



APPLICATION PROGRAMMING INTERFACE

Version	Date	Comment
1.0	19/12/2014	English version
2.0	15/09/2015	LAM Update by M.DIOUF
2.1	23/01/2020	LAM Update by M.SOW

SUMMARY

Scope.....	3
Account parameters.....	3
2.1 Prerequisite.....	3
2.2 Variables used in the documentation.....	3
2.3 Testing your account with a unit sent with web interface.....	4
Sendig a SMS using XML.....	4
3.1 Use XML Request.....	4
3.1.1 First Sample.....	5
3.1.2 How to send an XML via http.....	5
3.1.3 How to send “like XML” via http.....	5
3.2 XML details.....	7
3.2.1 How to construct .XML – Xml sample.....	7
3.2.2 XML structure.....	8
3.2.3 Element <text>.....	10
3.2.4 Element <to>.....	10
3.2.5 Element <param>.....	10
3.3 Error when integrating XML.....	11
Acknowledgement of receipt.....	11
4.1 Real time received of acknowledgement.....	11
4.2 Global Request Status on a push (highpush_stat.php).....	13
Campaign cancellation.....	15
Credit verification via API.....	17
Web interface of the Api.....	17
7.1 Login.....	17
7.2 Push List.....	18
7.2.1 Main screen and filters.....	18
7.2.2 Status day per day.....	19
7.2.3 Graph.....	20
7.2.4 Export Details.....	20
Annexe.....	22
8.1 Using Variables.....	22
8.2 Error XML.....	22
8.3 Status Codes.....	23
8.4 Different id.....	23

Scope

LAMPUSH is a solution to send SMS, using a computer interface. LAMPUSH provides two ways of working:

- A Web interface: the user connects via a browser
- An API: user's computer exchanges information directly with LAMPUSH platform

This document describes the second way, and presents how to use this API.

Account parameters

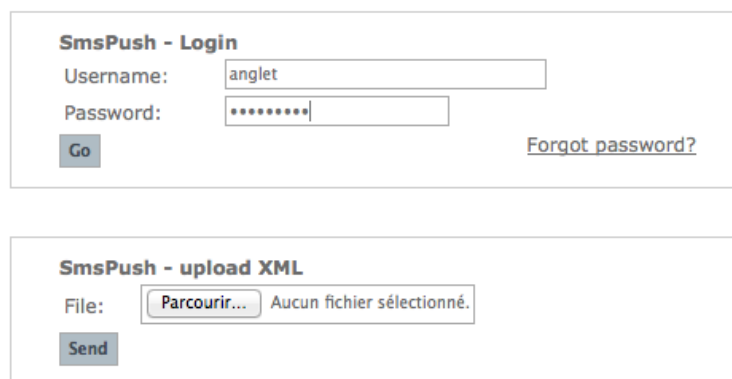
2.1 Prerequisite

To use this API, you need a valid account.

When creating your account, LAfricaMobile should have given you:

- A login (named also accountid)
- A password

This pair login/password is used as parameters in API and can also be used to connect to LAMPUSH platform via your web browser.



The image shows two screenshots of a web interface. The top screenshot is titled "SmsPush - Login" and contains a form with two input fields: "Username:" with the value "anglet" and "Password:" with a masked password "*****". Below the password field is a "Go" button and a link for "Forgot password?". The bottom screenshot is titled "SmsPush - upload XML" and contains a "File:" field with a "Parcourir..." button and the text "Aucun fichier sélectionné.". Below this field is a "Send" button.

Figure 1: Connection to the platform with Web browser

2.2 Variables used in the documentation

In this document, we use values below. Real values will be given to you by our technical team or by your own configuration (for parameters depending on your side, [ret_url](#) for example) Some of them might be not used depending of your needs.

