

Documentación servicios Web

DINAPAQ



ÍNDICE

1 Servicios	3
1.1 Servicio "LoginWSService"	3
1.1.1 Métodos:.....	3
1.1.2 Descripción y uso de los métodos:	3
1.1.2.1 Método Login	3
1.1.2.2 Método LoginCli.....	6
1.1.2.3 Método LoginDep	8
1.1.3 Excepciones.....	10
1.2 Servicio "WebServService"	11
1.2.1 Métodos:.....	11
1.2.2 Descripción y uso de los métodos:	12
1.2.2.1 Método ConsEnvEstados.....	12
1.2.2.2 Método ConsRecEstados.....	15
1.2.2.3 Método ConsEnvio	18
1.2.2.4 Método ConsRecogida	23
1.2.2.5 Método GrabaEnvio10	27
1.2.2.6 Método GrabaRecogida2	32
1.2.2.7 Método InfEnvios	37
1.2.2.8 Método ConsEtiquetaEnvio4.....	42
1.2.2.9 Método GrabaPedido.....	44
1.2.2.10 Método ConsEnvEstadosUltimo	47
1.2.2.11 Método ConsEnvIncidencias	50
1.2.2.12 Método ConsEnvEstadosFecha	54
1.2.2.13 Método ConsEnvIncidenciasFecha.....	58
1.2.2.14 Método GrabaEnvIncActuacion	62
1.2.2.15 Método ConsEnvTraz.....	64
1.2.2.16 Método ConsEnvTrazFecAge.....	68
1.2.2.17 Método DigitalizaEnvios	72
1.2.2.18 Método ConsEnvEstadosRef	75
1.2.2.19 Método ConsUltimoEstadoEnvio	78
1.2.2.20 Método InfEnviosRef.....	80
1.2.2.21 Método ConsURLDocAgrupados.....	85
1.2.2.22 Método GrabaEnvIncActuacionLibre.....	87
1.2.2.23 Método GrabaRecIncActuacionLibre.....	89
1.2.2.24 Método ConsRecogidasRef.....	92
1.2.3 Excepciones.....	96

1 Servicios

1.1 Servicio “LoginWSService”

Servicio de login para clientes mediante Web services.

MÉTODOS.	DESCRIPCIÓN
Login	Permite hacer login a los usuarios.
LoginCli	Permite hacer login a un cliente final.
LoginDep	Permite hacer login a un departamento.

1.1.1 Métodos:

El servicio LoginWSService consta de las siguientes funciones/procedimientos:

1.1.2 Descripción y uso de los métodos:

1.1.2.1 Método Login			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCod	Entrada	String	Código del usuario.
strPass	Entrada	String	Contraseña del usuario.
strNom	Salida	String	Nombre del usuario.
strCodAge	Salida	String	Código de agencia del usuario.
strCodCR	Salida	String	Código del centro regional del usuario.
strTipo	Salida	String	Tipo de usuario.
strVersion	Salida	String	Versión del programa.
strError	Salida	String	Error devuelto en el login.
strSesion	Salida	String	ID de la sesión.
strURLDetSegEnv	Salida	String	URL del detalle de seguimiento de envíos.
Valor devuelto	Valor	Boolean	Indica si se ha realizado el login correctamente.

Uso:

Si disponemos de un lenguaje que interprete los archivos .wsdl (como por ejemplo C#), simplemente tendremos que crear una referencia Web importando el fichero .wsdl y a continuación deberemos instanciar esa referencia para poder llamar a las funciones.

Ejemplo:

C#

```
string strNombre, strAgencia, strCodCR, strTipo, strVersion, strError, strSesion,
    strURLDetSegEnv;
```

```

LoginService.LoginWSService wsLoginService = new LoginService.LoginWSService();

if(wsLoginService.Login("usuario", "contraseña", out strNombre, out strAgencia,
    out strCodCR, out strTipo, out strVersion, out strError, out strSesion,
    out strURLDetSegEnv))
    MessageBox.Show("Login correcto");
else
    MessageBox.Show("Login incorrecto");
  
```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```

PHP

<?php
    $strRespuestaServer = http_post_data('http://--ip servidor--/soap',
    '<?xml version="1.0" encoding="utf-8"?>
        <soap:Envelope
            xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema">
            <soap:Body>
                <LoginWSService__Login>
                    <strCod>--usuario--</strCod>
                    <strPass>--contraseña--</strPass>
                </LoginWSService__Login>
            </soap:Body>
        </soap:Envelope>');
    $strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>
  
```

Una vez ejecutado el http_post_data la variable \$strRespuestaServer contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

Ejemplo:

```

XML

<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:HNS="http://tempuri.org/"
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:v1="http://tempuri.org/">
    <SOAP-ENV:Header>
        <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
            <ID>--OUT ID Sesión--</ID>
        </ROClientIDHeader>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
        <v1:LoginWSService__LoginResponse>
  
```

```
<v1:Result>--OUT Valor Devuelto--</v1:Result>
<v1:strNom>--OUT Nombre Usuario--</v1:strNom>
<v1:strCodAge>--OUT Código Agencia--</v1:strCodAge>
<v1:strCodCR>--OUT Código Cen. Reg.--</v1:strCodCR>
<v1:strTipo>--OUT Tipo Usuario--</v1:strTipo>
<v1:strVersion>--OUT Version Prog.--</v1:strVersion>
<v1:strError>--OUT Código error--</v1:strError>
<v1:strSesion>--OUT ID Sesión--</v1:strSesion>
<v1:strURLDetSegEnv>--OUT URL Det Seg Env--</v1:strURLDetSegEnv>
</v1:LoginWSService___LoginResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Excepciones:

Esta función devuelve las siguientes excepciones:

CÓDIGO	DESCRIPCIÓN
1	Usuario o password incorrecto.
2	Usuario no activo.
3	Usuario no tiene agencia asignada.
4	No hay permisos para utilizar este servicio

Para saber cómo controlar las excepciones devueltas por favor consulte el apartado 1.1.3, Excepciones.

