

Présentation des API Web Service Orange d'envoi/réception de SMS/Email

par François Marx

Date de publication : 31/01/2008

Dernière mise à jour : 21/02/2008

Les API Web Service SMS/Email d'Orange (gratuite, version alpha) offrent un cadre simple pour envoyer/recevoir des SMS et emails. Cet article a pour but de présenter les différentes méthodes des API à travers de courts exemples.

- I - Introduction
- II - L'API SMS
 - II-A - Envoyer un SMS
 - II-B - Recevoir un SMS
- III - L'API mail
 - III-A - Envoyer un mail
 - III-B - Lire des emails
- IV - Exemples
 - IV-A - Mini Poll
 - IV-B - Post It
- V - Utilisation des API avec le Zend Framework en PHP
- VI - Conclusion
 - VI-A - A vous de jouer
 - VI-B - Pour aller plus loin

I - Introduction

Pour faciliter l'accès à leur service, de nombreuses entreprises ont créé des Web Services librement accessibles sur le Web. L'API (on utilisera indifféremment dans la suite le terme de Web Service et API) cartographique de Google (déjà présentée sur le club des développeurs) est l'une des plus connues. Le site www.programmableweb.com recense pas moins de 615 API (chiffre à la date du 25/01/08).

Cet article se propose de vous présenter les **API d'Orange** (1) à travers quelques exemples.

Les API proposées par Orange couvrent actuellement l'envoi et la réception de SMS et d'emails. La plupart des API sont basées sur l'obtention d'une clé d'identification ou d'accès au Web Service qui permet notamment d'authentifier l'utilisateur. L'utilisateur devra s'enregistrer sur le site d'**Orange Partner** pour obtenir sa clé d'accès au service, il pourra aussi **configurer** différents paramètres des API. La clé est valable pour n'importe quel domaine et il n'y a pas de restrictions sur le type d'usage (commercial, privé...). Actuellement en version alpha, les API sont gratuites avec juste une limitation de volume pour l'envoi de SMS. Dans la suite de cet article, les API SMS et mail seront rapidement décrites puis nous verrons des cas d'utilisation pratiques principalement en PHP.

II - L'API SMS

Dans un style hybride REST-RPC, l'API SMS permet l'envoi et la réception de SMS. L'envoi/réception ne concerne actuellement que les mobiles Orange : cette restriction sera levée au cours du mois de mars 2008. Les requêtes peuvent utiliser indifféremment les méthodes GET et POST. Si l'on veut suivre la philosophie REST, on aurait tendance à choisir un GET pour récupérer des informations (par exemple lecture des emails) et un POST pour les actions (par exemple envoi de SMS).

II-A - Envoyer un SMS

L'envoi de SMS est aisé depuis son navigateur (dans notre cas, il s'agit d'un HTTP GET). Après avoir obtenu la clé d'accès, vous pourrez envoyer votre premier SMS en tapant l'URL suivante dans votre navigateur :

`http://sms.alpha.orange-api.net/sms/sendSMS.xml?id=[votre_clé_d_identification]a&to=[numéro_destinataire]&content=[texte]`

Ce qui donne pour un cas particulier:

`http://sms.alpha.orange-api.net/sms/sendSMS.xml?id=568d6a389aa&to=33612345678&content=Hello+World`

Ne pas oublier de remplacer la clé ci-dessus par votre propre clé d'accès ainsi que votre numéro de téléphone. L'appel aux API depuis un programme se fait aussi simplement. Le langage doit être capable d'effectuer des requêtes HTTP et de 'parser' du XML. A titre d'exemple, nous présentons ci-dessous l'envoi de SMS en PHP, Python et Java.

Code PHP

```
<?php
$phone_number = '33612345678';
$text_message = urlencode("Voici mon premier SMS avec les API d'Orange");
$api_access_key = '568d6a389aa';
$url =
    "http://sms.alpha.orange-api.net/sms/
sendSMS.xml?id=$api_access_key&to=$phone_number&content=$text_message";
$response = file_get_contents($url);
$xml = simplexml_load_string($response);
echo "Status: ", $xml->status->status_msg;
?>
```

Code Python

```
import urllib
import xml.etree.ElementTree as ET
phone_number = '33674867662'
text_message = 'Premier message Python'
urlbase = 'http://sms.alpha.orange-api.net/sms/sendSMS.xml?'
urlparam = urllib.urlencode([('id', '466b6e288fc'), ('to', phone_number), ('content', text_message)])
response = ET.parse(urllib.urlopen(urlbase+urlparam)).getroot()
print response.findtext('status/status_code')
```