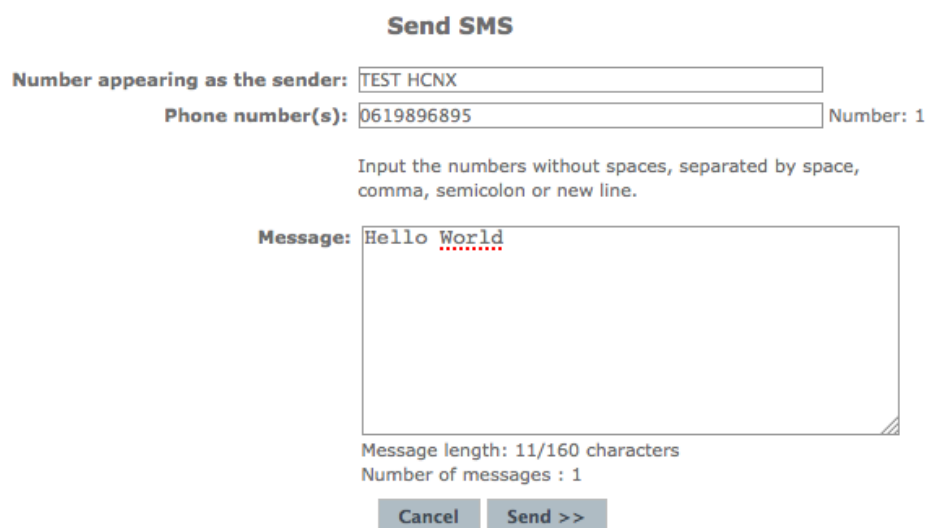
Document samples values:

- accountid=[fred](#)
- password=[xpasswordx](#)
- LAMPUSH Server url : <https://lampush-tls.lafricamobile.com>
- Phone Test Number:[+221771611010](#)

You have to replace these values by your owns during your testing.

2.3 Testing your account with a unit sent with web interface

Once connected, you can choose « Sending Unit » and prepare your first SMS



Send SMS

Number appearing as the sender:

Phone number(s): Number: 1

Input the numbers without spaces, separated by space, comma, semicolon or new line.

Message:

Message length: 11/160 characters
Number of messages : 1

Figure 1: Send a SMS using the Web Interface



Sendig a SMS using XML

3.1 Use XML Request

After sending a SMS manually as shown previously, we suppose that, if you are reading this document about API, it's because you want to send SMS automatically, without using a Web browser

One way to ask LAMPUSH to send SMS is to supply it an XML request.

This request can be given either as a file or transmitted as an HTTP(s) request in POST or GET format.

3.1.1 First Sample

```
<?xml version="1.0" encoding="iso-8859-1"?>
<push
accountid="fred"
password="xpasswordx" >
<message>
<text>Helllo world !</text>
<to>00221771611010</to>

</message>
</push>
```

3.1.2 How to send an XML via http

You can send this XML via an http request using POST or GET format. The content of the xml has to be in a parameter named xml.

Here is a sample with a GET
`https://lampush-tls.lafricamobile.com/api?xml=<push accountid="fred" password="xpasswordx"><message><text>Helllo world xml! </text><to>00221771611010</to></message></push>`

It's not recommended to use this way to send **ONE** SMS, it's just a sample to explain how API works. If you want to send unitary SMS using HTTP request, see OneShot sending.

So, when you enter this address in your web browser you can see an id like this:

[1866740934](#)

It's a push_id and we are going to explain further how to use it.

If an error occurs you can see a message like this:

```
error: 1 (XML_PARSING_ERROR)
error_code: 76
error_string:
```

line_number: 1
column_number: 144
byte_index: 119

3.1.3 How to send “like XML” via http

Instead of having all parameters encapsulated in a XML parameter, you can directly POST or GET parameters individually:

These parameters are:

Name	Mandator	Sample value	Comment
accountid	Y	fred	Value given by LAM
password	Y	xpasswordx	Value given by LAM
text	Y	Hello%20World	
datacoding	N		In further version special encoding may be supported, but you can use either iso-8859-1 (latin1) or UTF8
to	Y	00221771611010	Phone number

So, you can call this directly in your browser:

<https://lampush-tls.lafricamobile.com/api?accountid=fred&password=xpassword&text=Hello,%20bel%20%e9t%e9&to=00221771611010>

Here is a sample in PHP, with the script PHP in ISO format (:[setfileencoding?underlinux/vim](#) to see current encoding of your script, :[set fileencoding=latin1](#) to set latin)

```
<?php
$accountid='fred';
$password='xpasswordx';
$urlapi='https://lampush-tls.lafricamobile.com/api';
# (the usage may be limited to allowed ippaddress, you should have given #your own
server address to be authorized to use this url)

$to='00221771611010'; # Sample number
$sender='LAM_ISO';
$text='Hello, belété';

$text=urlencode($text);
$full_url_called=$urlapi.'?'. "accountid=$accountid&password=$password"
."&text=$text"
."&to=$to"
."&sender=$sender" ;
print "$full_url_called\n";
$result=file_get_contents($full_url_called);
print "\nresult=";
print_r($result);
print "\n----\n";
```

Like with XML format, the push_id is printed back in response.