1.1.2.2 Método LoginCli			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodAge	Entrada	String	Código de la agencia del cliente.
strCod	Entrada	String	Código del cliente.
strPass	Entrada	String	Contraseña del cliente.
strNom	Salida	String	Nombre del cliente.
strCodCR	Salida	String	Código del centro regional del cliente.
strTipo	Salida	String	Tipo de usuario.
strVersion	Salida	String	Versión del programa.
strError	Salida	String	Error devuelto en el login.
strSesion	Salida	String	ID de la sesión.
strURLDetSegEnv	Salida	String	URL del detalle de seguimiento de envíos.
Valor devuelto	Valor	Boolean	Indica si se ha realizado el login correctamente.

Uso:

El uso es similar a la anterior función.

Ejemplo:

```
C#

string strNombre, strCodCR, strTipo, strVersion, strError, strSesion,
    strURLDetSegEnv;

LoginService.LoginWSService wsLoginService = new LoginService.LoginWSService();

if(wsLoginService.LoginCli("cod-agencia", "cod-cliente", "contraseña", out strNombre,
    out strCodCR, out strTipo, out strVersion, out strError, out strSesion,
    out strURLDetSegEnv))
    MessageBox.Show("Login correcto");
else
    MessageBox.Show("Login incorrecto");
```

Para lenguajes que no soporten .wsdl:

Ejemplo:

```
PHP

<?php

$strRespuestaServer = http_post_data('http://--ip servidor--/soap',
    '<?xml version="1.0" encoding="utf-8"?>
        <soap:Envelope
            xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema">
            <soap:Body>
                <LoginWSService__LoginCli>
                    <strCodAge>--cod-agencia--</strCodAge>
                    <strCod>--cod-cliente --</strCod>
                    <strPass>--contraseña--</strPass>
```

```

        </LoginWSService___LoginCli>
    </soap:Body>
</soap:Envelope>';
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>

```

Una vez ejecutado el `http_post_data` la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

Ejemplo:

XML

```

<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/">
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--OUT ID Sesión--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:LoginWSService___LoginCliResponse>
      <v1:Result>--OUT Valor Devuelto--</v1:Result>
      <v1:strNom>--OUT Nombre Usuario--</v1:strNom>
      <v1:strCodCR>--OUT Código Cen. Reg.--</v1:strCodCR>
      <v1:strTipo>--OUT Tipo Usuario--</v1:strTipo>
      <v1:strVersion>--OUT Version Prog.--</v1:strVersion>
      <v1:strError>--OUT Código error--</v1:strError>
      <v1:strSesion>--OUT ID Sesión--</v1:strSesion>
      <v1:strURLDetSegEnv>--OUT URL Det Seg Env--</v1:strURLDetSegEnv>
    </v1:LoginWSService___LoginCliResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Excepciones:

Esta función devuelve las siguientes excepciones:

CÓDIGO	DESCRIPCIÓN
1	Cliente o password incorrecto.
2	No hay permiso para utilizar este servicio.

Para saber cómo controlar las excepciones devueltas por favor consulte el apartado 1.1.3, Excepciones

1.1.2.3 Método LoginDep			
MÉTODO	E/S	TIPO	DESCRIPCIÓN
strCodAge	Entrada	String	Código de la agencia del cliente.
strCodCli	Entrada	String	Código del cliente
strCod	Entrada	String	Código del departamento.
strPass	Entrada	String	Contraseña del departamento.
strNom	Salida	String	Nombre del departamento.
strCodCR	Salida	String	Código del centro regional del departamento.
strTipo	Salida	String	Tipo de usuario.
strVersion	Salida	String	Versión del programa.
strError	Salida	String	Error devuelto en el login.
strSesion	Salida	String	ID de la sesión.
strURLDetSegEnv	Salida	String	URL del detalle de seguimiento de envíos.
Valor devuelto	Valor	Boolean	Indica si se ha realizado el login correctamente.

Uso:

El uso es similar a la anterior función.

Ejemplo:

```
C#

string strNombre, strCodCR, strTipo, strVersion, strError, strSesion,
    strURLDetSegEnv;

LoginService.LoginWSService wsLoginService = new LoginService.LoginWSService();

if(wsLoginService.LoginDep("cod-agencia", "cod-cliente", "cod-departamento",
    "contraseña", out strNombre, out strCodCR, out strTipo, out strVersion,
    out strError, out strSesion, out strURLDetSegEnv))
    MessageBox.Show("Login correcto");
else
    MessageBox.Show("Login incorrecto");
```

Para lenguajes que no soporten .wsdl:

Ejemplo:

```
PHP

<?php
$strRespuestaServer = http_post_data('http://--ip servidor--/soap',
'<?xml version="1.0" encoding="utf-8"?>
    <soap:Envelope
        xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        <soap:Body>
```