Code Java

```
import java.io.*;
import java.net.*;

import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.xpath.XPath;
import javax.xml.xpath.XPathFactory;

import org.w3c.dom.Document;

public class SendSMS {
```

Code Java

```
public static void main(String[] args) throws Exception {
    String phoneNumber = "33674867662";
    String textMessage = URLEncoder.encode(
        "Voici mon premier SMS avec les API d Orange", "UTF-8");
    String apiKey = "568d6a389aa";
    String url = "http://sms.alpha.orange-api.net/sms/sendSMS.xml?id="
        + apiKey + "&to=" + phoneNumber + "&content="
        + textMessage;

    // Send GET request
    URL client = new URL(url);
    URLConnection conn = client.openConnection();
    InputStream responseBody = conn.getInputStream();

    // Convert in XML document
    Document response = DocumentBuilderFactory.newInstance().newDocumentBuilder().parse(responseBody);
    responseBody.close();
    // Xpath expression to get the status
    XPath xPath = XPathFactory.newInstance().newXPath();
    String status = xPath.evaluate("/response/status/status_msg", response);
    // display status
    System.out.println("Status: " + status);
}
}
```

Code VbScript, courtoisie de Bbil

```
Option explicit
Const ACCESS_KEY = "568d6a389aa"
Dim stNumDest, stMessage
stNumDest = "33612345678"
stMessage = "Hello Word"

EnvoiSms stNumDest, ACCESS_KEY, stMessage

Sub EnvoiSms(stNum, stACCESS_KEY, stMessage)
    Dim xmlDoc, stUrl
    stUrl = "http://sms.alpha.orange-api.net/sms/sendSMS.xml?id=" & _
        stACCESS_KEY & "&to=" & stNum & "&content=" & Escape(stMessage)
    Set xmlDoc = CreateObject("Microsoft.XMLDOM")
    xmlDoc.Async = "false"
    If xmlDoc.Load(stUrl) and not xmlDoc.selectSingleNode("/response/status/status_code") is Nothing
    Then
        MsgBox "Status_code = " & xmlDoc.selectSingleNode("/response/status/status_code").text
    else
        MsgBox "Erreur envoi", vbCritical
    End if
End sub
```

