Clickatell™
Any message, anywhere.

Content

Change history

Approximately six (6) months of changes are reflected.

Visit http://www.clickatell.com/downloads/http/Clickatell_HTTP.pdf to check for updates to this document.

Version	Date	Section	Changes to Documentation
2.3.9	14 April 2010	5	Updated 'add a registered API connection'

1. Overview

This technical document is intended for developers who wish to use the Clickatell HTTP API for sending messages, and describes the various programming methods and commands used by developers when using this API.

The HTTP API is the most popular API, because there are many ways to utilise it for message sending and it can be used for low or high-volume messaging. As HTTP is a means for relaying information, the HTTP API can be used with practically any web-service application. This is particularly useful for high-volume message sending.

To use this API, you need to register at: www.clickatell.com. When you sign up for an HTTP/S account (https://www.clickatell.com/central/user/client/step1_new.php?prod_id=2), you will be given a username, password and api_id: keep these at hand. Once you have registered and been activated you will receive 10 free credits with which to test our service. Messages sent with these credits contain a canned (pre-populated) message. You can test the API using these credits, and purchase credits to start sending your own, customised messages.

We will cover the **HTTP** method in this document. Additional documentation is available for the other methods. Sample code is provided on the site.

2. Introduction

This is one of the simpler server-based forms of communication to our gateway. It can be used either in the form of a HTTP POST, or as a URL (GET). We recommend POST for larger data transfer, due to the size limitations of GET. Communication to our API can be done either via HTTP on port 80 or HTTPS on port 443. All calls to the API must be URL-encoded. The parameter names are case-sensitive. Batch messaging is catered for in a variety of ways.

Note: It is important that the ENTIRE document is read before contacting support. Parameters are case-sensitive. All examples shown use HTTP GET.

3. Getting started

In order to use the Clickatell gateway you need a Clickatell account and at least one registered connection (API sub-product instance) between your application and our gateway. Each connection method is known as a sub-product (of our API product). Here's how:

Step 1 - register for a Clickatell account

If you do not already have a Clickatell Central account, you need to register for one as per below. If you have a Clickatell Central account, proceed to Step 2 for instructions on how to add an API connection to your account.

• Go to http://www.clickatell.com/products/sms_gateway.php, and choose the appropriate API sub-product (connection method) you wish to use.

• Click on the registration hyperlink.
• Select the Account type you would like to use (Local or International)
• Enter your personal information to complete the registration form
• Accept Terms & Conditions
• Click **Continue** - an email containing an activation link will be sent to the email address you have provided

(Local account type registrations will be shown a confirmation pop-up where you need to confirm that you wish to register for a local account type.)

If you have not received the activation email within a few minutes you can choose to resend it or edit the address provided.

Once you have activated your account by clicking on the link within your activation email, you will be directed to the login page. Once you have logged in you will be on the Clickatell Central landing page and you can use your free SMS test credits. Please note that for security reasons these 10 credits contain pre-set Clickatell content.

To start sending personalised messages, you need to verify your mobile number so you can purchase credits. The verification steps appear in the grey block at the bottom of the Clickatell Central landing page. This block will be visible until you have verified your mobile number.

Check the mobile number that you have entered and, if it is correct, click **Send Activation Code**. A verification code will be sent to your mobile number. If you would like the code to be sent to a different number, replace the number provided and follow the same steps. Enter the verification code in the space provided and click the **Verify Now** button.

When your mobile number is verified, you will be able to purchase credits and send messages. To add an API connection, please follow the steps detailed below.

Step 2 - add a registered API connection (sub-product)

If you are not already logged into your account, then you must do so at http://www.clickatell.com/login.php

• Select **Manage my Products** from the top menu.
• Select **Get Connections** from the **Manage my Products** submenu in the left pane.
• Select the API you wish to add to your account.
• Complete the details.

After successfully adding a connection, a confirmation message will be displayed with a unique API ID and information on how to get started.

 The getting started section displays the API connection parameters and authentication details. These details are required when connecting to the Clickatell gateway to send a message.

Note: For more information on managing your API connections within your Clickatell account see our API guide at http://support.clickatell.com/guides/clickatell/api_guide.php.

4. Basic commands

In order to send a message, the system will firstly need to authenticate you as a valid user. The preferred method of authentication is using the *auth* command. Whilst it involves an additional step, it is far more secure in that you only have to pass login details once, to obtain a session ID. If you do not use auth to obtain a session ID, you will have to pass your account details with every command.

All other commands are then made up of three segments: authentication, the basic message components (message content and recipients) and the additional message parameters. In the examples below, we will include the authentication and basic message components. The additional message parameters will be included only where they are relevant.

Basic Command Structure:

URL   Call Authentication Message Parameters

http://api.clickatell.com/http/sendmsg?session_id=xxxx&to=xxxx&text=xxxx
or
http://api.clickatell.com/http/sendmsg?api_id=xxxx&user=xxxx&password=xxxx&to=xxxx&text=xxxx

4.1 Authentication and session IDs

In order to deliver a message, the system needs to authenticate the request as coming from a valid source. We use a number of parameters to achieve this:

- **api_id:** This is issued upon addition of an HTTP sub-product to your account. A single account may have multiple API IDs associated with it.
- **user:** This is the username of your account.
- **password:** The current password you have set on your account.

Additionally we can force an IP lockdown, allowing only requests sent from IP addresses that you have specified. This can be set under the API product preferences within your account. Please ensure that after testing, you remove all unnecessary IP addresses in your preferences to tighten up on security.

You can have multiple sessions open, however the session ID will expire after fifteen minutes of inactivity. You will then have to re-authenticate to receive a new session ID. Alternatively, you can ping every 10 minutes or so to ensure that the current session ID is kept live.

Command:

Not encrypted: http://api.clickatell.com/http/auth?api_id=xxxx&user=xxxx&password=xxxx
Encrypted: https://api.clickatell.com/http/auth?api_id=xxxx&user=xxxx&password=xxxx

Response:
OK: Session ID
or:
ERR: Error number, error description

This session ID must be used with all future commands to the API, unless you authenticate each time within the command itself.