```

        <LoginWSService__LoginDep>
          <strCodAge>--cod-agencia--</strCodAge>
          <strCodCli>--cod-cliente--</strCodCli>
          <strCod>--cod-departamento--</strCod>
          <strPass>--contraseña--</strPass>
        </LoginWSService__LoginDep>
      </soap:Body>
    </soap:Envelope>');
    $strRespuestaServer = http_parse_message($strRespuestaServer)->body;
  ?>

```

Una vez ejecutado el `http_post_data` la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

Ejemplo:

```

XML
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/">
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--OUT ID Sesión--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:LoginWSService__LoginDepResponse>
      <v1:Result>--OUT Valor Devuelto--</v1:Result>
      <v1:strNom>--OUT Nombre Usuario--</v1:strNom>
      <v1:strCodCR>--OUT Código Cen. Reg.--</v1:strCodCR>
      <v1:strTipo>--OUT Tipo Usuario--</v1:strTipo>
      <v1:strVersion>--OUT Version Prog.--</v1:strVersion>
      <v1:strError>--OUT Código error--</v1:strError>
      <v1:strSesion>--OUT ID Sesión--</v1:strSesion>
      <v1:strURLDetSegEnv>--OUT URL Det Seg Env--</v1:strURLDetSegEnv>
    </v1:LoginWSService__LoginDepResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Excepciones:

Esta función devuelve las siguientes excepciones:

CÓDIGO	DESCRIPCIÓN
1	Departamento o password incorrecto.
2	No hay permisos para utilizar este servicio

Para saber cómo controlar las excepciones devueltas por favor consulte el apartado 1.1.3, Excepciones.

1.1.3 Excepciones

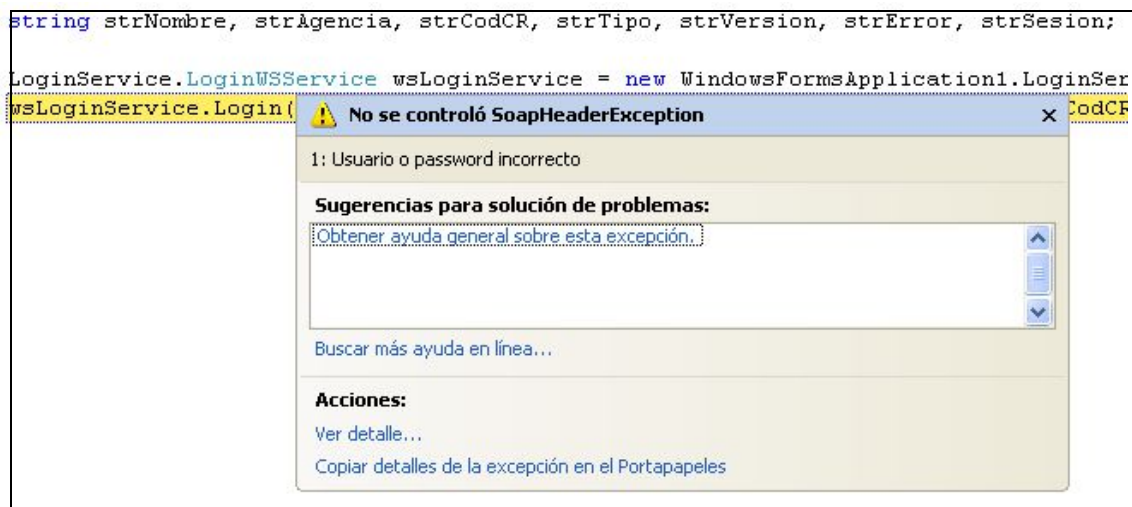
Cuando se produzca una excepción en el servidor éste nos devolverá un XML con el siguiente formato del que tendremos que sacar los valores:

```
XML
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:HNS="http://tempuri.org/"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID-SESION--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <SOAP-ENV:Fault>
      <faultcode>Exception</faultcode>
      <faultstring>--DESCRIPCION-EXCEPCION--</faultstring>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

En el caso de que el lenguaje que utilicemos disponga de cliente de servicios Web, es muy probable que la interpretación de este XML sea automática y sea el propio cliente el que lance una excepción nativa en el programa, que deberemos controlar:

La excepción siempre va a estar compuesta por un código (numérico), seguida de dos puntos (:), y a continuación la descripción de la excepción.

Ejemplo:



1.2 Servicio “WebServService”

Servicio de operaciones para clientes mediante Web services.

1.2.1 Métodos:

El servicio WebServService consta de las siguientes funciones/procedimientos:

MÉTODOS.	DESCRIPCIÓN
ConsEnvEstados	Permite consultar los estados de un envío a partir de su código de envío.
ConsRecEstados	Permite consultar los estados de una recogida a partir de su código de recogida.
ConsEnvio	Permite consultar los datos generales de un envío a partir de su código de envío.
ConsRecogida	Permite consultar los datos generales de una recogida a partir de su código de recogida.
GrabaEnvio10	Permite insertar un envío.
GrabaRecogida	Permite insertar o modificar una recogida.
InfEnvios	Permite consultar los datos de los envíos que tengan como fecha de salida o recepción los parámetros.
ConsEtiquetaEnvio2	Devuelve la etiqueta de un envío codificada en base64.
GrabaPedido	Graba un pedido.
ConsEnvEstadosUltimo	Permite consultar los últimos estados de los envíos indicados por parámetros.
ConsEnvIncidencias	Permite consultar las incidencias de un envío a partir de su código de envío.
ConsEnvEstadosFecha	Permite consultar los estados de envíos que coincidan con la fecha.
ConsEnvIncidenciasFecha	Permite consultar las incidencias de envíos que coincidan con la fecha.
GrabaEnvIncActuacion	Graba una actuación a una incidencia indicada por parámetros.

1.2.2 Descripción y uso de los métodos:

1.2.2.1 Método ConsEnvEstados			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodAgeCargo	Entrada	String	Agencia de cargo.
strCodAgeOri	Entrada	String	Agencia de origen.
strAlbaran	Entrada	String	Albarán.
strEnvEstados	Salida	String	XML con los datos de los estados del envío.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio [“LoginWSService”](#).

Uso:

Si disponemos de un lenguaje que interprete los archivos .wsdl (como por ejemplo C#), simplemente tendremos que crear una referencia Web importando el fichero .wsdl y a continuación instanciar esa referencia para poder llamar a las funciones.

Ejemplo:**C#**