Here is what is displayed when previous script is called:

```
https://lampush-tls.lafricamobile.com/api?
accountid=fred&password=xpasswordx&text=Hello%2C+bel+%E9t
%E9&to=0619896895&sender=LAM_ISO
```

```
result=1871005578
```

Note that you have %E9 to represent “é” in this case:character was coded in ISO-8859-1

result (1871005578 in our sample) is the push_id.

Here is a sample in PHP, with the script PHP in UTF8 format ((:set fileencoding?underlinux/vim to see current encoding of your script, :set fileencoding=utf8 to set UTF8)

```
<?php
$accountid='fred';
$password='xpasswordx';
$urlapi='https://lampush-tls.lafricamobile.com/api';
# (the usage may be limited to allowed ipaddress, you should have given #your own
server address to be authorized to use this url)
```

```
$to='00221771611010'; # Sample number
$sender='LAM_UTF8';
$text='Hello, belété';
```

```
$text=urlencode($text);
$full_url_called=$urlapi.'?'. "accountid=$accountid&password=$password"
."&text=$text"
."&to=$to"
."&sender=$sender" ;
print "$full_url_called\n";
$result=file_get_contents($full_url_called);
print "\nresult=";
print_r($result);
print "\n---\n";
```

When called, it displayed:

```
https://lampush-tls.lafricamobile.com/api?
accountid=fred&password=xpasswordx&text=Hello%2C+bel+%C3%A9t
%C3%A9&to=0619896895&sender=LAM_UF8
```

```
result=1871016995
```

Note that you have %C3%A9 to represent “é” in this case:character was coded in UTF8

3.2 XML details

3.2.1 How to construct .XML - Xml sample

Sample shown above, is very simple. XML has been designed to drive massive sending with numerous parameters.