4.2   Ping

This command prevents the session ID from expiring in periods of inactivity. The session ID is set to expire after 15 minutes of inactivity. You may have multiple concurrent sessions using the same session ID.

Command:
http://api.clickatell.com/http/ping?session_id=xxx

Response:
OK:
or:
ERR: Error number, error description

4.3   Send a message

To facilitate sending an SMS with a single command, we have included the ability to post **api_id**, **user** and **password** variables in *sendmsg*. This is only required if you do not authenticate yourself using the authentication command (*auth*). Using a session ID is preferred to authenticating each time.

One can send to multiple destination addresses by delimiting the addresses with commas. The basic parameters required are **to** (the handset number to which the message is being sent) and **text** (the content of the message). A maximum of 100 comma separated destination addresses per *sendmsg,* or *quicksend* command, are possible, if you are calling the command via a GET, or alternatively, 300 destination addresses if you are submitting via a POST.

In the examples displayed in this document we will only refer to these basic parameters. Other parameters may be used to enable different features. These are discussed in the following section.

Each message returns a unique identifier in the form of an API message ID. This can be used to track and monitor any given message. The API message ID (**apiMsgid)** is returned after each post.

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxxx&to=xxxx&text=xxxx
or:
http://api.clickatell.com/http/sendmsg?api_id=xxxx&user=xxxx&password=xxxx&to=xxxx&text=xxxx

Response Single Message:
ID: apimsgid
Response Multiple Messages:
ID: apimsgid To: xxxxxx
ID: apimsgid To: xxxxxx

Or:

Response Single Message:
ERR: Error number, error description
Response Multiple Messages:

ERR: Error number, error description
ERR: Error number, error description

4.4 Query a message

This command returns the status of a message. You can query the status with either the **apimsgid** or **climsgid**. The API Message ID (**apimsgid**) is the message ID returned by the Gateway when a message has been successfully submitted. If you specified your own unique client message ID (**climsgid)** on submission**,** you may query the message status using this value. You may also authenticate with **api_id**, **user** and **password**.

See Appendix B for status codes.

Command:
http://api.clickatell.com/http/querymsg?session_id=xxx&apimsgid=XXXXX
or:
http://api.clickatell.com/http/querymsg?user=xxxx&password=xxxx&api_id=xxxx& apimsgid=XXXXX

Response:
ID: xxxx Status: xxxx
Or:
ERR: Error number, error description

Note: Clickatell can also post message status updates to your application via means of a *Callback URL*. This is the recommended method to obtain message status updates as your application is not required to continually poll the Clickatell gateway. Detailed information can be found in the "Callback URL" section under "Message parameters".

Message status reports can be viewed online within your Clickatell Central account. These reports can also be exported in CSV or Excel format.

5. Message parameters

5.1 Table of parameters

There are a variety of messaging and SMS features supported by the gateway, which can be activated by including a number of additional parameters. These parameters include those in the table below. Parameters are case-sensitive.

Name	Parameter name	Short description	Default value	Restricted values
API product ID	api_id	The value for this **mandatory** parameter can be found or created by logging in online and going to Manage My Products.		
Username	user	The username you specified.		

Name	Parameter name	Short description	Default value	Restricted values
Password	password	Your account password.		
Session ID	session_id	The session ID from the *auth* command. Not applicable to the FTP, SMPP or SMTP APIs.		
Destination address	to	The number of the handset to which the message must be delivered. The number should be in international number format.		No '00' prefix or leading "+" symbol should be used.
Text	text	The text content of the message. Note that some characters take up two characters because of GSM encoding standards		http://support.clickatell.com/faq.php?mode=view_entry&kbid=121&kbcat=26
Source address	from	The source/sender address that the message will appear to come from also known as "Sender ID". These must be registered within your online account and approved by us before they may be used. MO numbers rented from us do not require approval.	gateway assigned number	A valid international format number between 1 and 16 characters long, or an 11 character alphanumeric string.
Enable callback	callback	Enables you to receive message delivery statuses via an HTTP, SOAP or XML callback which is posted to a URL of yours using the **GET** or **POST** method. This is done every time a message status is updated.	0	0,1,2,3 Read detailed description of parameter.
Delivery time	deliv_time	Delays delivery of SMS to mobile device in minutes relative to the time at which the SMS was received by our gateway. This should be greater than 10 minutes for best effect. Smaller time frames may be delivered too soon.		The upper limit is 7 days, or 10080 minutes.
Concatenation	concat	Enables you to send messages longer than a standard message.	1	1,2,3
Maximum credits	max_credits	Overrides the maximum charge specified online in "profiles". It works within the bounds of the profiles. In other words a profile must exist for the maximum credit that you set.	As per profiles	0.8,1,1.5,2,2.5,3
Required features	req_feat	Some parameters and features are not set as "required" by default, and may be dropped if the		Read detailed description of parameter.

Name	Parameter name	Short description	Default value	Restricted values
		least-cost route does not support them. This parameter allows you to ensure that the features set when an SMS is sent are supported by the gateway used. This could increase the cost per message if a more expensive gateway is used.		
Delivery queue	queue	Delivers the message through one of three queues assigned to each client account. Messages in the highest priority queue will be delivered first.	3	1, 2,3 1 is highest priority.
Gateway escalation	escalate	Prompts an escalation to an alternative route, if messages are queued on the least-cost route.	0	0 - off 1 - Escalate immediately to an alternative route if messages are queued on the least-cost route.
Mobile originated	mo	We route via a pre-defined carrier to enable the ability for a reply to be received back. This is only applicable to clients that have subscribed to a two-way messaging service.	0	0 – Off. We use our normal routing rules. 1 – Enable Reply.
Client message ID	cliMsgId	Client message ID defined by user for message tracking.		Up to 32 alphanumeric characters. No spaces.
Unicode message	unicode	Two digit language code. Convert your text to Unicode [UCS-2 encoding]. See http://www.Unicode.org/.	0	0 – No Unicode 1 – Send as Unicode.
Message type	msg_type	Optional parameter which must be set to send specially formatted messages; e.g. logos and ringtones.	SMS_TEXT	
User data header	udh	Allows you to set your own message types. Do not use if you set the **message type** parameter. When set, Clickatell assumes the data is 8 bit. See 8 bit messaging for more information.		Set UDH data manually.
Data	data	The data content of a message, if the UDH component is set		

Name	Parameter name	Short description	Default value	Restricted values
		manually.		
Validity period	*See detailed information on message parameter*	The validity period in minutes relative to the time at which the SMS was received by our gateway. The message will not be delivered if it is still queued on our gateway after this time period.	1440 minutes (24 hours)	Set value in X minutes from 1 – 1440 minutes.