```
WebService.WebServService wsWebService = new WebService.WebServService();  
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();  
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;  
  
string xmlResult = wsWebService.ConsEnvEstados("agencia-cargo", "agencia-origen", "albaran");
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>  
<ENV_ESTADOS  
  I_ID="--ID DEL ESTADO 1--"  
  V_COD_TIPO_EST="--TIPO ESTADO--"  
  D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"  
  V_COD_USU_ALTA="--USUARIO ALTA--"  
  V_COD_AGE_ALTA="--AGENCIA ALTA--"  
  V_COD_REP_ALTA="--REPARTIDOR ALTA--"
```

```

V_COD_CLI_ALTA="--CLIENTE ALTA--"
V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
/>
.
.
//DEVOLVERÁ UN BLOQUE POR CADA ESTADO QUE TENGA ESE ENVÍO
.
<ENV_ESTADOS
  I_ID="--ID DEL ESTADO N--"
  V_COD_TIPO_EST="--ETC --"
  D_FEC_HORA_ALTA="--ETC en formato mm/dd/yyyy hh:mm:ss--"
  V_COD_USU_ALTA="--ETC --"
  V_COD_AGE_ALTA="--ETC --"
  V_COD_REP_ALTA="--ETC --"
  V_COD_CLI_ALTA="--ETC --"
  V_COD_CLI_DEP_ALTA="--ETC --"
/>
</CONSULTA>

```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```

PHP

<?php
$strRespuestaServer = http_post_data('http://--ip servidor--/soap',
  '<?xml version="1.0" encoding="utf-8"?>
    <soap:Envelope
      xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema">
      <soap:Header>
        <ROClientIDHeader xmlns="http://tempuri.org/">
          <ID>--ID de la sesión establecida--</ID>
        </ROClientIDHeader>
      </soap:Header>
      <soap:Body>
        <WebServService___ConsEnvEstados>
          <strCodAgeCargo>-agencia-cargo-</strCodAgeCargo>
          <strCodAgeOri>--agencia-origen--</strCodAgeOri>
          <strAlbaran>--albaran--</strAlbaran>
        </WebServService___ConsEnvEstados>
      </soap:Body>
    </soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>

```

Una vez ejecutado el http_post_data la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/"
>
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService___ConsEnvEstadosResponse>
      <v1:strEnvEstados>
        <CONSULTA>
          <ENV_ESTADOS
            I_ID="--ID DEL ESTADO 1--"
            V_COD_TIPO_EST="--TIPO ESTADO--"
            D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
            V_COD_USU_ALTA="--USUARIO ALTA--"
            V_COD_AGE_ALTA="--AGENCIA ALTA--"
            V_COD_REP_ALTA="--REPARTIDOR ALTA--"
            V_COD_CLI_ALTA="--CLIENTE ALTA--"
            V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
          />
          .
          . //DEVOLVERÁ UN BLOQUE POR CADA ESTADO QUE TENGA ESE ENVÍO
          .
          <ENV_ESTADOS
            I_ID="--ID DEL ESTADO N--"
            V_COD_TIPO_EST="--ETC --"
            D_FEC_HORA_ALTA="--ETC en formato mm/dd/yyyy hh:mm:ss--"
            V_COD_USU_ALTA="--ETC --"
            V_COD_AGE_ALTA="--ETC --"
            V_COD_REP_ALTA="--ETC --"
            V_COD_CLI_ALTA="--ETC --"
            V_COD_CLI_DEP_ALTA="--ETC --"
          />
        </CONSULTA>
      </v1:strEnvEstados>
    </v1:WebServService___ConsEnvEstadosResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

1.2.2.2 Método ConsRecEstados

PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodRec	Entrada	String	Código de recogida.
strRecEstados	Salida	String	XML con los datos de los estados de la recogida.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio [“LoginWSService”](#).

Uso:

El uso es similar a la anterior.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

string xmlResult = wsWebService.ConsRecEstados("codigo-recogida");
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>
<REC_ESTADOS=""
  I_ID="--ID DEL ESTADO 1--"
  V_COD_TIPO_EST="--TIPO ESTADO--"
  D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
  V_COD_USU_ALTA="--USUARIO ALTA--"
  B_ULT="--ULTIMO --"
  V_COD_AGE_ALTA="--AGENCIA ALTA--"
  V_COD_REP_ALTA="--REPARTIDOR ALTA--"
  V_COD_CLI_ALTA="--CLIENTE ALTA--"
  V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
/>

. //DEVOLVERÁ UN BLOQUE POR CADA ESTADO QUE TENGA ESA RECOGIDA
.
<REC_ESTADOS
  I_ID="--ID DEL ESTADO N--"
  V_COD_TIPO_EST="--ETC --"
  D_FEC_HORA_ALTA="--ETC en formato mm/dd/yyyy hh:mm:ss--"
  V_COD_USU_ALTA="--ETC --"
```

