmsdata	String	Contains Base64 encoded string of mms content as a zip file
shortcode	Integer	The short code the message was sent to.
mcc	Integer	Mobile country code
mnc	Integer	Mobile network code
pricegroup	Integer	Tariff used for MO content.
keyword	String	The keyword used
keywordid	Integer	Sendega keyword id.
firstname	String	For use with Number enquiry service
surname	String	For use with Number enquiry service
address1	String	For use with Number enquiry service
address2	String	For use with Number enquiry service
zip	String	For use with Number enquiry service
city	String	For use with Number enquiry service
errorcode	Integer	Used when receiving premium MO messages
errormessage	String	Used when receiving premium MO messages
registered	String	Date when Sendega received MO msg. Format: 2013-04-11T15:55:18

DELIVERY REPORTS

All messages are delivered with status codes from the various telecom operators. Delivery reports are posted automatically to an external URL given in “dlrUlr” parameter when delivery report is received for a message. Parameters are sent as POST data. All parameter names are lower case, except extID. Your server has to respond with HTTP-200 Success

Parameter	Type	Description
msgid	String	Unique message id provided by Sendega while sending message
extID	String	Transparent ID reference included when sending message
msisdn	Double	The subscribers MSISDN starting with country code
errorcode	Integer	Short error code from operator – ref. table Error codes.
errormessage	String	Long error description if error.
status	Integer	4 = Delivered, 5 = Failed
statustext	String	Either “delivered” or “failed”
operatorerrorcode	String	Actual error code received from operator/network
registered	String	Date and time when message delivered to Sendega
sent	String	Date and time when delivered to operator/ network
delivered	String	Date and time when message was delivered to handset

ERROR CODES

Error code	Error description
100	Successfully executed
101	Invalid MSISDN
102	Invalid fromAlpha
103	Invalid fromNumber
104	Invalid deliverytime
105	Invalid pricegroup or pricegroup not supported
106	Unknown subscriber – **Remove from database**
107	Absent subscriber – **Remove from database**
108	Subscriber busy for MT-SMS or SIM card full
109	Invalid reference ID
110	Invalid content or content length. (Message to long or illegal characters in message)
111	Subscriber barred for CPA content - **Remove from database**
112	Subscriber is too young – **Remove from database**
113	Subscriber reached monthly turnover limit – This end-user should **not** be attempted to be re-billed within this month.
114	Subscriber is temporarily barred
115	Subscriber is permanently barred
116	Subscriber account balance to low
117	Invalid MSISDN or operator/network not covered
118	Operator returned internal error – contact Sendega for more information
119	Network timeout
120	Communication error with operator
121	Operator not supported or not reachable
122	Message queued at operator/network SMSC
123	Billing failed. Content delivered to subscriber
124	Billing successful. Content not delivered to subscriber
125	Subscriber not registered for this service at operator.
126	Message acked by carrier – awaiting new status
127	Message not delivered. Carrier returned unknown errorcode/errormessage
128	Rejected by operator
129	Message filtered by Sendega.
130	Subscriber is already registered for this service at operator
197	Prepaid account at Sendega does not have sufficient founds
198	Other/unknown operator error
199	Other/unknown Sendega error

SUBSCRIBER INFORMATION ENQUIRY

This service is only available in a selected number of countries. It returns the full name and address of the owner of an MSISDN Please contact Sendega support for country availability.

URL to use: https://smsc.sendega.com/ExtraServices/NumberEnquiry.asmx/GetSubscriberInformation

Parameter	Type	Default value	M/O*	Description
username	String		M	
password	String		M	
msisdn	Double		M	The subscribers MSISDN starting with country code
outputLanguage	String		M	Not in use

RETURN VALUES

```
<?xml version="1.0" encoding="utf-8"?>
<SubscriberInformation>
  <ErrorCode>0</ErrorCode>
  <Success>true</Success>
  <Companies/>
  <Persons>
    <SubscriberInformationPerson>
      <IsDMReserved>true</IsDMReserved>
      <LifePhaseExtended>25-49 years, single, without children</LifePhaseExtended>
      <LifePhase>Single</LifePhase>
      <Addresses>
        <Address>
          <Longitude>10.797232965695033</Longitude>
          <Latitude>59.913838751987996</Latitude>
          <City>OSLO</City>
          <Zip>0661</Zip>
          <Address1>Karoline Kristiansens vei 2</Address1>
        </Address>
      </Addresses>
      <Birthdate>1970-01-01T00:00:00</Birthdate>
      <Gender>M</Gender>
      <LastName>LastName</LastName>
      <FirstName>FirstName</FirstName>
    </SubscriberInformationPerson>
  </Persons>
</SubscriberInformation>
```

For companies, only Company name and organization number will be available, in addition to the address.

HLR (HOME LOCATION REGISTRY ENQUIRY)

This service is available for all mobile numbers, and will return information about MCC/MNC for a given number. Sendega HLR service is directly connected with the number portability databases in Norway, Sweden and Denmark. For other destinations several other HLR services are implemented for best results.

Please see page 28 for more information about MCC and MNC codes.