Additional parameters are also available to the HTTP API:

Name	Parameter name	Short description	Default value	Restricted values
Binary	binary	Optional parameter to force the message content to be sent as binary (8-bit) data.		0 - off 1 - force binary
Scheduled Time	scheduled_time	Specify when a message gets delivered.		

5.2   Message parameters in detail

5.2.1   Destination address (to)

SMS messages need to be sent in the standard international format, with country code followed by number. No leading zero to the number and no special characters such as "+" or spaces must be used. For example, a number in the UK being 07901231234 should be changed to 447901231234.

If have you set the preferred dial prefix preference within your client account after logging in on-line, any mobile numbers starting with zero will have the zero stripped and replaced with your default prefix. If the mobile number does not have a zero, the default prefix will not be changed.

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx

5.2.2   Source address (from)

The source address (**from**), also known as the sender ID, can be either a valid international format number between 1 and 16 characters long, or an 11 character alphanumeric string. These must be registered within your online account and approved by us before they may be used. MO numbers rented from us do not require approval

Note that characters such as spaces, punctuation, Unicode and other special characters may not always be supported to all destinations and could interfere with your delivery. We suggest that you refrain from using such characters on the source address. If this is set, then delivery acknowledgements may be unavailable. The use of an alphanumeric source address with 8-bit messaging may cause message failure. This service is not guaranteed across all mobile networks and may interfere with delivery to certain handsets.

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx&from=xxxx

Note: To ensure that this feature is supported when delivering your message, the required features (**req_feat**) parameter for this feature must be set.

5.2.3 Delivery acknowledgment (deliv_ack)

In order to determine whether an SMS has been received by a handset or not, we request delivery acknowledgement for every message we send. The ability to receive reliable delivery acknowledgements varies between mobile networks. Please test to a specific mobile network first, before assuming that you will receive handset acknowledgments for messages that are delivered.

If a GSM handset is 'absent', e.g. switched off or out of coverage, the SMS will be delivered according to a retry cycle once the handset is back in coverage. A delivery receipt will only be returned if and when the retry is delivered. If the validity period or retry cycle (typically 24 hours) is exceeded, the SMS will fail and show 'Error Delivering Message' or status 8.

A delivery acknowledgment can be monitored via the callback URL or online reports.

5.2.4 Callback URL (callback)

Final or intermediary statuses are passed back by the API depending on the **callback** value set in the original post. This is done by means of:

- HTTP GET
- HTTP POST
- XML GET
- XML POST
- SOAP GET
- SOAP POST
.

The callback URL and optional "Username" and "Password" authentication parameters can be set in the preferences section of the particular API product within your client account, after logging in online. The URL must begin with either *http://* (non-encrypted) or *https://* (encrypted). These are NOT your Clickatell username and password but are a username and password of your choice to add additional security.

The variables returned are **apiMsgId**, **cliMsgId**, **to**, **timestamp**, **from**, **status** and **charge.**

Callback value	Message status types returned	Message status code returned
0	No message status returned.	
1	Returns only intermediate statuses.	002, 003, 011
2	Returns only final statuses of a message.	004, 005, 006, 007, 008, 010, 012
3	Returns both intermediate and final statuses of a message.	All except 001

Examples

- **HTTP**

Sample callback to your callback URL using an HTTP get:

https://www.yoururl.com/script.asp?api_id=12345&apiMsgId=996f364775e24b8432f45d77da8eca47&cliMsgId=abc123&timestamp=1218007814&to=279995631564&from=27833001171&status=003&charge=0.300000

- **XML**

The following data is sent in XML MT callbacks in a parameter called 'data':

```
<?xml version="1.0"?>
<callback>
 <apiMsgId>996411ad91fa211e7d17bc873aa4a41d</apiMsgId>
 <cliMsgId></cliMsgId>
 <timestamp>1218008129</timestamp>
 <to>279995631564</to>
 <from>27833001171</from>
 <charge>0.300000</charge>
 <status>004</status>
</callback>
```

Sample callback to your callback URL using an **XML GET:**

https://www.yoururl.com/script.php?data=<?xml version="1.0"?><callback><apiMsgId>996411ad91fa211e7d17bc873aa4a41d</apiMsgId><cliMsgId></cliMsgId><timestamp>1218008129</timestamp><to>279995631564</to><from>27833001171</from><charge>0.300000</charge><status>004</status></callback>

- **SOAP**

With the SOAP callback method, a SOAP packet will be sent with a parameter called 'data'. Below is an example packet that will be sent to you via GET or POST.