Here is a more complete sample:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<push
accountid="fred"
password="xpasswordx
name="PushName"
userdata="User Data Multiple Sent"
ret_id="Push_1"
sender="API_LAMPUSH">
<message>
<text>Helllo world Multiple</text>
<to>00221771611010</to>
<to>00221771611011</to>
</message>
</push>
```

3.2.2 XML structure

XML contents one item <push> followed by one or several item <message>

Some attributes can be set either in <push> element or <message> element.

If defined in both elements, attributes in <message> element overrides attributes set in <push> element.

3.2.2.1 Element<push>

Element <push> contains one or several element <message>.

Attributes of <push>

Name	Mandatory	Sample value	Comment
accountid	Y	fred	Value given by LAM
password	Y	xpasswordx	Value given by LAM
start_date	N	20014-12-12 default=today	If set, the SMS won't be sent before this date.
start_time	N	13:00 default='00 :00'	In particular case, when start_date is set but in the past, then start_time is used with the date of the current day. Be careful:if start_time is NOT

			set and start date is set to a value > today it's means start_date ,midnight. Time is local time (Paris: GMT+2)
stop_time	N		If stop_time is set, no SMS is transmit to the sending gateway after this time, and remaining SMS are sent the following day, at start_time. Be aware that SMS transmitted to the gateway before this time may be sent to the operator after this time.
userdata	N	User Data Multiple Sent	It's a free field that is written back in status files. It's stored in push.user_data
sender	N	API_LAMSMS	If set, it replaces number that appears as 'Sender' in the SMS. Value must NOT begin with a digit and has 11 characters maximum (1FOE is forbidden, FOE12345678 is allowed).
ret_id	N	Push_1	if set, it is stored in push.psh_ext_id. <push>.ret_id is associated with the campaign (push). It's different from <to>.ret_id that is a id associated with a unitary SMS defined by client.
ret_url	N	http://lampush-tls.lafriamobile.com//peps_sample/RetrieveStatus.php	If set, when status changes a call to this url is done. Specifications of parameters defined in chapter 4.1.3
priority	N	2	Allow to increase priority of the push. Do not increase this value for marketing push

3.2.2.2 Element<message>

<message> contains one element <text> , one or several element <to>

Attributes of <message>

Name	Mandatory	Sample value	Comment
start_date	N	20014-12-12 default=today	If set, the SMS won't be sent before this date.

start_time	N	13:00 default='00 :00'	In particular case, when start_date is set but in the past, then start_time is used with the date of the current day. Be careful if start_time is NOT set and start date is set to a value > today it's means start_date ,midnight. Time is local time (Paris: GMT+2)
stop_time	N		If stop_time, is set, no SMS is transmit to the sending gateway after this time, and remaining SMS are sent the following day, at start_time. Be aware that SMS transmitted to the gateway before this time may be sent to the operator after this time.
userdata	N	User Data Multiple Sent	It's a free reference that is written back in status files.
sender	N	API_LAMPUSH	If set, it replaces number that appears as 'Sender' in the SMS. Value must have at less one alphabetical character and has 11 characters maximum ("1664" is not allowed otherwise "bier1664" is allowed).
ref_model	N	Y default='N'	If set to 'Y', means that <text> is not the text of the message, but the name of a model that is created via Web interface. See specific sample.
priority	N	2	Allow to increase priority of the push. Do not increase this value for marketing push
userdata	N	myClientName	Allow to give extra information that can be retrieve through reports

3.2.3 Element <text>

The <text> element is used for the text of the message.

<text> has no attributes, just a value.

In general case it's simply the text of the SMS.

If ref_model is set to 'Y' it's the name of a model (see specific sample)

Values of text :

The values of text is the text of the message

`<text>Hello world</text>`

or with a variable

`<text>Hello world, %FIRST_NAME%</text>`

or a model name

`<text>Bonjour</text>`

As shown above, `<text>` can contains some variable strings. In fact you can use any string as variable name. However, we encourage you to use `%VAR_NAME%` as string name for variable named 'var_name'. See chapter Element`<param>` below.

3.2.4 Element `<to>`

`<to>` is used to give information about the delivery user. It is part of an `<message>` element.

`<to>` may contains some `<param>` element, used to personalize text message.

The value is a phone number, you can use international format (+221771611010)

For international sending , please confirm with your account manager if the country is allowed in your account.

Attributes of `<to>`

Name	Mandato ry	Sample value	Comment
ret_id	N	231467	You can use this field to associate an (Unique) Id to each SMS sent. This ret_id is associated with a sms (different from <code><push>.ret_id</code>) and stored in database and returned back in status information.

3.2.5 Element `<param>`

`<param>` is used to give parameters to be used in the text message.

`<param>` are part of an `<to>` element.

`<param>` must have two attributes, "var" for variable name and "value".

Attributes of `<param>`

Name	Mandator y	Sample value	Comment
var	Y	%NAME%	We suggest to use % to surround the variable name in capital letters

value	Y	Peps	
-------	---	------	--

3.3 Error when integrating XML

If XML is incorrect (in .xml file or in a http request), an error message is generated.

It is displayed in response (http request).

Here is a sample of an error message:

error: 1 (XML_PARSING_ERROR)

error_code: 76

error_string:

line_number: 10

column_number: 7

byte_index: 205

Acknowledgement of receipt

There are two ways to follow SMS routing :

- Your server is informed by the gateway when status change
- You asked our server SMS status.

First way is much better, because it's real time, and does not load platform with uselessly calls.

The second way must only be used occasionally.

4.1 Real time received of acknowledgement

In this case our server call an url, when it detects status has changed. This url have to be supplied in the XML, in the parameter named 'ret_url' in the <push> section. (in no url is supplied, no call is done).

This ret url is called with these parameters in HTTP GET

- push_id (transmitted back by our server)
- ret_id (set in the initial call)
- to (phone number of the recipient)
- status (an status or an error_code in case of an error)
- text (text of the SMS sent)

Value for status may be :

SENT=>4 (depending of the configuration)
RECEIVED=>6
ERROR_NPAI=>11
ERROR_EXPIRED=>12
ERROR_INVOP=>13
ERROR_NETWORK=>14
ERROR_CREDIT=>15
ERROR_UNKNOWN=>16

ERROR=>2 (generic error)

For example with this XML

```
<?xml version="1.0" encoding="iso-8859-1"?>  
<push  
  accountid="fred"  
  password="xpasswordx"  
  name="PushName"  
  userdata="User Data Multiple Sent"  
  ret_id="12345"  
  sender="API_6"  
  ret_url="https://lampush-tls.lafricamobile.com//peps_sample/RetrieveStatus.php"  
>  
<message>  
<text>Helllo world : CheckStatus</text>  
<to>00221771611010</to>  
<to>00221761611011</to>  
</message>  
</push>
```

with script RetrieveStatus.php which just log parameters value (see sample in Annexe) we will obtain this

RetrieveStatus:

Array

```
(  
  [push_id] => 1871011352  
  [ret_id] => 12345  
  [to] => +221771611010  
  [status] => 11  
  [tag] => User Data Multiple Sent
```