URL to use: https://smsc.sendega.com/ExtraServices/NumberEnquiry.asmx/HomeLocationRegistry

Parameter	Type	Default value	M/O*	Description
username	String		M	
password	String		M	
msisdn	Double		M	The subscribers MSISDN starting with country code

RETURN VALUES

```
<NumberEnquiryResult>
  <MsisdnInformation>
    <HasOwnNetwork>true</HasOwnNetwork>
    <OperatorName>NetCom AS</OperatorName>
    <OperatorID>815</OperatorID>
    <Mnc>2</Mnc>
    <Mcc>242</Mcc>
    <Msisdn>4798402025</Msisdn>
  </MsisdnInformation>
  <ErrorCode>1000</ErrorCode>
  <Success>true</Success>
</NumberEnquiryResult>
```

SUBSCRIPTION SERVICE CHECK

Setting up subscription service check is mandatory if you are running subscription services and is used by Sendega customer service.

Parameters posted to script by Sendega (by HTTP POST):

Parameter	Type	Description
password	String	Given in Subscription Service Registration Form (Contact customer service)
msisdn	Double	The subscribers MSISDN including country code

The return type and value of this script should be well formed XML, with the following parameters:

Parameter	Type	Accepted values	Description
registered	Integer	1 / 0	If number is or has ever been registered
active	Integer	1 / 0	If number is currently active
startdate	DateTime	Valid time stamp	When number was registered at service
enddate	DateTime	Valid time stamp	If registered and closed, when service was stopped
stopurl	String	Valid url	An url to a script to close subscription for this MSISDN e.g.: http://myserver.example/cancel.php?id=1337
stopmsg	String		The stop keyword to cancel service e.g. MYSERVICE STOP
shortcode	Long		The shortcode the service is hosted at. e.g. 2440

If «registered» is set to 0, all other parameters are omitted, but should be set empty (not removed from response)

The field stopurl MUST be a valid url, DO NOT use the example url!

EXAMPLE RETURN DATA:

```
<?xml version=”1.0” encoding=”utf-8”?>
<customer>
  <registered>1</registered>
  <active>1</active>
  <startdate>2011-05-05T12:21:00</startdate>
  <enddate />
  <stopurl>http://myserver.example/cancel.php?id=1337</stopurl>
  <stopmsg>MYSERVICE STOP</stopmsg>
  <shortcode>2440</shortcode>
</customer>
```

CODE EXAMPLES

.NET C#

SEND SMS

Add the Content service by following these steps:

- Right-click at your project (not your solution) in server explorer
- Choose "Add service reference..."
- In the field "url" type *https://smsc.sendega.com/Content.asmx?wsdl*, and then press "go"
- Web reference name: "com.sendega.smsc"
- Press "add reference"

You are now ready to use the web service from your code as following in C#:

```
string username = "myUsername";
string password = "myPassword";
string sender = "Sendega";
string destination = "4790222333";
int pricegroup = 0;
int contentTypeID = 1;
string contentHeader = "";
string content = "Hello world!";
string dlrUrl = "http://myserver.example/mydlrUrl.aspx";
int ageLImit = 0;
string extID = "";
string sendDate = "";
string refID = "";
int priority = 0;
int gwID = 0;
int pid = 0;
int dcs = 0;

com.sendega.smsc.SendResult serverResult;
com.sendega.smsc.Content content = new com.sendega.smsc.Content();

serverResult = content.Send(username, password, sender, destination, pricegroup,
contentTypeID, contentHeader, content, dlrUrl, ageLimit, extID, sendDate, refID, priority,
gwID, pid, dcs);

if( serverResult.Success )
    Response.Write(string.Format("Message was sent. Id: {0}", serverResult.MessageID));
else
    Response.Write(string.Format("Message was not sent. Errornumber: {0}, Errormessage: {1}",
                                serverResult.ErrorNumber,
                                serverResult.ErrorMessage));
```

Do NOT use http://myserver.example/myDlrUrl.aspx as dlrUrl parameter.
THIS IS ONLY AN EXAMPLE VALUE! Use a valid URL, or set empty if not using delivery reports!

.NET VB

SEND SMS

Add the Content service by following these steps:

- Right-click at your project (not your solution) in server explorer
- Choose "Add service reference..."
- In the field "url" type *https://smsc.sendega.com/Content.asmx?wsdl*, and then press "go"
- Web reference name: "com.sendega.smsc"
- Press "Add reference"

You are now ready to use the web service from your code as following in VB:

```
Dim username As String = "myUsername"
Dim password As String = "myPassword"
Dim sender As String = "Sendega"
Dim destination As String = "4790222333"
Dim pricegroup As Integer = 0
Dim contentTypeID As Integer = 1
Dim contentHeader As String = ""
Dim content As String = "Hello World!"
Dim dlrUrl As String = "http://myserver.example/myDlrUrl.aspx"
Dim ageLimit As Integer = 0
Dim extID As String = ""
Dim sendDate As String = ""
Dim refID As String = ""
Dim priority As Integer = 0
Dim gwID As Integer = 0
Dim pid As Integer = 0
Dim dcs As Integer = 0
Dim serverResult As com.Sendega.smsc.Content.SendResult

Dim content As New com.sendega.smsc.ContentSoapClient("ContentSoap")

serverResult = content.Send(username, password, sender, destination, pricegroup, _
contentTypeID, contentHeader, content, dlrUrl, ageLimit, extID, sendDate, refID, priority, _
gwID, pid, dcs)

If serverResult.Success Then
    Response.Write(string.Format("Message was sent. Id: {0}", serverResult.MessageID))
Else
    Response Write("Message was not sent. Errornumber: {0}, Errormessage: {1}", _
                    serverResult.ErrorNumber, _
                    serverResult.ErrorMessage)
End If
```

Do NOT use http://myserver.example/myDlrUrl.aspx as dlrUrl parameter.
THIS IS ONLY AN EXAMPLE VALUE! Use a valid URL, or set empty if not using delivery reports!