Example of a SOAP packet that will be sent to you via **GET** or **POST:**

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<SOAP-ENV:Envelope SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns="mt_callback">
 <SOAP-ENV:Body>
  <tns:mt_callback xmlns:tns="mt_callback">
    <api_id xsi:type="xsd:int">1234</api_id>
```

```
      <apimsgid xsi:type="xsd:string">2e838df2ee3ea418272ae05aaf84ce5d</apimsgid>
      <climsgid xsi:type="xsd:string">abc123</climsgid>
      <to xsi:type="xsd:string">27999123456</to>
      <from xsi:type="xsd:string">27999000224</from>
      <timestamp xsi:type="xsd:int">1213690834</timestamp>
      <status xsi:type="xsd:int">003</status>
      <charge xsi:type="xsd:float">0.300000</charge>
    </tns:mt_callback>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

This is an example callback URL that will be sent to your application:

http://www.yoursite.com/your_url.php?data="<?xml version="1.0" encoding="ISO-8859-1"?><SOAP-ENV:Envelope SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns="mt_callback"><SOAP-ENV:Body><tns:mt_callback xmlns:tns="mt_callback"><api_id xsi:type="xsd:int">1234</api_id><apimsgid xsi:type="xsd:string">2e838df2ee3ea418272ae05aaf84ce5d</apimsgid><climsgid xsi:type="xsd:string">abc123</climsgid><to xsi:type="xsd:string">27999123456</to><from xsi:type="xsd:string">27999000224</from><timestamp xsi:type="xsd:int">1213690834</timestamp><status xsi:type="xsd:int">003</status><charge xsi:type="xsd:float">0.300000</charge></tns:mt_callback></SOAP-ENV:Body></SOAP-ENV:Envelope>"

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx&callback=3
or, with encryption:
https://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx&callback=3

5.2.5   Delivery time (deliv_time)

The delivery of an SMS message may be delayed by setting an amount of time in **minutes** relative to the time at which it was received by our gateway. We will store the message until the required time frame has elapsed. The maximum delay time is 10080 minutes or 7 days.

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx&deliv_time=120

Response:
ID: xxxxx
or
ERR: Error number, error description

When sending batches of messages, the delivery time should be set in the *startbatch* command. This will ensure that all messages are delivered X minutes after being posted to the Gateway.

5.2.6   Concatenation (concat)

If this value is set to 1, 2 or 3 the message will span across 1, 2 or 3 SMS messages where applicable. One text SMS will be sent for every 160 characters or 140 bytes. If a message is concatenated, it reduces the number of characters contained in each message by 7. With 8-bit concatenated messages, each SMS can support up to 140 bytes including the UDH headers.

Please also see http://support.clickatell.com/faq.php?mode=view_entry&kbid=121&kbcat=26 for information on characters that require two character places.

Please be aware that a single Unicode SMS can only contain a maximum of 70 characters. If a Unicode message is concatenated, it reduces the number of characters contained in each message part by 7.

Values set are:

Value	Status
1	Default - No concatenation: only 1 message.
2	Concatenate a maximum of 2 messages.
3	Concatenate a maximum of 3 messages.
N	Concatenate a maximum of N messages. (Delivery is dependent on mobile and gateway. A maximum of 3 is recommended).

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx&concat=2

5.2.7   Maximum credits (max_credits)

This parameter overrides the maximum charge associated with message delivery, as set by the profiles selected within your client account after logging in online. This parameter can be used to limit the cost of a message to a particular value and is bound by the maximum credit value specified in your profiles.

A valid API message ID can still be returned for messages that are not delivered as a result of the maximum credits value set. These messages will have a status of routing error (009).

The credit value in this parameter can be set to any amount of credits. To set your delivery profile, go to **Central**>>**My Settings**>>**Routing Profiles**.

5.2.8   Required features (req_feat)

This parameter specifies the features that must be present in order for message delivery to occur. If all features are not present, the message will not be delivered. This prevents SMS messages arriving at a destination via the least-cost gateway, without certain features. This would, for instance, prevent the dropping of an sender ID.

This means that we will not route messages through a gateway that cannot support the required features you have set. For certain message types, we always set the required feature bitmask where relevant. These are FEAT_8BIT, FEAT_UDH, FEAT_UCS2 and FEAT_CONCAT.

This parameter is set using a combined decimal number to refer to the additional required features.

E.g.: 32 + 512 = 544 – Numeric sender ID and Flash SMS both required.
The value you would set to ensure that Flash and numeric sender ID are both supported, would therefore be *544.*
To ensure that delivery acknowledgment and alphanumeric IDs are supported you would use the value *8240* (16 + 32 + 8192)*.*

Hex value	Decimal	Feature	Description
0x0001	1	FEAT_TEXT	Text – set by default.
0x0002	2	FEAT_8BIT	8-bit messaging – set by default.
0x0004	4	FEAT_UDH	UDH (Binary) - set by default.
0x0008	8	FEAT_UCS2	UCS2 / Unicode – set by default.
0x0010	16	FEAT_ALPHA	Alpha source address (from parameter).
0x0020	32	FEAT_NUMER	Numeric source address (from parameter).
0x0200	512	FEAT_FLASH	Flash messaging.
0x2000	8192	FEAT_DELIVACK	Delivery acknowledgments.
0x4000	16384	FEAT_CONCAT	Concatenation – set by default.

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx&req_feat=####

5.2.9   Delivery queue

Setting this parameter will assign the message to one of three queues assigned to each user account. This sets the priority of a message sent to us, relative to other messages sent from the same user account. Messages in queue number 1, will always be delivered before messages in queue number 2 and 3, while messages in the 3rd queue, will have the lowest priority (relative to queues 1 and 2).

This is useful when delivering, for example, a single high priority message while you have a large batch going through that same account. The large batch will be queued through queue number 3 (default), and urgent alerts (sent through queue 1), will be delivered ahead of those messages in the batch (queue 3), regardless of when they are actually sent to us.

Values set are:

Value	Status
1	Use first / primary user queue (highest priority).
2	Use second user queue.
3	Use third user queue (lowest priority) - Default status.

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx&queue=2

5.2.10 Gateway escalation (escalate)

By default, the message router will select the lowest cost route (matching features and reliability) that is available for a given destination.

This parameter ensures that, should a message be delayed due to gateway congestion or some other reason on the initial gateway selected by our router, then alternative routes that match the required features will be sought. This is done by moving through the available gateways in order of increasing cost, up to the maximum charge set by the user either using the parameter that defines the maximum credits or based on the profiles selected.

When urgent and high priority messages are sent, they should be posted with escalate set to 1 (on), combined with a high maximum credit value to ensure that the greatest number of gateways are available.