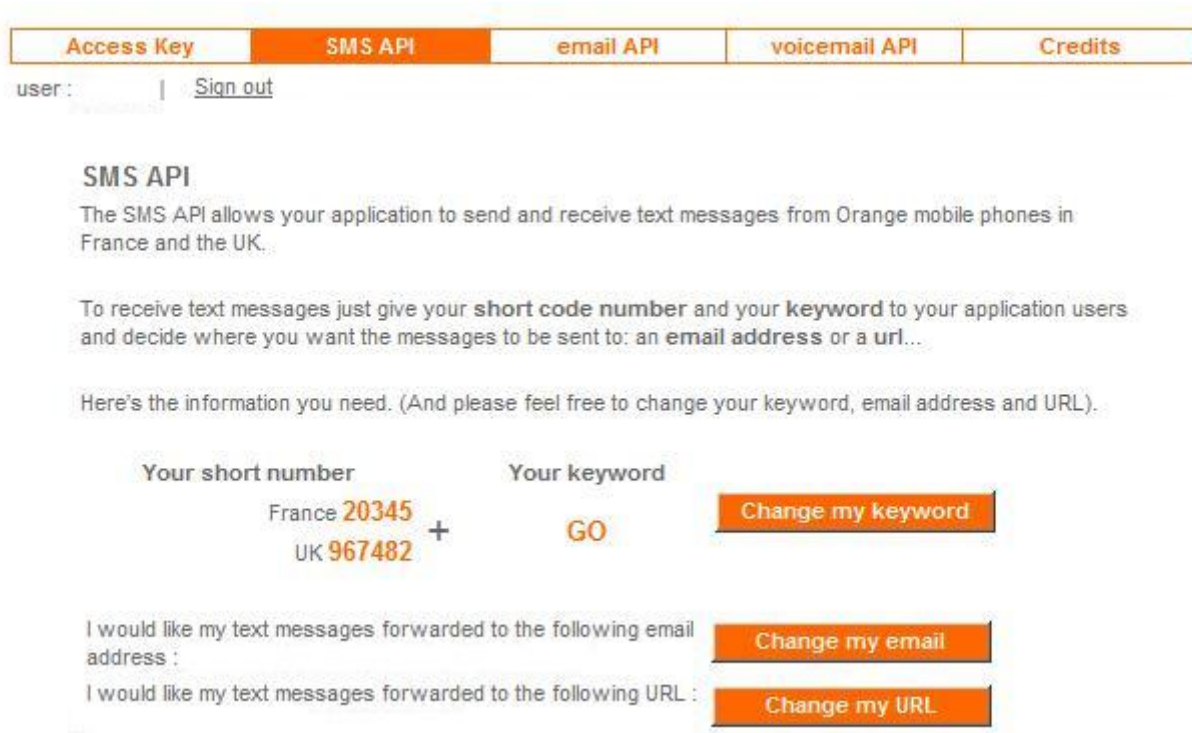
II-B - Recevoir un SMS

La réception de SMS est probablement la fonctionnalité la plus intéressante des API car elle permet au développeur de disposer de sa propre combinaison short code et préfixe. Tout d'abord dans l'interface d'administration, l'utilisateur des API devra configurer :

- Une adresse mail ou une URL pour recevoir le SMS
- Un préfixe (ou mot clé).

Chaque SMS expédié au 20345 (n° non surtaxé, prix d'un SMS classique) et commençant par le préfixe sera transféré à l'adresse email configurée dans l'interface d'administration. Une autre possibilité pour avoir une notification plus

immédiate est de spécifier l'adresse d'une URL, le serveur de SMS d'Orange faisant alors une requête HTTP GET sur cette URL avec comme paramètre le contenu du message.



The screenshot shows the Orange SMS API developer interface. At the top, there is a navigation bar with tabs for 'Access Key', 'SMS API', 'email API', 'voicemail API', and 'Credits'. Below the navigation bar, there is a 'user:' field and a 'Sign out' link. The main content area is titled 'SMS API' and contains the following text:

The SMS API allows your application to send and receive text messages from Orange mobile phones in France and the UK.

To receive text messages just give your **short code number** and your **keyword** to your application users and decide where you want the messages to be sent to: an **email address** or a **url**...

Here's the information you need. (And please feel free to change your keyword, email address and URL).

Under the heading 'Your short number', there are two options: 'France 20345' and 'UK 967482', separated by a plus sign. To the right of these options is a 'Change my keyword' button. Below this, there is a text input field with the placeholder 'I would like my text messages forwarded to the following email address :'. To the right of this field is a 'Change my email' button. Below that, there is another text input field with the placeholder 'I would like my text messages forwarded to the following URL :'. To the right of this field is a 'Change my URL' button.

Voici un exemple en PHP qui transfère votre SMS vers une liste de diffusion (utile pour la nouvelle année).

Code PHP

```
<?php
$api_access_key = '568d6a389aa';
$forward_list = array('nicolas' => '33612345678', 'martin' => '33687654321');
// save parameters from the receiveSMS method
$phone_number = $_GET['from']; // not used here
$msg = $_GET['content'];
// remove the first word (keyword)
$msg = urlencode(substr(strstr($msg, " "), 1));

foreach ($forward_list as $name => $ph) {
    // create the sendSMS URL using the correct parameters
    $msg_name = urlencode($name)."+".$msg;
    $url_sms =
"http://sms.alpha.orange-api.net/sms/sendSMS.xml?id=$api_access_key&to=$ph&content=$msg_name";
    // invoke it!
    $response = file_get_contents($url_sms);
    $xml = simplexml_load_string($response);
    echo "<br>Status: ", $xml->status->status_msg;
}
?>
```

III - L'API mail

Une boîte mail de 500Mo avec une adresse en orange.fr (par exemple api-12345678@orange.fr) vous a été attribuée lors de l'enregistrement. Cette API permet la gestion de cette boîte (il n'y a pas de webmail). On peut notamment l'utiliser pour récupérer les SMS transférés sur cette boîte mail.

III-A - Envoyer un mail

L'envoi de mail est simple. De façon semblable au SMS, vous pouvez envoyer un message en tapant simplement l'URL suivante dans votre navigateur :

`http://mail.alpha.orange-api.net/mail/sendMail.xml?id=[votre_clé_d_identification]`

`&from=[votre_nom]&to=[adresse_mail_destinataire]&subject=[le_sujet]&body=[le_corps_du_message]`

III-B - Lire des emails

Les différentes méthodes sont expliquées dans la documentation. Les principales méthodes sont `getMailList` qui retourne la liste des emails (sans le contenu proprement dit du message) et `getMail` qui retourne le contenu d'un message. D'autres méthodes sont disponibles pour marquer et compter les messages. Voici un petit programme PHP qui affiche le contenu de votre boîte aux lettres