CLASSIC ASP VB

SEND SMS

```
username      = "myUsername"
password      = "myPassword"
sender        = "Sendega"
destination   = "4790222333"
pricegroup    = "0"
contentTypeID = "1"
contentHeader = ""
content       = "Hello world!"
dlrUrl        = "http://myserver.example/myDlrUrl.asp"
ageLimit      = "0"
extID         = ""
sendDate      = ""
refID         = ""
priority      = 0
gwID          = 0
pid           = 0
dcs           = 0

Set objXml = Server.CreateObject("MSXML2.ServerXMLHTTP")
URL = "https://smsc.sendega.com/content.asmx/Send"_
    & "?username=" & username _
    & "&password=" & password _
    & "&sender=" & sender _
    & "&destination=" & destination _
    & "&pricegroup=" & pricegroup _
    & "&contentTypeID=" & contentTypeID _
    & "&contentHeader=" & contentHeader _
    & "&content=" & content _
    & "&dlrUrl=" & dlrUrl _
    & "&ageLimit=" & ageLimit _
    & "&extID=" & extID _
    & "&sendDate=" & sendDate _
    & "&refID=" & refID _
    & "&priority=" & priority _
    & "&gwID=" & gwID _
    & "&pid=" & pid _
    & "&dcs=" & dcs

objXml.Open "GET", URL, False
objXml.Send
serverResponse = objXml.responseText

If InStr(1,serverResponse, "true", 1) Then
       Response.Write("Message sent successfully")
Else
       Response.Write("Message not sent. Server response: "& serverResponse)
End If
```

Do NOT use http://myserver.example/myDlrUrl.asp as dlrUrl parameter.
THIS IS ONLY AN EXAMPLE VALUE! Use a valid URL, or set empty if not using delivery reports!

PHP

SEND SMS

SoapClient to send sms (PHP 5)

```php
$params["username"]      = "myUsername";
$params["password"]      = "myPass";
$params["sender"]        = "Sendega";
$params["destination"]   = 4790222333;
$params["pricegroup"]    = 0;
$params["contentTypeID"] = 1;
$params["contentHeader"] = "";
$params["content"]       = "HelloWorld";
$params["dlrUrl"]        = "http://myserver.example/mydlrscript.php";
$params["ageLimit"]      = 0;
$params["extID"]         = "";
$params["sendDate"]      = "";
$params["refID"]         = "";
$params["priority"]      = 0;
$params["gwID"]          = 0;
$params["pid"]           = 0;
$params["dcs"]           = 0;

$content = new SoapClient(
                "https://smsc.sendega.com/Content.asmx?wsdl",
                array( 'trace'      => true,
                       'exceptions' => true,
                       'encoding'  => 'ISO-8859-1'
                     )
                );

$response = $content->Send($params);
$serverResult = $response->SendResult;

if( $serverResult->Success )
    echo "Message was sent. Id: ".$serverResult->MessageID;
else
    echo "Message was not sent. Errornumber: ".$serverResult->ErrorNumber.", Errormessage:
".$serverResult->ErrorMessage;
```

Do NOT use http://myserver.example/mydlrscript.php as dlrUrl parameter.
THIS IS ONLY AN EXAMPLE VALUE! Use a valid URL, or set empty if not using delivery reports!

Http post/get by file_get_contents(). For PHP v4.3.0 or newer.

```
$serverResult = @file_get_contents("https://smsc.sendega.com/Content.asmx/Send".
                                   "?username=myUsername".
                                   "&password=myPassword".
                                   "&sender=Sendega".
                                   "&destination=4790222333".
                                   "&pricegroup=0".
                                   "&contentTypeID=1".
                                   "&contentHeader=".
                                   "&content=".urlencode("Hello World!").

"&dlrUrl=".urlencode("http://myserver.example/mydlrscript.php").
                                   "&ageLimit=0".
                                   "&extID=".
                                   "&sendDate=".
                                   "&refID=".
                                   "&priority=0".
                                   "&gwid=0".
                                   "&pid=0".
                                   "&dcs=0");
$xmlDocument = simplexml_load_string($serverResult);

if( $xmlDocument->SendResult->Success == 'True' )
    echo "Message was sent. Message id: ".$xmlDocument->SendResult->MessageID;
else
    echo "Message was not sent. Errorcode: ".$xmlDocument->SendResult->ErrorNumber.",
ErrorMessage: ".$xmlDocument->SendResult->ErrorMessage;
```

Do NOT use http://myserver.example/mydlrscript.php as dlrUrl parameter.
THIS IS ONLY AN EXAMPLE VALUE! Use a valid URL, or set empty if not using delivery reports!