```

B_ULT="--ULTIMO --"
V_COD_AGE_ALTA="--ETC --"
V_COD_REP_ALTA="--ETC --"
V_COD_CLI_ALTA="--ETC--"
V_COD_CLI_DEP_ALTA="--ETC--"
/>
</CONSULTA>

```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```

PHP
<?php
    $strRespuestaServer = http_post_data('http://--ip servidor--/soap',
        '<?xml version="1.0" encoding="utf-8"?>
        <soap:Envelope
            xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema">
            <soap:Header>
                <ROClientIDHeader xmlns="http://tempuri.org/">
                    <ID>--id de la sesión establecida--</ID>
                </ROClientIDHeader>
            </soap:Header>
            <soap:Body>
                <WebServService__ConsRecEstados>
                    <strCodRec>--código-recogida--</strCodRec>
                </WebServService__ConsRecEstados>
            </soap:Body>
        </soap:Envelope>');
    $strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>

```

Una vez ejecutado el http_post_data la variable \$strRespuestaServer contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

```

XML
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:HNS="http://tempuri.org/"
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:v1="http://tempuri.org/"
>
<SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
        <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>

```



```
</SOAP-ENV:Header>
<SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
  <v1:WebServService___ConsRecEstadosResponse>
    <v1:strRecEstados>
      <CONSULTA>
        <REC_ESTADOS
          I_ID="--ID DEL ESTADO 1--"
          V_COD_TIPO_EST="--TIPO ESTADO--"
          D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
          V_COD_USU_ALTA="--USUARIO ALTA--"
          B_ULT="--ULTIMO --"
          V_COD_AGE_ALTA="--AGENCIA ALTA--"
          V_COD_REP_ALTA="--REPARTIDOR ALTA--"
        />
        .
        .//DEVOLVERÁ UN BLOQUE POR CADA ESTADO QUE TENGA ESA RECOGIDA
        .
        <REC_ESTADOS
          I_ID="--ID DEL ESTADO N--"
          V_COD_TIPO_EST="--ETC --"
          D_FEC_HORA_ALTA="--ETC en formato mm/dd/yyyy hh:mm:ss--"
          V_COD_USU_ALTA="--ETC --"
          B_ULT="--ULTIMO --"
          V_COD_AGE_ALTA="--ETC --"
          V_COD_REP_ALTA="--ETC --"
        />
      </CONSULTA>
    </v1:strRecEstados>
  </v1:WebServService___ConsRecEstadosResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

1.2.2.3 Método ConsEnvio			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodAgeCargo	Entrada	String	Agencia de cargo.
strCodAgeOri	Entrada	String	Agencia de origen.
strAlbaran	Entrada	String	Albarán.
strEnvio	Salida	String	XML con los datos del envío.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio **"LoginWSService"**.

Uso:

El uso es similar a la anterior.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

string xmlResult = wsWebService.ConsEnvio("agencia-cargo", "agencia-origen", "albaran");
```