Code PHP

```
<?php
$api_key = '568d6a389aa';
$url = "http://mail.alpha.orange-api.net/mail/getMailList.xml?id=$api_key";
$response = file_get_contents($url);
$xml = simplexml_load_string($response);
echo "<html>Mailbox<br>";
foreach ($xml->list->message as $m) {
    echo '<a
href="http://mail.alpha.orange-api.net/mail/getMailList.xml?id=', $api_key, '&number=', $m->number;
echo '> ', $m->subject, '</a> ', $m->from, ' ', $m->date, '<br>';
}
?>
```

IV - Exemples

IV-A - Mini Poll

Pour illustrer l'intérêt de la notification de réception d'un SMS, je vous propose de programmer un outil de sondage instantané par SMS. Le fonctionnement est le suivant :

- L'utilisateur envoie un SMS au numéro court 20345 avec le préfixe configuré par le développeur de Mini Poll (imaginons que vous ayez configuré POLL comme préfixe s'il est encore disponible). Pour lancer un sondage, l'initiateur envoie un SMS au 20345 avec le contenu POLL une question et des réponses. Chaque réponse doit commencer par A) ou B) #
- Le script PHP (qui s'exécute sur votre serveur ou chez votre hébergeur) est notifié par le serveur d'Orange (HTTP GET avec en paramètre le contenu du SMS) d'un nouveau SMS et il renvoie aux participants du sondage la question et les réponses (une extension serait de pouvoir configurer la liste de diffusion)
- Les utilisateurs répondent avec la lettre correspondant à la réponse (SMS commençant par POLL suivi de la lettre A ou B ou #)
- On peut voir en temps réel les résultats du sondage sur un site WEB (fichier html).

Dans un souci de simplicité, la liste de diffusion est configurée en dur dans le code, il n'y a pas de contrôle qu'un utilisateur ne vote pas deux fois et un seul sondage peut être organisé simultanément.

Mise en pratique, il faut tout d'abord configurer l'url de votre script dans l'interface Web de configuration des API orange et configurer un préfixe SMS (ou mot de code).

Le code est découpé en 2 scripts PHP, le premier (dont l'URL doit être configurée dans l'interface d'administration) reçoit les SMS, stocke dans un fichier les réponses et diffuse le sondage à la liste de diffusion. Le second fichier mixte HTML et PHP pour afficher les résultats du sondage.

Code PHP

```
<?php
define('LIMIT_NEW_POLL_ANSWER',20);

$difffusion_poll_list = array('nicolas' => '33612345678','martin' => '33687654321');
$api_access_key = '568d6a389aa';

// save parameters from the receiveSMS method
$mobile_number = $_GET['from'];
$sms_content= $_GET['content'];

// remove the keyword from the text message
$sms_content = substr(strstr($sms_content,' '),1);

// is a new poll or answer
if (strlen($sms_content) > LIMIT_NEW_POLL_ANSWER) {
// new poll
// store the content of the current poll
$fid = fopen('./data/poll_results.txt','w');
fwrite($fid,'a=1 b=2 c=3 d=4 e=5\n'); // init the poll results
fclose($fid);
$fid = fopen('./data/poll_question.txt','w');
fwrite($fid,$sms_content); // write the poll question
fclose($fid);
// forward the poll to diffusion list
$text_message = urlencode("Poll from $mobile_number $sms_content");
foreach ($difffusion_poll_list as $phone_number) {
// send sms
```

Code PHP

```

$url =
"http://sms.alpha.orange-api.net/sms/
sendSMS.xml?id=$api_access_key&to=$phone_number&content=$text_message";
file_get_contents($url);
}
}
else {
// answer to poll
$fid = fopen('./data/poll_results.txt','r');
list($a,$b,$c,$d,$e) = fscanf($fid,'a=%d b=%d c=%d d=%d e=%d');
fclose($fid);
// the first letter after the keyword is considered as the answer
$answer = strtolower($sms_content{0});
if (strpos("abcde",$answer) != FALSE) {
    ${$answer}++;
    $fid = fopen('./data/poll_results.txt','w');
    fprintf($fid,'a=%d b=%d c=%d d=%d e=%d\n',$a,$b,$c,$d,$e);
    fclose($fid);
}
}
?>
    
```