Http post/get using fopen() for older versions of php.

```
$url = "https://smsc.sendega.com/sendSMS.asmx/Send".
        "?username=myUsername".
        "&password=myPassword".
        "&sender=Sendega".
        "&destination=4790222333".
        "&priceGroup=0".
        "&contentTypeID=1".
        "&contentHeader=".
        "&content=".urlencode("Hello World!").
        "&dlrUrl=".urlencode("http://myserver.example/mydlrscript.php").
        "&ageLimit=0".
        "&extID=".
        "&sendDate=".
        "&refID=".
        "&priority=0".
        "&gwid=0".
        "&pid=0".
        "&dcs=0";

$handle = fopen($url, "r");
$serverResult = '';

while (!feof($handle)) {
  $serverResult .= fread($handle, 8192);
}

fclose($handle);

preg_match('/\<Success\>(.+)\<\/Success\>/i', $serverResult, $success);
preg_match('/\<MessageID\>(.+)\<\/MessageID\>/i', $serverResult, $messageID);
preg_match('/\<ErrorNumber\>(.+)\<\/ErrorNumber\>/i', $serverResult, $errorNumber);
preg_match('/\<ErrorMessage\>(.+)\<\/ErrorMessage\>/i', $serverResult, $errorMessage);

if( $success == 'True' )
    echo "Message was sent, ID: ".$messageID;
else
    echo "Message eas not sent. Errorcode: ".$errorNumber.", Errormessage:
".$errorMessage;
```

Do NOT use http://myserver.example/mydlrscript.php as dlrUrl parameter.
THIS IS ONLY AN EXAMPLE VALUE! Use a valid url, or set empty if not using delivery reports!

RUBY ON RAILS

SEND SMS

```
require 'soap/wsdlDriver'
require 'uri'

wsdl_url = "https://smsc.sendega.com/Content.asmx?WSDL"
content = SOAP::WSDLDriverFactory.new(wsdl_url).create_rpc_driver

result = content.Send(
  :username => 'myUsername',
  :password => 'myPassword',
  :sender => 'Sendega',
  :destination => '4790222333',
  :pricegroup => 0,
  :contentTypeID => 1,
  :contentHeader => '',
  :content => 'Hello world!',
  :dlrUrl => 'http://myservicer.com/myDlrScript.ror',
  :ageLimit => 0,
  :extID => '',
  :sendDate => '',
  :refID => '',
  :priority => 0,
  :gwID => 0,
  :pid => 0,
  :dcs => 0)

if result.sendResult.success == “true”
    print "Message was sent. ID: " + result.MessageID
else
    print "Message was not sent. Errorcode: " + result.ErrorNumber + ", Errormessage: " +
          result.ErrorMessage
end
```

Do NOT use http://myserver.example/mydlrscript.ror as dlrUrl parameter.
THIS IS ONLY AN EXAMPLE VALUE! Use a valid url, or set empty if not using delivery reports!

PYTHON

SEND SMS

```python
import suds   # https://fedorahosted.org/suds/wiki/Documentation
from suds.client import Client
import logging

#logging.basicConfig(level=logging.INFO)
#logging.getLogger('suds.client').setLevel(logging.DEBUG)
#logging.getLogger('suds.transport').setLevel(logging.DEBUG)
#logging.getLogger('suds.wsdl').setLevel(logging.DEBUG)
#logging.getLogger('suds.xsd.schema').setLevel(logging.DEBUG)


url = "https://smsc.sendega.com/Content.asmx?WSDL"

client = Client(url)
client.set_options(port='ContentSoap')

#print client

result = client.service.Send(
    username = "myUsername",
    password = "myPassword",
    sender = "Sendega",
    destination = "4790222333",
    pricegroup = 0,
    contentTypeID = 1,
    contentHeader = "",
    content = "Hello World!",
    dlrUrl = "http://myserver.example/mydlrUrl/",
    ageLimit = 0,
    extID = "",
    sendDate = "",
    refID = "",
    priority = 0,
    gwID = 0,
    pid = 0,
    dcs = 0)

print result

# (SendResult){
#   MessageID = "703cd4f8-b2f7-4396-853c-60b98388a8d0"
#   ErrorNumber = 0
#   Success = True
# }

print result.MessageID
```

Do NOT use http://myserver.example/mydlrurl/ as dlrUrl parameter.
THIS IS ONLY AN EXAMPLE VALUE! Use a valid url, or set empty if not using delivery reports!

MISCELLANEOUS

SENDING VCARD / VCALENDAR

Standard sending method for sending vCard through Sendega sms gateway is to send the message hexadecimal encoded with header 06050423F40000 and the vCard text encoded in the content field in the following format:

```
BEGIN:VCARD<CR><NL>
VERSION:2.1<CR><NL>
N:Doe;John<CR><NL>
TEL;CELL;PREF:+441231234<CR><NL>
END:VCARD<CR><NL>
```

Note that <CR><NL> is encoded 0d0a. DCS for sending vCard is 20 decimal, and pid is 0. For sending the vcard as MMS, see documentation for sending mms. The .vcf file is then attached in the .zip-file while sending, and added to the .smil-file as a "ref" element.