Una vez ejecutada la función la variable **xmlResult** contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>
<ENVIOS
  V_COD_AGE_CARGO="--AGENCIA DE CARGO--"
  V_COD_AGE_ORI="--AGENCIA DE ORIGEN--"
  V_ALBARAN="--ALBARÁN--"
  D_FECHA="--FECHA DEL ENVÍO en formato mm/dd/yyyy hh:mm:ss--"
  V_COD_AGE_DES="--AGENCIA DE DESTINO--"
  V_COD_TIPO_SERV="--TIPO DE SERVICIO--"
  V_COD_CLI="--CÓDIGO DEL CLIENTE--"
  V_COD_CLI_DEP="--CÓDIGO DEL DEPARTAMENTO--"
  V_NOM_ORI="--NOMBRE REMITENTE--"
  V_TIPO_VIA_ORI="--TIPO VÍA REMITENTE--"
  V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
```

V_NUM_ORI="--NÚMERO VÍA REMITENTE--"
 V_PISO_ORI="--PISO REMITENTE--"
 V_POB_ORI="--POBLACION REMITENTE--"
 V_CP_ORI="--COD. POSTAL REMITENTE--"
 V_COD_PRO_ORI="--PROVINCIA REMITENTE--"
 V_TLF_ORI="--TLFNO. REMITENTE--"
 V_NOM_DES="--NOMBRE DESTINATARIO--"
 V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
 V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
 V_NUM_DES="--NÚMERO DESTINATARIO--"
 V_PISO_DES="--PISO DESTINATARIO--"
 V_POB_DES="--POBLACIÓN DESTINATARIO--"
 V_CP_DES="--COD. POSTAL DESTINATARIO--"
 V_COD_PRO_DES="--PROVINCIA DESTINATARIO--"
 V_TLF_DES="--TLFNO DESTINATARIO--"
 I_DOC="--NÚMERO DE DOCUMENTOS--"
 I_PAQ="--NÚMERO DE PAQUETES--"
 I_BUL="--NÚMERO DE BULTOS--"
 F_PESO_ORI="--PESO--"
 F_ALTO_ORI="--ALTO--"
 F_ANCHO_ORI="--ANCHO--"
 F_LARGO_ORI="--LARGO--"
 F_PESO_VOL_ORI="--PESO VOLUMÉTRICO--"
 F_PESO_VOLPES="--PESO VOLPES--"
 F_ALTO_VOLPES="--ALTO VOLPES--"
 F_ANCHO_VOLPES="--ANCHO VOLPES--"
 F_LARGO_VOLPES="--LARGO VOLPES--"
 F_PESO_VOL_VOLPES="--PESO VOLUMETRICO VOLPES--"
 F_REEMBOLSO="--REEMBOLSO--"
 F_VALOR="--VALOR--"
 F_ANTICIPO="--ANTICIPO--"
 F_COB_CLI="--COBRO CLIENTE--"
 F_PORTE_DEB="--PORTE DEBIDO--"
 V_OBS="--OBSERVACIONES--"
 D_FEC_ENTR="--FECHA ENTREGA en formato mm/dd/yyyy hh:mm:ss--"
 B_SABADO="--SABADO--"
 B_TECLE_DES="--TECLEADO EN DESTINO--"
 B_RETORNO="--RETORNO--"
 B_GEST_ORI="--GESTIÓN EN ORIGEN--"
 B_GEST_DES="--GESTIÓN EN DESTINO--"
 B_ANULADO="--ANULADO--"
 B_ACUSE="--ACUSE--"
 V_COD_REP="--CÓDIGO REPARTIDOR--"
 V_COD_USU_ALTA="--USUARIO ALTA--"
 V_REF="--REFERENCIA--"
 V_ASOCIADO_RET="--ENVÍO ASOCIADO--"
 V_TIPO_ASOC="--TIPO DE ENVÍO ASOCIADO--"
 V_COD_SAL_RUTA="--CÓDIGO SALIDA RUTA--"
 V_TIPO_ENV="--TIPO DE ENVÍO--"
 F_BASE_IMP="--BASE IMPONIBLE--"
 F_IMPUESTO="--IMPUESTO--"
 D_FEC_HORA_ALTA="--FECHA/HORA ALTA--"
 B_VALIDADO="--VALIDADO--"
 B_CLIENTE="--CLIENTE--"
 B_PORTE_DEB_CLI="--PORTE DEBIDO CLIENTE--"
 V_COD_AGE_ALTA="--AGENCIA DE ALTA--"
 V_COD_CLI_ALTA="--CLIENTE DE ALTA--"
 V_COD_CLI_DEP_ALTA="--DEPARTAMENTO DE ALTA--"
 U_GUID="--GUID DEL ENVÍO--"
 V_PERS_CONTACTO="--PERSONA DE CONTACTO--"
 D_FEC_HORA_ENTR_POD="--FECHA Y HORA DE ENTREGA DEL POD--"
 V_NOM_POD="--NOMBRE DEL REPCEPTOR DEL POD--"
 V_DNI_POD="--DNI DEL REPCEPTOR DEL POD--"
 F_PESO_MAYOR_4="--PESO MAYOR ENTRE PESO ORIGEN Y PESO VOLUMÉTRICO--"
 V_ULT_DIR="--ÚLTIMA DIRECCIÓN--"

```

V_ULT_TIPO_VIA="--ÚLTIMO TIPO DE VÍA--"
V_ULT_NUM="--ÚLTIMO TIPO NÚMERO--"
V_ULT_PISO="--ÚLTIMO PISO--"
V_ULT_POB="--ÚLTIMA POBLACIÓN--"
V_ULT_CP="--ÚLTIMO CÓDIGO POSTAL--"
V_NOM_AGE_DES="--NOMBRE DE AGENCIA DESTINO--"
V_DES_SERV="--DESCRIPCIÓN TIPO SERVICIO--"
V_DES_SAL_RUTA="--DESCRIPCIÓN SALIDA RUTA--"
V_NOM_CLI="--NOMBRE DEL CLIENTE--"
SD_HORA_ENTR_INI="--HORA INICIO ENTREGA--"
SD_HORA_ENTR_FIN="--HORA FIN DE ENTREGA--"
V_NOM_CLI_DEP="--NOMBRE DEL DEPARTAMENTO--"
V_COD_PAIS="--CÓDIGO DEL PAÍS--"
V_NOM_PAIS="--NOMBRE DEL PAÍS--"
V_COD_REC_ASOC="--CÓDIGO DE RECOGIDA ASOCIADO--"
/>
</CONSULTA>

```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```

PHP

<?php
    $strRespuestaServer = http_post_data('http://--ip servidor--/soap',
        '<?xml version="1.0" encoding="utf-8"?>
        <soap:Envelope
            xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema">
            <soap:Header>
                <ROClientIDHeader xmlns="http://tempuri.org/">
                    <ID>--id de la sesión establecida--</ID>
                </ROClientIDHeader>
            </soap:Header>
            <soap:Body>
                <WebServService___ConsEnvio>
                    <strCodAgeCargo>--agencia-cargo--</strCodAgeCargo>
                    <strCodAgeOri>--agencia-origen--</strCodAgeOri>
                    <strAlbaran>--albaran--</strAlbaran>
                </WebServService___ConsEnvio>
            </soap:Body>
        </soap:Envelope>');
    $strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>