```

    [text] =>Hello world : CheckStatus
)
RetrieveStatus:
Array
(
    [push_id] => 1871011352
    [ret_id] => 12345
    [to] => +221771611010
    [status] => 6
    [tag] => User Data Multiple Sent
    [text] =>Hello world : CheckStatus
)

```

That match to the following GET calls :

```

/RetrieveStatus.php?push_id=1871011213&ret_id=12345&to=
%2B221771611010&status=6&tag=User+Data++Multiple+Sent&text=Hello+world+
%3A+CheckStatus

```

```

/RetrieveStatus.php?push_id=1871011213&ret_id=12345&to=
%2B221771611010&status=11&tag=User+Data++Multiple+Sent&text=Hello+world+
%3A+CheckStatus

```

Note that, in this case the ret_id is in fact a push.ret_id

4.2 Global Request Status on a push (highpush_stat.php)

If you don't want to have your own database with status but want punctually to know the status of a set of SMS you can call our server via an HTTP request.

The script to call is named [highpush_stat.php](#)

To avoid heavy load of platform DO NOT call this script

Name	Mandato ry	Sample value	Comment
accountid	Y	fred	Value given by LAM Nota: Even it's this accountid is usually the same that your LAMSMS accountid, in fact it's a SMSPusher account id.
password	Y	xpasswordx	Value given by LAM

			Nota: Even it's this password is usually the same that your LAMPUSH password, in fact it's a SMSPusher password.
push_id	Y if ret_id not defined	1871011352	Internal value of the push_id, returned when XML is provided via HTTP request.
ret_id	Y if push_id not defined	12345	Value you provide in XML, associated to the push not to the message

To request, you have to give your accountid, your password and either the push_id or the ret_id..

In our sample with XML :

```
<?xml version="1.0" encoding="iso-8859-1"?>
<push
accountid="fred"
password="xpasswordx"
name="PushName"
userdata="User Data Multiple Sent"
ret_id="12345"
sender="API_6"
ret_url="https://lampush-tls.lafricamobile.com//peps_sample/RetrieveStatus.php" >
<message>
<text>Hello world : CheckStatus</text>
<to>00221771611010</to>
<to>00221761611011</to>
</message>
</push>
```

We can request either with

https://lampush-tls.lafricamobile.com/highpush_stat.php?accountid=fred&password=xpasswordx&push_id=1871011352

or with

https://lampush-tls.lafricamobile.com/highpush_stat.php?accountid=fred&password=xpasswordx&ret_id=12345

Nota : In case there are several pushes with the same ret_id, the latest is taken into account.

The http request returns a CSV file (Content-Disposition: attachment; filename=export.csv
Content-Type: text/csv)

lines in black italic are not sent

Send date	Total	Net_error	Waiting	NPAI	expired	Other error	received	
08/01/15 11:38	2			1			1	
Send date	Phone Number	Status	Status Update	push_id		text	Error Status	userdata
08/01/15 11:38	00221771611 010	RECEIVED	08/01/15 12:22	187101135 2		Hello world : CheckStat us		User Data Multiple Sent
08/01/15 11:38	00221761611 011	ERROR	08/01/15 12:22	187101135 2		Hello world : CheckStat us	INVALID	User Data Multiple Sent

Firs line is a summary for the push Sent Date

- Total Number of SMS
- Total net_error (ERROR_NETWORK => 14)
- Total waiting (no Error or Received answer)
- Total npai (ERROR_NPAI=>11)
- Total expired (ERROR_EXPIRED=>12)
- Total other error
- Total Received

There is a another line per SMS.

- PushDate
- Phone Number (to)
- Status (ERROR/RECEIVED/SENT)
- Last status update (update of the status to RECEIVED may happen far after the receipt of the SMS)
- Empty
- Text of the message
- Error Status (if status = Error)
- userdata (field userdata of the xml)



Campaign cancellation

In case the push campaign is planned in the future and is identified by a ret_id you can cancel it.

This can be done with a call to the api with a parameter action set to cancel_campaign.

In our sample the url is : <https://lampush-tls.lafricamobile.com/api>

Parameters

Name	Mandatory	Sample value	Comment
accountid	Y	fred	Value given by LAM Nota: Even it's this accountid is usually the same that your LAM accountid
password	Y	xpasswordx	Value given by LAM Nota: Even it's this password is usually the same that your LAMSMS password, in fact it's a SMSPusher password.
action	Y	cancel_campaign	
ext_id	Y	12345	ret_id you provide in XML, associated to the push not to message

Here is a full sample :

https://lampush-tls.lafricamobile.com/api?accountid=fred&password=xpasswordx&ext_id=12345&action=cancel_campaign

In response the number of campaign effectively canceled is displayed:

0 CAMPAIGNS HAS BEEN STOPPED

In this case, as the campaign has already been sent, no campaign is effectively cancelled.

In case there are several campaign with the same ret_id that are not already sent, all the campaigns are cancelled in one call.

Credit verification via API

You can verify credit of your account, before transmitting your Push. For this, you can call main url with **credits** request. Parameters to be supplied are accountid and password.

For our sample with <https://lampush-tls.lafricamobile.com> as root url :
<https://lampush-tls.lafricamobile.com/credits?accountid=fred&password=xpasswordx>

Response is a XML that indicates, for each route type allowed for the account, credits.

Credits are either global or monthly.

-1 indicates infinite value.

0 means “out of stock”

Here is the response for the account fred, which have only one infinite route “Gold”