Issue with vCard and vCal to Android and iPhone

It seems to be a known fact that vCal and vCard sent to Android phones and iPhones is not received correctly. Some android phones shows the vCard content as plain text in a notification window, with no possibility to save, and some just discard the content and does not show anything, even though the phone sends an "received" ack to the operator. This last mentioned behavior is also noticed at iPhones.

It seems the only way to send a vCard/vCal to an Android phone or an iPhone is to attach a vCard/vCal file in an MMS.

Note: In newer Android phones, this last method, of sending vCard in MMS, is not supported. It must therefore be considered that Android phones cannot receive vCard or vCal – and vCard / vCal must be considered an obsolete standard. vCal and vCard has never been standardized in any 3GPP protocol documents, and is the reason why this is not implemented in newer phones.

Android phones can, however, open a .vcf-file if downloaded from the Internet. A possible solution will therefore be to send an wap link or sms with url to a vCard online.

TEXT ENCODING

GSM7 is the 7 bit alphabet used by default in GSM sms messages. A sms message is 140 bytes long, and in order to send longer messages than 140 characters, there has been developed a 7 bit encoding set - called GSM7. This allows 160 characters to be sent, using only 140 bytes. But this alphabet is considerably reduced, and not all characters are supported. Some characters will also count as 2 bytes (See extended table).

When using our HTTP API, use the "ISO-8859-1 Latin1 Hexadecimal" values to encode your request (with leading %). E.g.: "support@sendega.com" should be encoded: "support%40sendega.com". Characters with green background in the list do not have to be encoded. Characters in red are not possible to send using HTTP API.

One sms is 160 characters (consider characters in extended table as two), for sms with length more than 160 characters each part of the concatenated sms message are 153 characters. This gives:

Number of sms	Maximum number of characters	
	GSM7	Unicode
1	160	70
2	306	134
3	459	201

CHARACTER TABLE (GSM7 AND ISO-8859-1)

Character		GSM 7-bit default alphabet (Decimal)	GSM 7-bit default alphabet (hexadecimal)	ISO-8859-1 Latin 1 (Decimal)	ISO-8859-1 Latin 1 (Hexadecimal)
@	At sign	0	00	64	40
£	Pound sign	1	01	163	A3
$	Dollar sign	2	02	36	24
¥	Yuan/Yen sign	3	03	165	A5
è	Small letter e with grave accent	4	04	232	E8
é	Small letter e with acute accent	5	05	233	E9
ù	Small letter u with grave accent	6	06	129	F9
ì	Small letter i with grave accent	7	07	236	EC
ò	Small letter o with grave accent	8	08	242	F2
Ç	Capital letter C with cedilla	9	09	199	C7
<LF>	Line feed	10	0A	10	0A
Ø	Capital letter O with stroke	11	0B	216	D8
ø	Small letter o with stroke	12	0C	248	F8
<CR>	Carriage return	13	0D	13	0D
Å	Capital letter A with ring	14	0E	197	C5
å	Small letter a with ring	15	0F	229	E5
Δ	Capital letter Greek delta	16	10		
_	Underscore	17	11	95	5F
Φ	Capital letter Greek phi	18	12		
Γ	Capital letter Greek gamma	19	13		
Λ	Capital letter Greek lambda	20	14		
Ω	Capital letter Greek omega	21	15		
Π	Capital letter Greek pi	22	16		
Ψ	Capital letter Greek psi	23	17		

Σ	Capital letter Greek sigma	24	18		
Θ	Capital letter Greek theta	25	19		
Ξ	Capital letter Greek xi	26	1A		
<ESC>	Escape	27	1B	27	1B
Æ	Capital letter Æ	28	1C	198	C6
æ	Small letter æ	29	1D	230	E6
ß	Small letter German eszett	30	1E	223	DF
É	Capital letter E with acute accent	31	1F	201	C9
<space>	Space	32	20	32	20
!	Exclamation mark	33	21	33	21
"	Quotation mark	34	22	34	22
#	Number sign	35	23	35	23
¤	Currency sign	36	24	164	A4
%	Percent sign	37	25	37	25
&	Ambersand	38	26	38	26
'	Apostrophe	39	27	39	27
(	Left parenthesis	40	28	40	28
)	Right parenthesis	41	29	41	29
*****	Asterisk	42	2A	42	2A
+	Plus sign	43	2B	43	2B
,	Comma	44	2C	44	2C
-	Minus sign / hyphen	45	2D	45	2D
.	Full stop / period	46	2E	46	2E

Character		**GSM 7-bit default alphabet (Decimal)**	**GSM 7-bit default alphabet (hexadecimal)**	**ISO-8859-1 Latin 1 (Decimal)**	**ISO-8859-1 Latin 1 (Hexadecimal)**
/	Slash	47	2F	47	2F
0	Digit zero	48	30	48	30
1	Digit one	49	31	49	31
2	Digit two	50	32	50	32
3	Digit three	51	33	51	33
4	Digit four	52	34	52	34
5	Digit five	53	35	53	35
6	Digit six	54	36	54	36
7	Digit seven	55	37	55	37
8	Digit eight	56	38	56	38
9	Digit nine	57	39	57	39
:	Colon	58	3A	58	3A
;	Semicolon	59	3B	59	3B
<	Less-than sign	60	3C	60	3C
=	Equals sign	61	3D	61	3D
>	Greater-than sign	62	3E	62	3E
?	Question mark	63	3F	63	3F
¡	Inverted exclamation mark	64	40	64	40
A	Capital letter A	65	41	65	41
B	Capital letter B	66	42	66	42
C	Capital letter C	67	43	67	43
D	Capital letter D	68	44	68	44
E	Capital letter E	69	45	69	45
F	Capital letter F	70	46	70	46
G	Capital letter G	71	47	71	47