Values set are:

Value	Status
0	Off – Default value.
1	On - Escalate immediately to an alternative route if the messages are queued on the least-cost route.

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx&escalate=1

5.2.11 Mobile originated (mo)

This parameter is only used when a message is sent to a handset and a reply is expected.

PLEASE NOTE: This parameter is only valid for clients that have signed up and paid for our two-way messaging service. An alternative to our least-cost gateway may be used, which could result in a higher cost per message. Please email Clickatell support **for pricing or view online.**

When sending a normal MT message to a handset and you expect a reply to your registered MO number, please set the **mo** parameter to "1".

Values to set are:

Value	Status
0	Off - Default status. We use the normal routing feature.
1	Enables reply ability. We route via a pre-defined carrier to enable the ability to reply.

It is important that the user specifies the correct **from** parameter together with this parameter. If no **from** parameter is specified, we will use a default originator number as set by Clickatell. You will NOT receive these replies.

If you specify the originator (the purchased mo number), then we will route the message such that it can be replied to by the recipient. This reply will be sent to you.

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx&mo=#

5.2.12 Client message ID (climsgid)

This parameter is set by the user to enable internal message tracking. It allows the user to set their own tracking ID for each message. Once set for a given message, this may be used in place of the Clickatell issued API message ID (apimsgid) for querying message.

A client message ID (**climsgid**) may be any combination of alphanumeric characters excluding spaces. A maximum of 32 characters may be used.

Client message IDs may be used with the *querymsg* command.

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx&climsgid=xxxx

5.2.13 Unicode (unicode)

If this value is set to 1, the text field must contain two-byte Unicode. Each SMS can handle a maximum of 70 characters. Each Unicode character must be hex-encoded. More information is available at http://www.Unicode.org/.

Note: When using the batch send facility for delivering Unicode messages, it is not possible to substitute variables into the message content. This is only possible with Germanic characters.

Values set are:

Value	Status
0	Off - default status.
1	On - delivers the text as two-byte Unicode.

We provide a converter to convert text to Unicode within your client account online. Go to "Converters" from within your account online.

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx&unicode=1
Eg.: ΩΨ Θ becomes: ….&text=03A903A80398&unicode=1

5.2.14 Message type (msg_type)

A wide variety of messages can be sent through our gateway. We have pre-defined a number of SMS message-types in the API, so that you do not have to set the UDH (user data header) manually. You may optionally set the UDH rather than using one of the message types set below. Message types are case sensitive.

For non-Nokia message types (EMS, etc.), please generate your own UDH and data according to the manufacturers specifications of the message type you wish to send.

This parameter need not be included if the SMS is a standard text message.

Values set are:

Value	Description
SMS_TEXT	This is the default message type. It is optional to specify this parameter.
SMS_FLASH	To send an SMS that displays immediately upon arrival at the phone.
SMS_NOKIA_OLOGO	Send an operator logo to a Nokia handset.
SMS_NOKIA_GLOGO	Send a group logo to a Nokia handset.
SMS_NOKIA_PICTURE	Send a picture message to certain Nokia handsets.
SMS_NOKIA_RINGTONE	Send a ringtone to a Nokia handset.
SMS_NOKIA_RTTL	Send an RTTTL format ringtone to Nokia handsets.
SMS_NOKIA_CLEAN	Remove operator logo from a Nokia handset.
SMS_NOKIA_VCARD	Send a business card to a Nokia handset.
SMS_NOKIA_VCAL	Send an event calendar to a Nokia handset.

Command:
Please see the messaging examples at the end of this document.

5.2.15 Validity period (validity)

A message may be given a time frame for which it is valid. After this period the message will expire. This parameter takes an amount of time in **minutes** relative to the time at which the message was received by our gateway. If the message is queued on our gateway for a period exceeding the validity period set then a routing error of 115 will be returned. The default validity period is 1440 minutes (24 hours).

Note: The validity period is not passed on to the upstream gateway.

Command:
http://api.clickatell.com/http/sendmsg?session_id=xxx&to=xxxx&text=xxxx&validity=120

5.2.16   Scheduled Time

The purpose of this parameter is to allow you to specify when you want a message to be delivered. This parameter is different to the existing deliv_time parameter as it does not specify a delay time, but a delivery time.

The new parameter will accept a delivery time in one of the following formats:
1.   Unix timestamp
2.   Date in time in UTC format (YYYY-MM-DDTHH:mm:ssZ)

Examples:
1) Unix timestamp:
      scheduled_time:1233133393

2) UTC date format:
      scheduled_time:2009-01-30T14:00:00Z

The same limitations of the deliv_time parameter apply here:
1.   The maximum scheduled time range is 7 days
2.   Actual delivery time of scheduled messages can always be handled up to 5 minutes too early

5.2.17   Binary

If this value is set to 1, the parameter will force the message data as binary (8-bit) data.

Our API will only handle a message as 8bit depending on what values has been set in your UDH data. To ensure that your message is sent as 8bit, this parameter can be used. This can also be used when you have no UDH data with a hex string.

Values set are:

Value	Status
1	On – forces message data 8-bit data

Command:
http://api.clickatell.com/http/sendmsg.php?api_id=2995013&user=xyz&password=xyz&to=279991122334&
udh=040402F1F2& data=000102030405060708090A0B0C0D0E0F&binary=1

6. Additional commands

6.1 Delete/stop message

This enables you to stop the delivery of a particular message. This command can only stop messages which may be queued within our router, and not messages which have already been delivered to a SMSC. This command is therefore only really useful for messages with deferred delivery times.

Command:
http://api.clickatell.com/http/delmsg?session_id=xxx&apimsgid=XXXXX
or:
http://api.clickatell.com/http/delmsg?session_id=xxx&climsgid=XXXXX

Response:
ID: xxxx Status: xxxx
or:
ERR: Error number, error description

6.2 Query balance

This will return the number of credits available on this particular account. The account balance is returned as a floating point value.