```
<credits>
<route>
<type>gold</type>
<credits>-1</credits>
<credits_month>-1</credits_month>
</route>
</credits>
```

Web interface of the Api

7.1 Login

Web interface is useful to see quickly status of your SMS, statistics when you use API.

You need your login and password and use the root url, of the LAMSMSserver <https://lampush-tls.lafricamobile.com> in our sample.

7.2 Push List

7.2.1 Main screen and filters

Push list

Subject: userdata: ret_id: [Filter](#)
 Telephone: Route: [See Graph](#)
 Start date: 01/01/2015 End date: 08/01/2015 [Export messages in current view](#)

Select a date to view it's stats.

Date	Route	Messages	Export
08/01/2015	Gold	Total MSISDN sent	22
		Blacklisted	0
		Awaiting sending:	0
		SMS received:	20
07/01/2015	Gold	Total MSISDN sent	38
		Blacklisted	0
		Awaiting sending:	0
		SMS received:	31

In this first screen you can see a summary of SMS sent via API and download an export.

Different fields and links

Name	Currentl y Used	Sample value	Comment
Subject	N		Field to be displayed in specific export, not used in this version
userdata	Y	Multiple	Filter on push where userdata contains the value, Multiple in our sample

message	Y	12345	Filter on push where ret_id contains the value, 12345 in our sample. Be careful, it's not a filter on the message text, but on the ret_id
Telephone	Y	+221771611010	Filter message where phone number (to) equals to the value.
Start Date	Y		Beginning of the period
End Date			End of the period
Filter	Y		Click on this link to display the number of SMS, on the period with filter. It's day per day displayed.
Graph	Y		Click on this link displays global statistics about status on the period. Some statistics are displayed about operators too.
Export messages in current view	Y		Generate a csv file with all the information about SMS send, with filter applied. See sample below.

7.2.2 Status day per day

Once you have clicked on 'filter link

Date	Route	Messages	Export
09/01/2015		Total MSISDN sent	0 GSM Network error: 0
		Blacklisted	0 Invalid number: 0
		Awaiting sending:	0 Expired: 0
		SMS received:	0 Other errors: 0
08/01/2015	Gold	Total MSISDN sent	4 GSM Network error: 0
		Blacklisted	0 Invalid number: 2
		Awaiting sending:	0 Expired: 0
		SMS received:	2 Other errors: 0

Data between 09/01/2015 - 08/01/2015

1

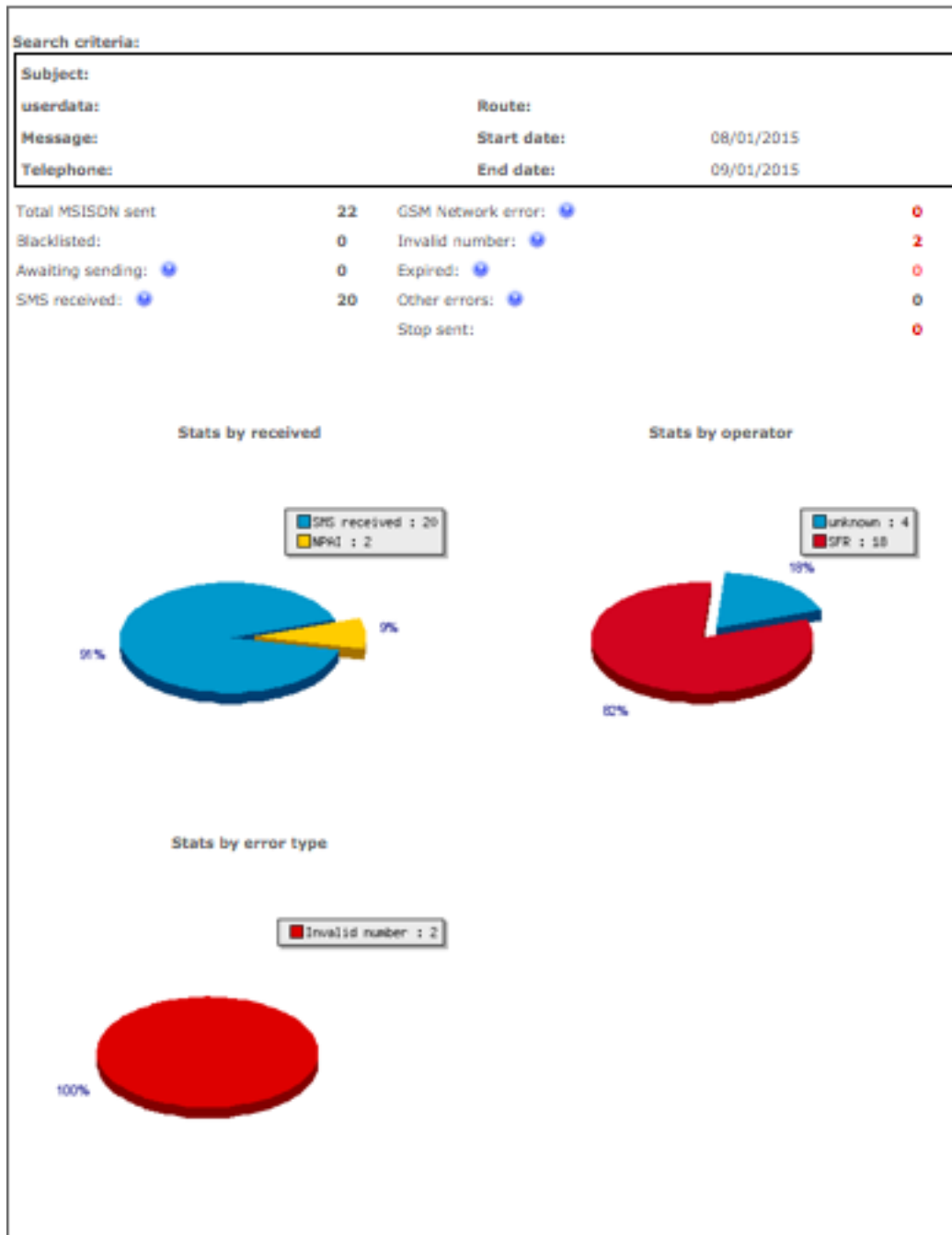
On this screen you can click on the small icon of a day to download detail.

ID	Account	User name	Name	Userdata	Number	Operator	Message	Date	Status	Error
1871011213	Gold	fred	Push Name	User Data Multiple Sent	00221771611010		Hello world : CheckStatus	08/01/15 11:26	RECEIVED	