H	Capital letter H	72	48	72	48
I	Capital letter I	73	49	73	49
J	Capital letter J	74	4A	74	4A
K	Capital letter K	75	4B	75	4B
L	Capital letter L	76	4C	76	4C
M	Capital letter M	77	4D	77	4D
N	Capital letter N	78	4E	78	4E
O	Capital letter O	79	4F	79	4F
P	Capital letter P	80	50	80	50
Q	Capital letter Q	81	51	81	51
R	Capital letter R	82	52	82	52
S	Capital letter S	83	53	83	53
T	Capital letter T	84	54	84	54
U	Capital letter U	85	55	85	55
V	Capital letter V	86	56	86	56
W	Capital letter W	87	57	87	57
X	Capital letter X	88	58	88	58
Y	Capital letter Y	89	59	89	59
Z	Capital letter Z	90	5A	90	5A
Ä	Capital letter A with dieresis	91	5B	196	C4
Ö	Capital letter O with dieresis	92	5C	214	D6
Ñ	Capital letter N with tilde	93	5D	209	D1
§	Section sign	95	5F	167	A7

Character		**GSM 7-bit default alphabet (Decimal)**	**GSM 7-bit default alphabet (hexadecimal)**	**ISO-8859-1 Latin 1 (Decimal)**	**ISO-8859-1 Latin 1 (Hexadecimal)**
Ü	Capital letter U with dieresis	94	5E	220	DC
¿	Inverted question mark	96	60	191	BF
a	Small letter a	97	61	97	61
b	Small letter b	98	62	98	62
c	Small letter c	99	63	99	63
d	Small letter d	100	64	100	64
e	Small letter e	101	65	101	65
f	Small letter f	102	66	102	66
g	Small letter g	103	67	103	67
h	Small letter h	104	68	104	68
i	Small letter i	105	69	105	69
j	Small letter j	106	6A	106	6A
k	Small letter k	107	6B	107	6B
l	Small letter l	108	6C	108	6C
m	Small letter m	109	6D	109	6D
n	Small letter n	110	6E	110	6E
o	Small letter o	111	6F	111	6F
p	Small letter p	112	70	112	70
q	Small letter q	113	71	113	71
r	Small letter r	114	72	114	72
s	Small letter s	115	73	115	73
t	Small letter t	116	74	116	74
u	Small letter u	117	75	117	75
v	Small letter v	118	76	118	76
w	Small letter w	119	77	119	77

x	Small letter x	120	78	120	78
y	Small letter y	121	79	121	79
z	Small letter z	122	7A	122	7A
ä	Small letter a with dieresis	123	7B	228	E4
ö	Small letter o with dieresis	124	7C	246	F6
ñ	Small letter n with tilde	125	7D	241	F1
ü	Small letter u with dieresis	126	7E	252	FC
à	Small letter a with grave accent	127	7F	224	E0

Rows in red background are not possible to url encode and send using http post/get. As this is not possible, we therefore recommend sending Greek characters as Unicode messages. See chapter “Unicode encoded messages”

EXTENDED GSM7 TABLE

All characters in extended table is counted as two characters

Character		GSM 7-bit default alphabet (Decimal)	GSM 7-bit default alphabet (hexadecimal)	ISO-8859-1 Latin 1 (Decimal)	ISO-8859-1 Latin 1 (Hexadecimal)
<FF>	Form feed	27 10	1B 0A	12	0C
^	Caret / circumflex	27 20	1B 14	94	5E
{	Left curly bracket	27 40	1B 28	123	7B
}	Right curly bracket	27 41	1B 29	125	7D
\	Backslash	27 47	1B 2F	92	5C
[	Left square bracket	27 60	1B 3C	91	5B
~	Tilde	27 61	1B 3D	126	7E
]	Right square bracket	27 62	1B 3E	93	5D
|	Vertical bar / pipeline	27 64	1B 40	124	7C
€	Euro sign	27 101	1B 65	128	80

UNICODE ENCODED MESSAGES

Messages can be sent as Unicode by defining parameter DCS to value 8 (binary 0000 1000), and message text encoded as HtmlEncoded values. Note that Unicode messages have a length of 70 per message for single-message or 67 characters per message part in concatenated messages.

For more information about Unicode messages and message formatting, contact Sendega Customer Service.

MOBILE COUNTRY CODES (MCC) AND MOBILE NETWORK CODES (MNC)

All routing and handling of sms and mms messages in Sendega is based on MCC (mobile country codes) and MNC (mobile network codes) used to identify mobile stations in wireless telephone networks, particularly GSM and UMTS networks. The MCC is part of the International Mobile Subscriber Identity (IMSI) number, which uniquely identifies a particular subscriber and is stored on the SIM card.