```

Una vez ejecutado el http_post_data la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:HNS="http://tempuri.org/"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:v1="http://tempuri.org/">
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService___ConsEnvioResponse>
      <v1:strEnvio>
        <CONSULTA>
          <ENVIOS
            V_COD_AGE_CARGO="--AGENCIA DE CARGO--"
            V_COD_AGE_ORI="--AGENCIA DE ORIGEN--"
            V_ALBARAN="--ALBARÁN--"
            D_FECHA="--FECHA DEL ENVÍO en formato mm/dd/yyyy hh:mm:ss--"
            V_COD_AGE_DES="--AGENCIA DE DESTINO--"
            V_COD_TIPO_SERV="--TIPO DE SERVICIO--"
            V_COD_CLI="--CÓDIGO DEL CLIENTE--"
            V_COD_CLI_DEP="--CÓDIGO DEL DEPARTAMENTO--"
            V_NOM_ORI="--NOMBRE REMITENTE--"
            V_TIPO_VIA_ORI="--TIPO VÍA REMITENTE--"
            V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
            V_NUM_ORI="--NÚMERO VÍA REMITENTE--"
            V_PISO_ORI="--PISO REMITENTE--"
            V_POB_ORI="--POBLACION REMITENTE--"
            V_CP_ORI="--COD. POSTAL REMITENTE--"
            V_COD_PRO_ORI="--PROVINCIA REMITENTE--"
            V_TLF_ORI="--TLFNO. REMITENTE--"
            V_NOM_DES="--NOMBRE DESTINATARIO--"
            V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
            V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
            V_NUM_DES="--NÚMERO DESTINATARIO--"
            V_PISO_DES="--PISO DESTINATARIO--"
            V_POB_DES="--POBLACIÓN DESTINATARIO--"
            V_CP_DES="--COD. POSTAL DESTINATARIO--"
            V_COD_PRO_DES="--PROVINCIA DESTINATARIO--"
            V_TLF_DES="--TLFNO DESTINATARIO--"
            I_DOC="--NÚMERO DE DOCUMENTOS--"
            I_PAQ="--NÚMERO DE PAQUETES--"
            I_BUL="--NÚMERO DE BULTOS--"
            F_PESO_ORI="--PESO--"
            F_ALTO_ORI="--ALTO--"
            F_ANCHO_ORI="--ANCHO--"
            F_LARGO_ORI="--LARGO--"
            F_PESO_VOL_ORI="--PESO VOLUMÉTRICO--"
            F_PESO_VOLPES="--PESO VOLPES--"
            F_ALTO_VOLPES="--ALTO VOLPES--"
            F_ANCHO_VOLPES="--ANCHO VOLPES--"
            F_LARGO_VOLPES="--LARGO VOLPES--"
            F_PESO_VOL_VOLPES="--PESO VOLUMETRICO VOLPES--"
            F_REEMBOLSO="--REEMBOLSO--"
            F_VALOR="--VALOR--"
            F_ANTICIPO="--ANTICIPO--"
            F_COB_CLI="--COBRO CLIENTE--"
            F_PORTE_DEB="--PORTE DEBIDO--"
            V_OBS="--OBSERVACIONES--"
            D_FEC_ENTR="--FECHA ENTREGA en formato mm/dd/yyyy hh:mm:ss--"
          </ENVIOS>
        </CONSULTA>
      </v1:strEnvio>
    </v1:WebServService___ConsEnvioResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```

B_SABADO="--SABADO--"
B_TECLE_DES="--TECLEADO EN DESTINO--"
B_RETORNO="--RETORNO--"
B_GEST_ORI="--GESTIÓN EN ORIGEN--"
B_GEST_DES="--GESTIÓN EN DESTINO--"
B_ANULADO="--ANULADO--"
B_ACUSE="--ACUSE--"
V_COD_REP="--CÓDIGO REPARTIDOR--"
V_COD_USU_ALTA="--USUARIO ALTA--"
V_REF="--REFERENCIA--"
V_ASOCIADO_RET="--ENVÍO ASOCIADO--"
V_TIPO_ASOC="--TIPO DE ENVÍO ASOCIADO--"
V_COD_SAL_RUTA="--CÓDIGO SALIDA RUTA--"
V_TIPO_ENV="--TIPO DE ENVÍO--"
F_BASE_IMP="--BASE IMPONIBLE--"
F_IMPUESTO="--IMPUESTO--"
D_FEC_HORA_ALTA="--FECHA/HORA ALTA--"
B_VALIDADO="--VALIDADO--"
B_CLIENTE="--CLIENTE--"
B_PORTE_DEB_CLI="--PORTE DEBIDO CLIENTE--"
V_COD_AGE_ALTA="--AGENCIA DE ALTA--"
V_COD_CLI_ALTA="--CLIENTE DE ALTA--"
V_COD_CLI_DEP_ALTA="DEPARTAMENTO DE ALTA--"
U_GUID="--GUID DEL ENVÍO--"
V_PERS_CONTACTO="--PERSONA DE CONTACTO--"
D_FEC_HORA_ENTR_POD="--FECHA Y HORA DE ENTREGA DEL POD--"
V_NOM_POD="--NOMBRE DEL REPCEPTOR DEL POD--"
V_DNI_POD="--DNI DEL REPCEPTOR DEL POD--"
F_PESO_MAYOR_4="--PESO MAYOR ENTRE PESO ORIGEN Y PESO VOLUMÉTRICO--"
V_ULT_DIR="--ÚLTIMA DIRECCIÓN--"
V_ULT_TIPO_VIA="--ÚLTIMO TIPO DE VÍA--"
V_ULT_NUM="--ÚLTIMO TIPO NÚMERO--"
V_ULT_PISO="--ÚLTIMO PISO--"
V_ULT_POB="--ÚLTIMA POBLACIÓN--"
V_ULT_CP="--ÚLTIMO CÓDIGO POSTAL--"
V_NOM_AGE_DES="--NOMBRE DE AGENCIA DESTINO--"
V_DES_SERV="--DESCRIPCIÓN TIPO SERVICIO--"
V_DES_SAL_RUTA="--DESCRIPCIÓN SALIDA RUTA--"
V_NOM_CLI="--NOMBRE DEL CLIENTE--"
SD_HORA_ENTR_INI="--HORA INICIO ENTREGA--"
SD_HORA_ENTR_FIN="--HORA FIN DE ENTREGA--"
V_NOM_CLI_DEP="--NOMBRE DEL DEPARTAMENTO--"
V_COD_PAIS="--CÓDIGO DEL PAÍS--"
V_NOM_PAIS="--NOMBRE DEL PAÍS--"
V_COD_REC_ASOC="--CÓDIGO DE RECOGIDA ASOCIADO--"
/>
</CONSULTA>
</v1:strEnvio>
</v1:WebServService___ConsEnvioResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