Command:
http://api.clickatell.com/http/getbalance?session_id=xxx
or
http://api.clickatell.com/http/getbalance?api_id=xxx&user=xxx&password=xxx

Response:
Credit: xxxx.x
or:
ERR: Error number, error description

6.3 Coverage Query

This command enables users to check our coverage of a network or number, without sending a message to that number. Authentication is required for this API call. This call should NOT be used before sending each message.

Command:
http://api.clickatell.com/utils/routeCoverage.php?session_id=xxxx&msisdn=xxxx
or:
http://api.clickatell.com/utils/routeCoverage.php?api_id=xxx&user=xxx&password=xxxx&msisdn=xxxx

Where msisdn is the number you wish to route to.

Response:
OK: This prefix is currently supported. Messages sent to this prefix will be routed. Charge: 1
Or:

ERR: This prefix is not currently supported. Messages sent to this prefix will fail. Please contact support for assistance.

6.4 MMS push

When an MMS message is sent to a phone, the 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, from a URL specified in the SMS notification message. This command enables users to send an MMS notification message. Authentication is required for this API call.

MMS documentation (WAP-209-MMSEncapsulation-20020105-a.pdf, Version 05-Jan-2002) can be found at http://www.openmobilealliance.org/tech/affiliates/wap/wapindex.html.

Parameter	Description	Example	Default value	Restricted value	Required
mms_subject	Subject	My+message			yes
mms_class	Class	80		80 (Personal) 81 (Advertisement) 82 (Informational) 83 (Auto)	yes
mms_expire	How long before the MMS expires	3000		Time in seconds	yes
mms_from	From text	John			yes
mms_url	URL with the MMS content. The URL must be URL encoded.	http://www.mywebsite.com/example.mms			yes

Command:
http://api.clickatell.com/mms/ind_push.php?user=xxxx&api_id=xxxx&password=xxxx&to=xxxx&from=xxxx&mms_subject=xxxx&mms_class=xx&mms_expire=xxxx&mms_from=xxxx&mms_url=http://xxxx.xx/xx.mms

Response:
ID: xxxx To: xxxx
or:
ERR: Error number, error description

6.5   WAP push service indication

WAP Push Service Indication (SI) is a WAP address embedded within the header of a specially formatted SMS. This is displayed as an alert message to the user, and gives the user the option of connecting directly to a particular URL via the handsets WAP browser (if supported). This command enables you to send a WAP Push Service Indication.

Please note: Incorrect date formats for **si_created** and **si_expires** may lead to handsets discarding messages with delivery receipts.

WAP documentation (WAP-167-ServiceInd-20010731-a.pdf, Version 31-July-2001) can be found at http://www.openmobilealliance.org/tech/affiliates/wap/wapindex.html.

Parameter	Description	Example	Default value	Restricted values	Required
si_id	Unique ID for each message				Yes
si_url	The URL that is used to access the service or content. The URL must be URL encoded.	http://www.65mydomain.com?picture=6566			Yes
si_text	Notification text. Provides a means to specify additional information.	Here is your picture.			Yes
si_created	A date in UTC (Coordinated Universal Time) format. Used to specify the date and time associated with the creation or last modification of the content indicated by the URL, which may differ from the date and time when the SI was created.	2008-01-01T19:30:41Z			No
si_expires	Expiry date in UTC format. This allows you to specify a time after which the SI will automatically be deleted from the handset. If not specified it will never expire.	2008-12-12T19:30:40Z			No
si_action	A string specifying the action to be taken when the SI is received.		signal-medium,	signal-none, signal-low, signal-medium, signal-high, delete.	No

Command:
http://api.clickatell.com/mms/si_push?api_id=xxxx&user=xxxx&password=xxxx&to=xxxx&si_id=xxxx&si_url=xxxx&si_created=xxxx&si_expires=xxxx&si_action=xxxx&si_text=xxxx

Response:
ID: xxxx To: xxxx
or:
ERR: Error number, error description

6.6   Get message charge query

This command enables the user to query both the status and charge of a delivered message in a single API call. Authentication is required for this API call and will only work for messages less than 15 days old. Clickatell can also post the message charge to your application via means of a *Callback URL* (this is the preferred method). Detailed information can be found in the "Callback URL" section under "Message parameters".

Command:
http://api.clickatell.com/http/getmsgcharge?session_id=xxxx&apimsgid=xxxxx
or:
http://api.clickatell.com/http/getmsgcharge?api_id=xxxx&user=xxxx&password=xxxx&apimsgid=xxxxx

Response:
apiMsgId: xxxx charge: xxx status: xxx
or:
ERR: Error number, error description

6.7   Token (voucher) pay

This command allows you to spend a voucher that has been generated, within your client account, after logging in online. This is very useful for topping up sub-users accounts with credits. You would generate a session ID for the sub-user account, into which you wish to add the credits. You may also use the standard login details. The voucher number is currently a 16 digit numeric value.

Command:
http://api.clickatell.com/http/token_pay?session_id=xxx&token=xxxxxxxxxxxxxxxxxxx
or
http://api.clickatell.com/http/token_pay?api_id=xxx&user=xxx&password=xxxx&token=xxxxx

Response:
OK: 605 - If the transaction was successful
Or:
ERR: Error number, error description
ERR: Error number, error description

7. Batch messaging

This facility enables one to do high volume delivery and server-side message merging. It offers the end-user the ability to define all elements common to a batch, and then send only the parameters that change on a message by message basis.

One initially defines a batch using the *startbatch* command, which will return a unique batch ID. You then use either *senditem* or *quicksend* with the batch ID, depending on whether the message needs to be personalised. See SMS examples below.

Hi #field1#, your doctor's appointment is at #field2# tomorrow, could become:
Hi Fred, your doctor's appointment is at 10:30 tomorrow.
Hi Jane, your doctor's appointment is at 14:00 tomorrow.

7.1 Start batch

Once you have issued this command, you will be returned a batch ID that is to be used when sending multiple batch items. Included functionality also allows for message merging where you can substitute fields that you have defined in your template. The field names are called *field1* though to *fieldN*.