1871011 213	Gold	fred	Push Name	User Data Multiple Sent	00221771 611010		Hello world : CheckStatus	08/01/15 11:26	ERROR	INVALID
1871011 352	Gold	fred	Push Name e	User Data Multiple Sent	00221771 611010		Hello world : CheckStatus	08/01/15 11:38	RECEIVED	
1871011 352	Gold	fred	Push Name e	User Data Multiple Sent	00221771 611010		Hello world : CheckStatus	08/01/15 11:38	ERROR	INVALID

See "export detail" for meaning of different columns.

7.2.3 Graph



Nota : In the current version statistics by operator are not correct, due to the lack of treatment of ported number

7.2.4 Export Details

When you click on “Export Message” a list of all the SMS corresponding to filters is proposed to download. :

On this screen you can click on the small icon of a day to download detail.

ID	Account	Username	Name	Userdata	Number	Operator	Message	Date	Status	Error
1871011213	Gold	fred	Push Name	User Data Multiple Sent	00221771611010		Hello world : CheckStatus	08/01/15 11:26	RECEIVED	
1871011213	Gold	fred	Push Name	User Data Multiple Sent	00221771611010		Hello world : CheckStatus	08/01/15 11:26	ERROR	INVALID
1871011352	Gold	fred	Push Name	User Data Multiple Sent	00221771611010		Hello world : CheckStatus	08/01/15 11:38	RECEIVED	
1871011352	Gold	fred	Push Name	User Data Multiple Sent	00221771611010		Hello world : CheckStatus	08/01/15 11:38	ERROR	INVALID

Columns are :

Name	Sample value	Comment
ID	1871011213	push_id of the SMS
Account	Gold	Route type of account used for SMS sending
Username	12345	Login of the account used (accountid)
Name	PushName	Name of the push, as set in " name " parameter of the <push> in XML.
UserData	User Data Multiple Sent	User data set in userdata , parameter of the <push> or <message> in XML
Number	+221771611010	Phone number in international format. With Excel, leading + is not displayed but it is present in the .cvs file (...;"221771611010";...)
Operator	60801	op_id of the operator. As mentioned this value is not always available right now in this current version of the API
Message	Hello World	Text of the message send
Date	08/01/15 11:38	Date of SMS sent.
Status	RECEIVED	Status of the SMS (SEND/RECEIVED/ERROR)
Error	INVALID	Possible values are INVALID : Operator cannot send the SMS, maybe the number was invalid, because of NPAI by example. EXPIRED : Operator tried to send SMS to a phone, but there was no answer. NETWORK_ERROR : Could not contact operator, maybe because number is not allocated to an operator BLOCKED : SMS was rejected with this

		status. NO_CREDIT : account has no credit left BLACKLISTED : number is blacklisted
--	--	--

Annexe

8.1 Using Variables

#cat VariableExample.xml