Code PHP/HTML

```

<html>
<title>Sondage</title>
Sondage<br>
<?php
    $fid = fopen('./data/poll_results.txt','r');
    list($a,$b,$c,$d,$e) = fscanf($fid,'a=%d b=%d c=%d d=%d e=%d');
    fclose($fid);
    $question = file_get_contents('./data/poll_question.txt'); // read the entire question
?>
Question:<br>
<?php echo $question ?>
<br>
Résultats:
<ul>
    <li>Réponse A: <?php echo $a ?> votes</li>
    <li>Réponse B: <?php echo $b ?> votes</li>
    <li>Réponse C: <?php echo $c ?> votes</li>
    <li>Réponse D: <?php echo $d ?> votes</li>
    <li>Réponse E: <?php echo $e ?> votes</li>
</ul>
</html>
    
```

IV-B - Post It

Pour illustrer l'API mail, je vous propose de réaliser une application de Post It Web. Ce programme utilise les principales méthodes de l'API Mail. Le fonctionnement est le suivant :

- Imaginons que vous ayez configuré POSTIT comme préfixe s'il est encore disponible.
- Il faut aussi configurer dans l'interface d'administration l'adresse email de forward des SMS vers l'adresse email qui vous est attribuée en @orange.fr
- L'utilisateur envoie alors un SMS avec le contenu POSTIT suivi du contenu du post it ou un mail avec POSTIT comme premier mot du corps de l'email et le contenu du post it dans le corps du texte.
- Le script php génère une page Web post it où l'utilisateur verra l'intégralité de ces post it.

Code PHP

```
<?php
echo '<html><body><ul>';
echo '<br>Post It<br>';
$apiaccesskey = '568d6a389aa';
$keyword = 'postit'; // in lowercase
$url = 'http://mail.alpha.orange-api.net/mail/getMailList.xml?id='.$apiaccesskey;
$response = file_get_contents($url);
$xmllist = simplexml_load_string($response);
foreach ($xmllist->list->message as $m) {
    $url =
    'http://mail.alpha.orange-api.net/mail/getMail.xml?id='.$apiaccesskey.'&number='.$m->number;
    $response = file_get_contents($url);
    $xmlmsg = simplexml_load_string($response);
    // search for the keyword in the first characters (add 2 to keyword length because some mailer
    add \n, ' ')
    $sp = strpos(strtolower(substr($xmlmsg->message->body,0,strlen($keyword)+2)), $keyword);
    if ($sp !== FALSE) {
        // display date and message
        // remove keyword and limit message length to 160 characters
        echo '<li>'.$xmlmsg->message->date.' --
        '.substr($xmlmsg->message->body, $sp+strlen($keyword), 160).'</li>';
    }
}
echo '</ul></html>';
?>
```

V - Utilisation des API avec le Zend Framework en PHP

Un sous-ensemble des API Yahoo, Google, Amazon est intégré au Zend Framework. Ce n'est évidemment pas le cas des API d'Orange (du moins pour le moment). Cependant, on peut parfaitement utiliser les API SMS et Mail avec les classes Web Services génériques mis à disposition par ce framework. Pour le Zend Framework, la classe que nous allons utiliser est `Zend_Rest_Client`.

L'exemple ci-dessous affiche le nombre de messages dans sa boîte mail.

Code PHP framework Zend

```
<?php
require_once 'Zend/Rest/Client.php';
$rest = new Zend_Rest_Client('http://mail.alpha.orange-api.net');
$rest->id("mettre_votre_clé_d'accès");
$r = $rest->get('/mail/getMailList.xml');
echo 'Nombre de messages ' . $r->nbMsg;
?>
```

VI - Conclusion

VI-A - A vous de jouer

Cet article s'est attaché à décrire les API SMS et Mail d'Orange qui pourront enrichir les canaux de communication de vos programmes. Les possibilités sont nombreuses : de la commande de **pizzas**, aux jeux par SMS en passant par une intégration dans **Facebook**. D'autres API sont actuellement en cours de développement et feront l'objet d'autres articles dans un proche avenir.

VI-B - Pour aller plus loin

Ce blog présente des applications réalisées à partir des API d'Orange : <http://telcoapps.wordpress.com/>

Le site programmableweb présente un catalogue de WEB Services : <http://www.programmableweb.com/apilist>

L'article de J. Sorel présente les API GoogleMaps : <http://jsorel.developpez.com/tutoriels/sig/googlemaps1/>

La documentation de l'API SMS :

http://www.orangepartner.com/site/enuk/develop/devplayzone/smsapi/p_sms_api_alpha.jsp

La documentation de l'API Mail :

http://www.orangepartner.com/site/enuk/develop/devplayzone/emailapi/p_email_api_alpha.jsp

1 : L'auteur de ce tutorial travaille pour Orange et a contribué aux développements des API.