This command can take all the parameters of *sendmsg,* with the addition of a template, and the exception of both the destination address and the text fields. The template parameter must be URL encoded. It must be used before either the *senditem* or *quicksend* command.

Command:
http://api.clickatell.com/http_batch/startbatch?session_id=xxx&...............&template=Hi #field1#, your balance is #field2#.&from=Sender&deliv_ack=1

Response:
ID: batch id
or:
ERR: Error number, error description

7.2 Sending messages to existing batch

Note: The fields 1-N that you defined in the *startbatch* command are used to optionally personalise the message.

Command:
http://api.clickatell.com/http_batch/senditem?session_id=xxx&batch_id=xxx&to=123456789&field1=Joe&field2=$150........

Response:
ID: apimsgid
or:
ERR: Error number, error description

7.3 Quick send to batch

Where one has the requirement to send the same message to multiple recipients, you can use the *quicksend* command. This command offers low overhead and maximum throughput. It is essentially a reference to a predefined template and a string of destination addresses.

Command:
http://api.clickatell.com/http_batch/quicksend?session_id=xxx&batch_id=xxx&to=123456789,234567890,345678901,etc

Response:
ID: apimsgid To: xxxxx
or:
ERR: Error number, error description

Note: A response is returned for each destination address on a new line. The **newline character** (\n) is used to create the line break.

7.4 End batch

This command ends a batch and is not required (following a batch send). Batches will expire automatically after 24 hours.

Command:
http://api.clickatell.com/http_batch/endbatch?session_id=xxx&batch_id=xxx

Response:
OK
or:
ERR: Error number, error description

8. 8-BIT messaging

Through the HTTP interface, one is also able to send 8-bit messages. These are most often used for ringtones and logos, but one can also send vCards, vCalendar appointments and EMS messages. When sending 8-bit messages, you need to set the user data header (UDH) of the SMS as well as sending the data. If you are comfortable with the creation of your own UDH, we also enable you to set it directly using the **udh** parameter. To simplify the process, we have provided a number of pre-defined message types (see the **msg_type** parameter).

With the standard *text* parameter, line breaks are automatically inserted. The parameter **data,** is thus used for 8-bit messaging.

Example
api_id:1234
user:xxxxxxxxx
password:xxxxxxxxxxx
to:xxxxxxxxxxxxxxx
msg_type:SMS_NOKIA_RINGTONE
data:024A3A5585E195B198040042D9049741A69761781B6176156174288
data:B525D85E0A26C24C49A617628930BB125E055856049865885D200

9. Message examples

Here are some example URLs that demonstrate how to use the API. All values in these examples should be replaced by your own values.

9.1 Simple examples

sendmsg command including authentication and Sender ID:
http://api.clickatell.com/http/sendmsg?api_id=1&user=demo&password=demo&to=1234567890123&text=initial+test+message&from=ME

Initial authentication:
http://api.clickatell.com/http/auth?api_id=1&user=demo&password=demo

All further commands will use a session ID generated using auth command above:

sendmsg command:
http://api.clickatell.com/http/sendmsg?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&to=1234567890&from=ME&text=initial+test+message

Flash SMS:
http://api.clickatell.com/http/sendmsg?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&to=1234567890&from=ME&msg_type=SMS_FLASH&text=flash+test+message

sendmsg with delivery acknowledgment and callback request:
http://api.clickatell.com/http/sendmsg?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&to=1234567890&from=ME&callback=3&deliv_ack=1&text=callback+and+deliveryack

Account balance:
http://api.clickatell.com/http/getbalance?session_id=e74dee1bbed22ee3a39f9aeab606ccf9

Query message status:
http://api.clickatell.com/http/querymsg?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&apimsgid=4889e40291643afeb5a7c4cce7811abb

Keeping your session alive:
http://api.clickatell.com/http/ping?session_id=e74dee1bbed22ee3a39f9aeab606ccf9

9.2 Batch SMS examples

9.2.1 Sending a personalised messages to multiple recipients

http://api.clickatell.com/http_batch/startbatch?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&template=Hi+%23field1%23+this+is+a+personalised+message

9.2.2 Sending two personalised messages

http://api.clickatell.com/http_batch/senditem?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&batch_id=f677d2fbb858a79aad0556dc71dd4383&to=1234567890&field1=David
http://api.clickatell.com/http_batch/senditem?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&batch_id=f677d2fbb858a79aad0556dc71dd4383&to=2345678901&field1=John

9.2.3 Sending multiple SMS using batches

http://api.clickatell.com/http_batch/quicksend?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&batch_id
=f677d2fbb858a79aad0556dc71dd4383&to=1234567890,2345678901,3456789012,4567890123,5678901
234

9.3 8-bit SMS examples

Note: You cannot set an alphanumeric Sender ID (**from** parameter) when sending 8-bit messages.

9.3.1 Sending a ringtone

Sending a ringtone using the **msg_type** parameter:
http://api.clickatell.com/http/sendmsg?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&to=1234567890
&msg_type=SMS_NOKIA_RINGTONE&text=024A3A5585E195B198040042D9049741A69761781B61761
56174288B525D85E0A26C24C49A617628930BB125E055856049865885D200

Sending same ringtone with the **udh** parameter:
http://api.clickatell.com/http/sendmsg?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&to=1234567890
&udh=06050415810000&data=024A3A5585E195B198040042D9049741A69761781B6176156174288B52
5D85E0A26C24C49A617628930BB125E055856049865885D200

9.3.2 Sending an operator logo

http://api.clickatell.com/http/sendmsg?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&to=1234567890
&msg_type=SMS_NOKIA_OLOGO&text=00480e010FC0000000000000003FF000000000000000
70380F9B006000001B601818DB006000C01BCF0C3058006000C01BDF8C301B3E66F9EF9BDF8C301B
626C8CD8DBDF8C301B60787CDFDBCF0C305B6078CCD81B601818DB626C8CD8DB70380F9B3E66F
CEF9B3FF0000000000000000FC0000000000000000000000000000000000