```
<?xml version="1.0" encoding="iso-8859-1"?>
<push
accountid="fred"
password="xpasswordx"
>
<message>
<text>Hello world mon %ami% de la ville de %NOM_VILLE% </text>
<to>0619896895
    <paramvar="%ami%" value="peps"/>
    <param var="%NOM_VILLE%" value="DIJON"/>
</to>
</message>
</push>
```

Thus XML sends this SMS :
Hello world mon peps de la ville de DIJON

8.2 Error XML

The following errors are detected in XML

```
XML_ERROR_NONE
XML_ERROR_NO_MEMORY
XML_ERROR_SYNTAX
XML_ERROR_NO_ELEMENTS
XML_ERROR_INVALID_TOKEN
XML_ERROR_UNCLOSED_TOKEN
XML_ERROR_PARTIAL_CHAR
XML_ERROR_TAG_MISMATCH
XML_ERROR_DUPLICATE_ATTRIBUTE
XML_ERROR_JUNK_AFTER_DOC_ELEMENT
XML_ERROR_PARAM_ENTITY_REF
XML_ERROR_UNDEFINED_ENTITY
XML_ERROR_RECURSIVE_ENTITY_REF
XML_ERROR_ASYNC_ENTITY
XML_ERROR_BAD_CHAR_REF
XML_ERROR_BINARY_ENTITY_REF
XML_ERROR_ATTRIBUTE_EXTERNAL_ENTITY_REF
XML_ERROR_MISPLACED_XML_PI
```


XML_ERROR_UNKNOWN_ENCODING
XML_ERROR_INCORRECT_ENCODING
XML_ERROR_UNCLOSED_CDATA_SECTION
XML_ERROR_EXTERNAL_ENTITY_HANDLING

8.3 Status Codes

NOT_SENT=>1 (SMS sent to the gateway, not yet to the operator)
ERROR=>2 (Impossible to connect to the gateway)
QUEUE=>3 (Queued by the gateway)
SENT=>4 (Sent to the operator)
RECEIVED=>6 (Received by the end-user)
ERROR_NPAI=>11 (“N’habites Pas à l’AdresseIndiquée” =”Return To Sender”): operator no longer knows this number
ERROR_EXPIRED=>12 (timeout, no RECEIVED, nor explicit ERRORafter SENT)
ERROR_INVOP=>13 (Invalid operator, this block number is not affected to an operator)
ERROR_NETWORK=>14 (error when posting SMS to the operator)
ERROR_CREDIT=>15 (insufficient credit on the recipient mobile)
ERROR_UNKNOWN=>16 (other error ...)

ERROR_PARSE=>17 :
ERROR_BLACKLISTED=>18 (Recipient is black listed)
ERROR_UNKNOWN_PROTOCOL=>19 (unknown protocol version)
ERROR_NOT_AUTHENTICATED=>20 (not authenticated)
ERROR_MESSAGE_TOO_LONG=>22 (max length 1600 characters oversized for the message)
ERROR_BLOCKED=>23 (Recipient is blocked by operator)

8.4 Different id

push_id : Id of the push, associated to an XML push, sent back by the gateway.

ret_id : id of a SMS, known by HighPush.

sms_id : internal id of a SMS, used by LAfricaMobile.