



## SMS-services and billing

# sms:transit

*informational leaflet*





# **sms:transit**

## **Table of content**

- 1. Service description**
- 2. Functionality**
- 3. Installation and configuration**
- 4. Technical description**
  1. Dynamic rate scale
  2. Static rate scale
- 5. Implementation details**
  1. Message statuses
- 6. Samples and source code**
- 7. Contacts**



# sms:transit

## Service description

**sms:transit** is the most powerful and at the same time the simplest solution available at the sms services' market today. Being a fully-functional message aggregator, our service leaves you to decide on each and every aspect of your future system; its possibilities in this case are limited only to your imagination.

## How does it work

Using an **sms:transit** as a starting point, you're given all the resources necessary to develop your very own sms-service of any kind, be it an access restriction system, a chat or a mobile content distribution infrastructure. You decide by yourself how to implement incoming messages processing and what to send back to user.

## Installation and setup

In order to start using **sms:transit**, you need a personal approval of the [Sales and partnership](#) division. Technical details on the service's implementation and fine-tuning follow.



## Technical info

**sms:transit** working principle is simple: the end user sends a message, the aggregator transfers it using the HTTP request directly to your script (Result URL), the string returned is being sent back to the user.

If your script isn't available at the time aggregator tries to access it (the HTTP request status is *not* 200 OK), all the relative data is being sent to you by e-mail for the manual processing, and the user gets back a default reply (which is defined in service's settings).

In addition to the primary processing script, you should specify message status confirmation script (Status URL). Its usage is *vital* for countries which make use of MT-style billing (currently only Australia, Belgium, Ireland and the United Kingdom); it is advisable to use it everywhere else, too.

With **sms:transit**, you're able to accept messages of all prices available for a given country simultaneously - and that's a conceptual difference of an **sms:transit** from the rest of our services. You can either use this feature, or you can set the fixed price in the settings dialogue; both options are described in detail here.



## Dynamic rate scale

If you didn't explicitly set the message price in the service's settings (or set it to zero, which does the same thing), you're automatically accepting all prices available for each country, as stated above. With such settings, your transaction result processor script (Result URL) must take in account the price of the received message.

In all supported countries (except for the United Kingdom) there is a separate number for each given price; actually, the number *declares* the message price. In order for the message to be associated with your service, it must be formed like

**prefix transit id** message text

and sent to number available in that country. Note that the task of providing the user with all necessary information on how to send the message is now placed fully on your shoulders; about the actual numbers and prefixes see below.



## Static rate scale

If you have set a different than 0 message price in the service's settings, you're using the so-called static rate scale. The rate scale of your **sms:key** is always available in XML at the address like

```
http://service.smscoin.com/xml2/transit/transit id/
```

The XML file mentioned includes also the data on messages' target numbers (the `number` field) and prefixes needed by the local cellular operators in order to transfer the message to our system (the `prefix` field); in different countries there are, respectively, different numbers and prefixes. In order for the message to be associated with your service, it must be formed like

```
prefix transit id message text
```

and sent to number available in that country, according to the fee you're planning to collect. Note that the task of providing the user with all necessary information on how to send the message is now placed fully on your shoulders.

If some country isn't present in the rate scale mentioned, it means that the country just lacks the fee you set in the service's properties; try lowering the fee. All messages sent from such country will be simply ignored.



## Implementation details

Message processing script (Result URL) accepts the following parameters:

Parameter	Type	Description
<code>country</code>	char(2)	Two-letter country code.
<code>shortcode</code>	int	Number that accepted the message.
<code>provider</code>	char(16)	Cellular operator used (if relevant).
<code>prefix</code>	char(16 )	Message prefix.
<code>cost_local</code>	float	Message cost in local currency.
<code>cost_usd</code>	float	Message cost in USD.
<code>phone</code>	char(32)	The phone number of an end user, including the country code; there are certain operators that send some unique identifier instead.
<code>msgid</code>	char(32)	Unique message id.
<code>sid</code>	int	Service id.
<code>content</code>	char(128)	Message text.
<code>sign</code>	char(32)	MD5 hash of a string consisting of <code>secret_code</code> , <code>country</code> , <code>shortcode</code> , <code>provider</code> , <code>prefix</code> , <code>cost_local</code> , <code>cost_usd</code> , <code>phone</code> , <code>msgid</code> , <code>sid</code> and <code>content</code> (in that order) splitted by double colon ("::"), where <code>secret_code</code> is your <b>sms:transit</b> secret code.
<code>billing</code>	char(2)	billing type, MO or MT. if Mt, additional request will be sent to <code>Status_url</code> (see below)
<code>mcc</code>	int(3)	Mobile Country Code
<code>mnc</code>	int(3)	Mobile Network Code
<code>profit</code>	float	Your percentage of SMS price in USD. (parameter <code>cost_usd</code> ).



## SMS-services and billing

It's important that your script **must** reply with HTTP status 200 OK and return string that will be sent back to user. Replies in `title@@@link` format (e.g. `example@@@http://example.com`) are being automatically converted to WAP-links when sent (works for Russia, Ukraine, Kazakhstan, Israel, Lithuania). The `@@@` string is reserved and should not be used otherwise.

Status confirmation script (Status URL) accepts the following parameters:

Parameter	Type	Description
<code>msgid</code>	char(32)	Unique message id.
<code>phone</code>	char(32)	The phone number of an end user, including the country code; there are certain operators that send some unique identifier instead.
<code>status</code>	char(16)	Message status (see below).
<code>sign</code>	char(32)	MD5 hash of a string consisting of <code>secret_code</code> , <code>msgid</code> , <code>phone</code> and <code>status</code> (in that order), splitted by double colon (" <code>::</code> "), where <code>secret_code</code> is your <b>sms:transit</b> secret code.



## Message statuses

Working with MT-billing-enabled countries (as said above, then you receive billing=MT to Result\_url), a message cannot be considered processed while its status is unknown. Status may be one of the following:

- **delivered** - the message was delivered, with fee collected;
- **rejected** - user has rejected the payment;
- **failed** - the message could not be delivered;
- **fraud** - the payment for this message has been cancelled; this can happen as a result of fraudulent actions of a subscriber or in an event of TOS violation by the service owner. This status can be possibly applied to any message even *after* the **delivered** notification.

In the rest of the world the message is considered pre-paid by default, and the status confirmation script is only being notified of the messages considered **fraud** (see above).

In the State of Israel, in addition to **fraud**, there are two more post-statuses which can be sent - **unconfirmed** (an end-user has cancelled the payment) and **time-out** (server-side time-out has occurred while delivering the message).



## SMS-services and billing

### **Samples and source code**

Download source code in [PHP](#), [Perl](#), [ASP.NET](#).

Please note: code samples are purely illustrative, and thereby their use in production environments without substantial revision is highly unadvised.



## SMS-services and billing

### Contacts

#### Tech support:

**ICQ:** 429853174  
**E-mail:** [support@smscoin.com](mailto:support@smscoin.com)

#### Sales and partnership:

**ICQ:** 570342641  
**E-mail:** [sales@smscoin.com](mailto:sales@smscoin.com)

#### Phones:

in Moscow:	+7 (499) 5700158
in St. Petersburg:	+7 (812) 3090048
in Israel:	(+972) 49111010
in Lithuania:	(+370) 52111429
in Deutschland:	+49 (180) 35350033090

**Fax:** (+972) 48408497

**Office:** Israel, Haifa  
Yochanan HaSandlar St 11

#### Correspondence address:

Agregator Ltd.  
P.O. Box 121  
Tirat Carmel, 39100  
Israel

---

<http://smscoin.net>