9.3.3 Removing an operator logo

http://api.clickatell.com/http/sendmsg?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&to=1234567890
&msg_type=SMS_NOKIA_CLEAN&text=00

9.3.4 Sending a VCARD

http://api.clickatell.com/http/sendmsg?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&to=1234567890
&msg_type=SMS_NOKIA_VCARD&text=BEGIN%3AVCARD%0D%0AVERSION%3A2.1%0D%0AN%3ABl
oggs%3BJoe%0D%0ATEL%3BPREF%3A%2B1234567890%0D%0AEND%3AVCARD%0D%0A

9.3.5 Sending a VCAL

http://api.clickatell.com/http/sendmsg?session_id=e74dee1bbed22ee3a39f9aeab606ccf9&to=1234567890
&msg_type=SMS_NOKIA_VCAL&text=BEGIN%3AVCALENDAR%0D%0AVERSION%3A1.0%0D%0ABEG
IN%3AVTODO%0D%0ACATEGORIES%3AMISCELLANEOUS%0D%0ASUMMARY%3AMeet+buyers+at+
Mario's%0D%0ADTSTART%3A20030301T133000%0D%0AEND%3AVTODO%0D%0AEND%3AVCALEN
DAR%0D%0A

Appendix A: Error codes

The following list of error messages are generated by the Clickatell gateway during a validation phase before we accept the message. These error messages are sent back to your application. There will be no message charge if these errors are generated when sending a message. Data regarding messages that do not pass initial validation will not be included in your Clickatell Central reports.

Number	Description	Detail
001	Authentication failed	
002	Unknown username or password	
003	Session ID expired	
004	Account frozen	
005	Missing session ID	
007	IP Lockdown violation	You have locked down the API instance to a specific IP address and then sent from an IP address different to the one you set.
101	Invalid or missing parameters	
102	Invalid user data header	
103	Unknown API message ID	
104	Unknown client message ID	
105	Invalid destination address	
106	Invalid source address	
107	Empty message	
108	Invalid or missing API ID	
109	Missing message ID	This can be either a client message ID or API message ID. For example when using the stop message command.
110	Error with email message	
111	Invalid protocol	
112	Invalid message type	
113	Maximum message parts exceeded	The text message component of the message is greater than the permitted 160 characters (70 Unicode characters). Select **concat** equal to 1,2,3-N to overcome this by splitting the message across multiple messages.

114	Cannot route message	This implies that the gateway is not currently routing messages to this network prefix. Please email support@clickatell.com with the mobile number in question.
115	Message expired	
116	Invalid Unicode data	
120	Invalid delivery time	
121	Destination mobile number blocked	This number is not allowed to receive messages from us and has been put on our block list.
122	Destination mobile opted out	
123	Invalid Sender ID	A sender ID needs to be registered and approved before it can be successfully used in message sending.
128	Number delisted	This error may be returned when a number has been delisted.
201	Invalid batch ID	
202	No batch template	
301	No credit left	
302	Max allowed credit	

10. Appendix B: Message statuses

These are message statuses that are generated after the Clickatell gateway has accepted the message for delivery. Data regarding messages passing initial validation and accepted for delivery will be included in your Clickatell Central reports.

Number	Hex	Description	Detail
001	0x001	Message unknown	The message ID is incorrect or reporting is delayed.
002	0x002	Message queued	The message could not be delivered and has been queued for attempted redelivery.
003	0x003	Delivered to gateway	Delivered to the upstream gateway or network (delivered to the recipient).
004	0x004	Received by recipient	Confirmation of receipt on the handset of the recipient.
005	0x005	Error with message	There was an error with the message, probably caused by the content of the message itself.
006	0x006	User cancelled message delivery	The message was terminated by a user (stop message command) or by our staff.
007	0x007	Error delivering message	An error occurred delivering the message to the handset.
008	0x008	OK	Message received by gateway.
009	0x009	Routing error	The routing gateway or network has had an error routing the message.
010	0x00A	Message expired	Message has expired before we were able to deliver it to the upstream gateway. No charge applies.
011	0x00B	Message queued for later delivery	Message has been queued at the gateway for delivery at a later time (delayed delivery).
012	0x00C	Out of credit	The message cannot be delivered due to a lack of funds in your account. Please re-purchase credits.

11. Terminology

- **Mobile originated (MO):** A message sent (originating) from a mobile handset to an application via Clickatell.
- **Mobile terminated (MT):** A message sent from an application to (terminating on) a mobile handset via Clickatell.
- **Premium rated message (MO):** A mobile user is charged a premium for the message that they send to a particular short or long code. This service is not available in all regions; please contact an Account Manager for more information.
- **Revenue share:** This refers to the portion of the premium charge associated with a premium rated message, which is passed on to the content provider.
- **Content provider:** This is the Clickatell customer who is offering one or more services that are usually premium rated SMS system.
- **Customer:** A registered Clickatell customer utilising the Clickatell API for message delivery and receipt.
- **Sender ID:** The "from" address that appears on the user's handset. This is also known as the message originator or source address. A Sender ID must be registered within your account and approved by us before it may be used.
- **Destination address**: The mobile number/MSISDN of the handset to which the message must be delivered. The number should be in international number format, e.g. country code + local mobile number, excluding the leading zero (0).
- **Source address:** See 'Sender ID' above.
- **Short code:** A short number which is common across all the operators for a specific region.
- **Subscriber:** The mobile network subscriber who owns the mobile number (MSISDN) which will send or receive SMSs, or be billed for premium rated services.
- **Upstream gateway:** A network operator, third party or our own short message service centre (SMSC).

12. Contact details

Phone: +27 21 910 7700
Fax: +27 21 910 7701
Website: www.clickatell.com
Help URL: http://support.clickatell.com/index.php
Support: support@clickatell.com
Info: info@clickatell.com
Sales: sales@clickatell.com

--- 0 ---