



## SMS Gateway

Conexión a sistema de mensajería MT de M-Lat  
vía Servicio Web

## Introducción

El presente documento especifica detalles técnicos de la conexión con la mensajería MT de M-Lat utilizando la tecnología de servicios Web. Este es uno de los mecanismos que ofrece el producto M-Gateway para permitir aplicaciones de terceros realizar envíos de SMS a través de la plataforma de M-Lat.

En adelante se llamará Servicio Web a aquellos componentes que ofrecen procedimientos remotos utilizando como transporte HTTP codificando parámetros de entrada y salida en documentos XML apegado a los estándares del W3C.

Para realizar invocaciones se debe utilizar una biblioteca que permita manejar los protocolos y el procesamiento del XML. Para los lenguajes de programación más populares se disponen de herramientas que a partir de la descripción WSDL crean componentes que ofrecen una versión local de los procedimientos remotos y oculta la complejidad de los protocolos y el procesamiento del XML.

## El servicio Web de M-Lat

La URL de la descripción WSDL del servicio es

<http://m-lat.net:8080/axis2/services/InternationalGateway2?wsdl>

Nótese que se utiliza una conexión segura y para usarla correctamente en las integraciones se debe forzar el uso de endpoint seguro. Ejemplo:

<https://m-lat.net:8443/axis2/services/InternationalGateway2.InternationalGateway2HttpSoap12Endpoint>

Se ofrecen los procedimientos remotos:

### *sendOneSMS*

Envía un mensaje de texto corto. Permite un mensaje de hasta 160 caracteres personalizando el remitente. Devuelve el identificador del mensaje. Dato útil para identificar los acuses de recibo.

Parámetros:

- user: nombre de usuario del cliente (usado para la autenticación)
- password: contraseña del cliente (usado para la autenticación)
- src: remitente personalizado. Puede ser un número de teléfono al cual responder o una palabra de hasta 11 caracteres. Note que no todas las operadoras soportan esta característica
- dest: número destino. Debe incluir sólo números empezando por el código del país
- text: texto del mensaje. La longitud máxima es de 160 caracteres. Evite el uso de caracteres especiales como acentos y ñ

### *sendManySMS*

Envía varios mensajes de texto corto. En realidad es un atajo a hacer las llamadas individuales al método de enviar un sms. Hace más eficiente la comunicación al requerir una sola conexión al servidor.

- user: nombre de usuario del cliente (usado para la autenticación)
- password: contraseña del cliente (usado para la autenticación)
- srcs: remitentes personalizados para cada mensaje
- dests: números destinos para cada mensaje

- texts: textos de los mensajes a enviar

Se devuelve una lista de identificadores de los mensajes enviados útil para identificar mensajes de los acuses de recibo.

### Invocaciones desde Java

La biblioteca axis2 <http://ws.apache.org/axis2> permite la creación de un conjunto de clases con métodos equivalentes a los publicados por el servicio web. La aplicación wsdl2java.sh recibe por parámetros la dirección del WSDL y genera varias clases.

El comando:

```
$ wsdl2java.sh -uri http://m-lat.net:8080/axis2/services/InternationalGateway2?wsdl
```

dejará en el directorio actual una carpeta “src” con las clases

- net.mlat.sms.international.ExceptionException0.java
- net.mlat.sms.international.InternationalGateway2CallbackHandler.java
- net.mlat.sms.international.InternationalGateway2Stub.java

El último es la clase con los métodos del servicio web. Para hacer invocaciones se debe crear una instancia de dicha clase forzando el endpoint sobre SSL.

```
// se establece el endpoint sobre SSL
InternationalGateway2Stub stub=new InternationalGateway2Stub(
    "https://m-lat.net:8443/axis2/services/" +
    "InternationalGateway2.InternationalGateway2HttpSoap12Endpoint");
InternationalGateway2Stub.SendOneSMS send=
    new InternationalGateway2Stub.SendOneSMS();
send.setUser("<usuario>");
send.setPassword("<password>");
send.setSrc("<remitente>"); // remitente
send.setDest("<número destino>"); // número destino
send.setText("<texto del mensaje>"); // texto del mensaje
InternationalGateway2Stub.SendOneSMSResponse resp;
resp=stub.sendOneSMS(send); // envío del mensaje
int id_mensaje=resp.get_return();
```

### Invocaciones desde .NET

Como parte de la distribución del .NET Framework y de mono se incluye la herramienta wsdl.exe que a partir de la URL del WSDL genera un conjunto de clases con métodos equivalentes a los publicados por el servicio web.

Ejemplo usando mono:

```
$ wsdl http://m-lat.net:8080/axis2/services/InternationalGateway2?wsdl
```

Es posible que deba ejecutar primero “mozroots –import” para que se reconozca el certificado de seguridad y pueda ser posible la conexión SSL.

La aplicación generará un archivo “InternationalGateway2.cs”. En este archivo se requiere referencia a los assembly System.Web.Services y System.Xml. Ejemplo de uso de la clase generada:

```

using System;

namespace WSTest
{
    class MainClass
    {
        public static void Main(string[] args)
        {
            InternationalGateway2 ws=new InternationalGateway2();
            ws.Url="https://m-lat.net:8443/axis2/services/"+
                "InternationalGateway2.InternationalGateway2HttpSoap11Endpoint";
            int id=ws.sendOneSMS("<usuario>","<password>","<remitente>","
                "<número destino>","<texto del mensaje>");
            Console.WriteLine("identificador sms: "+id);
        }
    }
}

```

### Invocaciones desde PHP 5

Se requiere que el mod\_php esté compilado con soporte a SoapClient (--enable-soap). De esa manera se dispone de la clase SoapClient que permite hacer la invocación. Un ejemplo:

```

<?php
try {
    $user='<usuario>';
    $password='<password>';
    $text='prueba sms desde php';
    $src='MLAT';
    $recipient='<numero destino>';

    $mlat=new SoapClient(
        'https://m-lat.net:8443/axis2/services/InternationalGateway2?wsdl',
        array('location'=>
            'https://m-lat.net:8443/axis2/services/InternationalGateway2?wsdl'));
    $mlat->sendOneSMS( array( 'user'=>$user, 'password'=>$password, 'src'=>$src,
        'dest'=>$recipient, 'text'=>$text));

} catch(Exception $ex) {
    echo $ex->getMessage();
    echo "\n";
    echo $ex->getTraceAsString();
    echo "\n";
}
?>

```