



2sms Perl XML Tech Guide

About 2sms

2sms allows business customers to send text messages from your computer to cell phones anywhere in the world. We help customers improve operations, demonstrate duty of care, and react more responsively to customers.

Text messaging reaches more people than email and faster than calling.

- ✓ To learn more - www.2sms.com
- ✓ Want a No cost No obligation trial account? - www.2sms.com/register.aspx
- ✓ For technical information - www.2sms.com/software.aspx



For Help, call 2sms on 877-276-7266 (USA), 0800 65 222 77 (UK) or +44 1234 757 800 (Rest of World) or email support@2sms.com. Support is available 24/7.

Using Perl to Send a Text Message

If you wish to send a single message to a single cell phone, use the following Perl script to form up your XML.

```
#!/usr/bin/perl -w
$file="cgi-lib";
do $file || die "Fatal Error: Can't load cgi library - $file\n";
&ReadParse;
use CGI qw(:all);
use LWP::UserAgent;
$ua = new LWP::UserAgent;
$ua->agent("AgentName/0.1 " . $ua->agent); # Create a user agent object

#setup a few things first
my $user = "user\@company.com"; #username
my $pass = "MyPassword"; #password
#my $text=$in{text}; #text passed from previous form
#my $mobnum=$in{mobnum}; #mobile number passed from previous form as well

$text = substr($text,0,160); #just to make sure we dont get anything else in there

$myOutMsg = '<?xml version="1.0" encoding="UTF-8" ?>';
$myOutMsg .= '<Request xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" ';
```

Customer Doc #	SAL-187
Issue No./Date	1.0 25-Jul-07
Classification	N/A

```
'xsi:noNamespaceSchemaLocation="http://schema.2sms.com/1.0/0410_RequestSendMessage.xsd" ';
$myOutMsg .= 'Version="1.0">';
$myOutMsg .= '<Identification>';
$myOutMsg .= '<UserID> . $user .</UserID>';
$myOutMsg .= '<Password>.$pass .</Password>';
$myOutMsg .= '</Identification>';
$myOutMsg .= '<Service>';
$myOutMsg .= '<ServiceName>SendMessage</ServiceName>';
$myOutMsg .= '<ServiceDetail>';
$myOutMsg .= '<SingleMessage>';
$myOutMsg .= '<Destination>.$mobnum.</Destination>';
$myOutMsg .= '<Text>.$text.</Text>';
$myOutMsg .= '</SingleMessage>';
$myOutMsg .= '</ServiceDetail>';
$myOutMsg .= '</Service>';
$myOutMsg .= '</Request>';
```

```
print header, start_html("title of page"); #start the html of the page
```

```
my $req = new HTTP::Request POST => 'http://www.2sms.com/xml/xml.jsp'; #Create a request
$req->content_type('text/xml'); #form our header
$req->content($myOutMsg); #create the content
my $res = $ua->request($req); #Pass request to the user agent and get a response back
```

```
if ($res->is_success) { #Check the outcome of the response
    print $res->content; #you can comment this bit out if you want, it basically just prints the response from the server
    print"<center><font face=\"arial, verdana, helvetica\" size=\"2\">Your message has been sent</center></font>\n";
} else {
    print "<center><font face=\"arial, verdana, helvetica\" size=\"2\">There seems to be an error please hit your back button and try again</font>\n";
}
print end_html; #close of the html
exit; #exit.
```

Customer Doc #	SAL-187
Issue No./Date	1.0 25-Jul-07
Classification	N/A

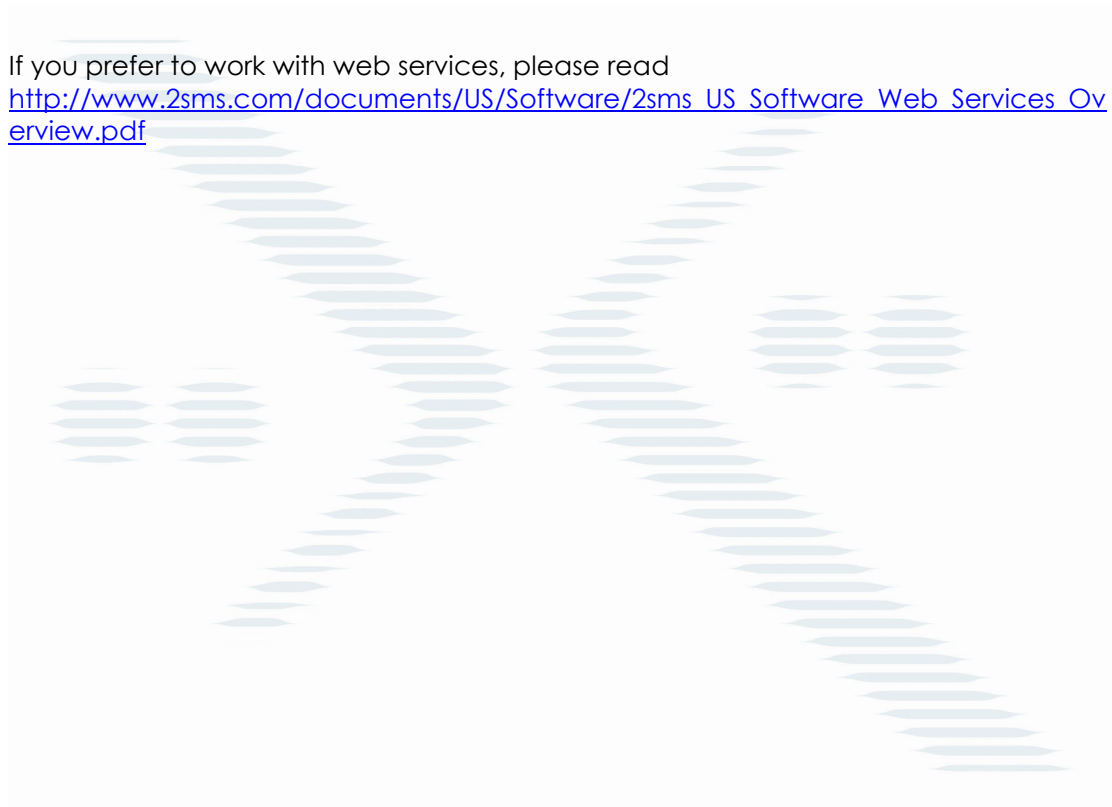
2sms XML Services

2sms supports a range xml services, including sending messages, running reports, and managing your address book held on your 2sms account. Please read http://www.2sms.com/documents/US/Software/2sms_US_Software_XML_Overview.pdf for more information.

You can modify the Python code listed in this document to send XML to access any of the supported xml services, whose details are listed at <http://schema.2sms.com>.

If you prefer to work with SOAP, please read http://www.2sms.com/documents/US/Software/2sms_US_Software_XML_Overview.pdf

If you prefer to work with web services, please read http://www.2sms.com/documents/US/Software/2sms_US_Software_Web_Services_Overview.pdf



Customer Doc #	SAL-187
Issue No./Date	1.0 25-Jul-07
Classification	N/A