A mobile network code (MNC) is used to identify the mobile phone operator/carrier using the GSM, CDMA, iDEN, TETRA and UMTS public land mobile networks and some satellite mobile networks.

A complete list of all MNC codes can be found on our website: http://www.sendega.com/info/sms-support-dokumentasjon.

MCC	Country
412	Afghanistan
276	Albania
603	Algeria
544	American Samoa
213	Andorra
631	Angola
365	Anguilla
344	Antigua and Barbuda
722	Argentina
283	Armenia
363	Aruba
505	Australia
232	Austria
400	Azerbaijan
364	Bahamas
426	Bahrain
470	Bangladesh
342	Barbados
257	Belarus
206	Belgium
702	Belize
616	Benin
350	Bermuda
402	Bhutan
736	Bolivia
218	Bosnia and Herzegovina
652	Botswana
724	Brazil
348	British Virgin Islands
528	Brunei Darussalam
284	Bulgaria
613	Burkina Faso
642	Burundi
456	Cambodia
624	Cameroon
302	Canada
625	Cape Verde
346	Cayman Islands

623	Central African Republic
622	Chad
730	Chile
460	China
732	Colombia
654	Comoros and Mayotte
548	Cook Islands
712	Costa Rica
612	Côte d'Ivoire
219	Croatia
368	Cuba
280	Cyprus
230	Czech Rep.
630	Democratic Republic of the Congo
238	Denmark
638	Djibouti
366	Dominica
370	Dominican Republic
740	Ecuador
602	Egypt
706	El Salvador
627	Equatorial Guinea
657	Eritrea
248	Estonia
636	Ethiopia
750	Falkland Islands
288	Faroe Islands
542	Fiji
244	Finland
208	France
742	French Guiana
547	French Polynesia
628	Gabon
607	Gambia
282	Georgia
262	Germany
620	Ghana
266	Gibraltar
202	Greece
290	Greenland
352	Grenada
535	Guam (US)
704	Guatemala
611	Guinea
632	Guinea-Bissau
738	Guyana
372	Haiti
708	Honduras
454	Hongkong, China
216	Hungary
274	Iceland
404	India
405	India
510	Indonesia
901	International

432	Iran
418	Iraq
272	Ireland
425	Israel
222	Italy
338	Jamaica
440	Japan
441	Japan
416	Jordan
401	Kazakhstan
639	Kenya
545	Kiribati
467	Korea (North)
450	Korea (South)
419	Kuwait
437	Kyrqyzstan
457	Laos
247	Latvia
415	Lebanon
651	Lesotho
618	Liberia
606	Libya
295	Liechtenstein
246	Lithuania
270	Luxembourg
455	Macao China
294	Macedonia
646	Madagascar
650	Malawi
502	Malaysia
472	Maldives
610	Mali
278	Malta
551	Marshall Islands
340	Martinique & Guadeloupe
609	Mauritania
617	Mauritius
334	Mexico
550	Micronesia
259	Moldova
212	Monaco
428	Mongolia
297	Montenegro
354	Montserrat
604	Morocco
643	Mozambique
414	Myanmar
649	Namibia
536	Nauru
429	Nepal
204	Netherlands
362	Netherlands Antilles
546	New Caledonia
530	New Zealand
710	Nicaraqua

614	Niger
621	Nigeria
534	Northern Mariana Islands (US)
242	Norway
422	Oman
410	Pakistan
552	Palau
423	Palestine
714	Panama
537	Papua New Guinea
744	Paraquay
716	Peru
515	Philippines
260	Poland
268	Portugal
330	Puerto Rico (US)
427	Quatar
629	Republic of Congo
647	Reunion
226	Romania
250	Russian Federation
635	Rwanda
356	Saint Kitts and Nevis
358	Saint Lucia
308	Saint Pierre and Miquelon
360	Saint Vincent and the Grenadines
549	Samoa
292	San Marino
626	Sao Tome and Principe
420	Saudi Arabia
608	Senegal
220	Serbia & Montenegro
633	Seychelles
619	Sierra Leone
525	Singapore
231	Slovakia
293	Slovenia
540	Solomon Islands
637	Somalia
655	South Africa
214	Spain
413	Sri Lanka
634	Sudan
746	Suriname
653	Swaziland
240	Sweden
228	Switzerland
417	Syrian Arab Republic
466	Taiwan
436	Tajikistan
640	Tanzania
520	Thailand
514	Timor-Leste (East Timor)
615	Togo
539	Tonga

374	Trinidad and Tobago
605	Tunisia
286	Turkey
438	Turkmenistan
376	Turks and Caicos Islands (UK)
641	Uganda
234	UK
235	UK
255	Ukraine
424	United Arab Emirates
430	United Arab Emirates
431	United Arab Emirates
310	United States
311	Unted States
748	Uruguay
332	US Virgin Islands
434	Uzbekistan
541	Vanuatu
225	Vatican City State
734	Venezuela
452	Vietnam
543	Wallis and Futuna (France)
421	Yemen
645	Zambia
648	Zimbabwe

SMPP – TON AND NPI

TON (Type of number) is used together with NPI (Number plan indicator) to define the number type of the sender and destination. TON and NPI is by default set by Sendega. It's a numeric field with the following values:

TON	Description
0	Unknown. The network has to figure out the format
1	International. International MSISDN with leading country code
2	National. MSISDN without leading country code
3	Network specific. Not widely used
4	Subscriber number. Not used, and usually ignored
5	Alphanumeric
6	Abbreviated number
7	Reserved

P.t. there is not possible to set these parameters manually when sending messages through Sendega at the moment. These parameters are automatically set by Sendega's system. In most cases Sendega will send NPI = 0 and TON = 0, which puts the responsibility of setting these correctly to the operator.

NPI	Description
0	Unknown. Determined by operator. Most often this is used as default.
1	ISDN (E.163/E.164). Telephone number. Some operators require this instead of 0.
2	Reserved
3	Data (X.121)
4	Telex (F.69)
5	Reserved
6	Land mobile (E.212)
7	Reserved
8	National
9	Private
10	ERMES
11	Reserved
12	Reserved
13	Reserved
14	Internet (IP)
15	Reserved
16	Reserved
17	Reserved
18	WAP Client ID (not yet defined)

DOCUMENT CODING SCHEME (DCS)

The DCS parameter is a complex field, defined by a 35 pages document, the 3GPP TS 23.038 (Technical Specification Group Core Network and Terminals; Alphabets and language-specific Information). We'll here try to break it down a bit, and give a couple of examples of how to use this parameter. This will require a basic understanding of binary numbers.

The length of this parameter is 1 byte, which is 1 octet or 8 bits. Each bit (value 0 or 1) in this byte has its own meaning, and also defines how the rest of the bytes should be read. We number these bits from 0 to 7, starting from the right. The DCS parameter is divided into coding groups. E.g. when bit 7 and 6 is set to zero (00xxxxxx) the DCS coding group is called «General data coding indication», and this is the most used group, and this is used by default.

The special case of this group is when bit 7..0 is set to zero (0000 0000). This indicates a standard sms, handled normally, encoded in GSM7 standard alphabet. Another «commonly used» type of this class is when bit 4 is set to 1, indicating that bit 0..1 has a class meaning, and bit 0..1 both set to 0 (class 0) is commonly mentioned as a "flash message", which immediately shows in the mobile phone's display when received. This feature is supported by most of the gateways. To send a message with class 0, encoded in GSM7, the DCS parameter should be set to 16 decimal (bit 4 value 1).

A more «fun» group to look at is group «Message Marked for Automatic Deletion Group», which has bit 7..6 set to 01, and 5..0 follows the same pattern as «General Data Coding Indication». This means that message with DCS set to 64 is a message marked for deletion after read, encoded in GSM7 and sent as a «standard text message». This is an advanced feature, *not* supported by all gateways and we will recommend using “Replace message” described in the next chapter (Protocol ID)

To mention one last group, we can take the group «Message Waiting Indication Group: Discard Message». These messages should be received by the ME (Mobile Equipment), a delivery status should be acked to the operator, but the ME should discard the content and not show this to the end user. This is mentioned as «silent sms» or «sms ping». This group is defined by bit 7..4 set to 1100, bits 3..0 is discarded. This means that the DCS parameter should be set to 192. For this functionality we will also refer to “Discard message” described in the next chapter (Protocol ID) as most handsets haven’t implemented these new DCS features.

PROTOCOL ID (PID)

The Protocol-Identifier is the information element by which the Short Message Transfer Layer (SM-TL) either refers to the higher layer protocol being used, or indicates interworking with a certain type of telematic device. (Ref, 3GPP TS 23.040 V9.0.0 (2009-06), clause 3.2.3)

This is used to indicate that the message should be delivered to another type of unit, like a telex, videotex, Internet Electronic Mail etc. By setting this parameter to 0, you indicate that protocol type should be concluded by SMSC on basis of the DA.

The most well-known use of this parameter for aggregators and end users is the "Replace Short Message" functionality. "Replace Short Message" sets 3 conditions:
- Message sent with same OA
- Message sent with same "Replace Short Message" id
- Message sent through exactly the same SMSC at operator

In order for this to work correctly, the message has to be sent as ContentTypeID=2, with abbreviated number as sender. The pricegroup can be set to zero. We will emphasize that zero rated premium sms has a higher rate than bulk sms. Contact Sendega Support for information about prices associated with your account, and if this feature is supported at your desired destination.

As we can control the first two, the last one is harder to control when it's getting more common that operators use a server cloud for SMSC, and we cannot guarantee that the same exactly server is used for all messages, even when we got a direct connection to the operator. The protocol IDs used for Replace Short Message is bit 7 = 0, bit 6 = 1, and bits 5..0 between 000001 and 000111. This should give decimal values between 65 and 71. Messages sent with e.g 65 will replace other messages with PID 65, but not with any other values.

PID with value bit 7 = 0, 6 = 1 and 5..0 as 000000 is of type "Short Message Type 0", which should be acked by handset, but message contents should be discarded, and not shown in handset. This gives PID parameter value 64 I decimal. (This feature may be replaced by DCS class "Discard Message", bits 7..4 as 1100 for some operators.)

For more information about this parameter, consult 3GPP TS 23.040, or contact Sendega technical support