1.2.2.4 Método ConsRecogida			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCod	Entrada	String	Código de la recogida.
strRecogida	Salida	String	XML con los datos de la recogida.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

El uso es similar a la anterior.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();  
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();  
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;  
  
string xmlResult = wsWebService.ConsRecogida("código-recogida");
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>  
<RECOGIDAS  
  V_COD="--CÓDIGO DE RECOGIDA--"  
  V_COD_AGE_ORI="--AGENCIA DE ORIGEN--"  
  V_COD_AGE_SOL="--AGENCIA QUE SOLICITA--"
```

```

V_COD_AGE_DES="--AGENCIA DE DESTINO--"
V_COD_AGE_CARGO="--AGENCIA DE CARGO--"
D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
D_FEC_REC="--FECHA RECOGIDA en formato mm/dd/yyyy hh:mm:ss--"
SD_HORA_REC_INI="--HORA INICIO RECOGIDA en formato mm/dd/yyyy hh:mm:ss--"
SD_HORA_REC_FIN="--HORA FIN RECOGIDA en formato mm/dd/yyyy hh:mm:ss--"
I_BUL="--BULTOS--"
F_PESO="--PESO--"
F_VALOR="--VALOR--"
F_ANTICIPO="--ANTICIPO--"
V_COD_VEH="--CÓDIGO DEL VEHÍCULO--"
V_NOM_ORI="--NOMBRE REMITENTE--"
V_TIPO_VIA_ORI="--TIPO VÍA REMITENTE--"
V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
V_NUM_ORI="--NÚMERO VÍA REMITENTE--"
V_PISO_ORI="--PISO REMITENTE--"
V_POB_ORI="--POBLACION REMITENTE--"
V_CP_ORI="--COD. POSTAL REMITENTE--"
V_TLF_ORI="--TLFNO. REMITENTE--"
V_COD_PRO_ORI="--PROVINCIA REMITENTE--"
V_NOM_DES="--NOMBRE DESTINATARIO--"
V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
V_NUM_DES="--NÚMERO DESTINATARIO--"
V_PISO_DES="--PISO DESTINATARIO--"
V_POB_DES="--POBLACIÓN DESTINATARIO--"
V_CP_DES="--COD. POSTAL DESTINATARIO--"
V_TLF_DES="--TLFNO DESTINATARIO--"
V_COD_PRO_DES="--PROVINCIA DESTINATARIO--"
V_OBS="--OBSERVACIONES--"
V_COD_CLI="--CÓDIGO DE CLIENTE--"
V_COD_CLI_DEP="--CÓDIGO DE DEPARTAMENTO--"
V_TIPO_REC="--TIPO RECOGIDA--"
V_PERS_CONTACTO="--PERSONA CONTACTO--"
B_AUT_KM="--AUTORIZA KILÓMETROS--"
V_COD_TIPO_SERV="--TIPO SERVICIO--"
B_SABADO="--SABADO--"
V_COD_REP="--CÓDIGO REPARTIDOR--"
V_COD_USU_ALTA="--USUARIO ALTA--"
V_COD_ENV="--CÓDIGO DE ENVÍO--"
SD_HORA_REC_INI_TARDE="--HORA INICIO RECOGIDA TARDE en formato mm/dd/yyyy hh:mm:ss--"
SD_HORA_REC_FIN_TARDE="--HORA FIN RECOGIDA TARDE en formato mm/dd/yyyy hh:mm:ss--"
V_COD_AGE_ALTA="--AGENCIA ALTA--"
V_COD_CLI_ALTA="--CLIENTE ALTA--"
V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
V_COD_ENV="--CÓDIGO DE ENVÍO--"
/>
</CONSULTA>

```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

PHP

```
<?php
    $strRespuestaServer = http_post_data('http://--ip servidor--/soap',
        '<?xml version="1.0" encoding="utf-8"?>
            <soap:Envelope
                xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
                xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                xmlns:xsd="http://www.w3.org/2001/XMLSchema">
                <soap:Header>
                    <ROClientIDHeader xmlns="http://tempuri.org/">
                        <ID>--id de la sesión establecida--</ID>
                    </ROClientIDHeader>
                </soap:Header>
                <soap:Body>
                    <WebServService___ConsRecogida>
                        <strCodRec>--Código-recogida--</strCodRec>
                    </WebServService___ConsRecogida>
                </soap:Body>
            </soap:Envelope>');
    $strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>
```

Una vez ejecutado el http_post_data la variable \$strRespuestaServer contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:HNS="http://tempuri.org/"
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:v1="http://tempuri.org/">
    <SOAP-ENV:Header>
        <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
            <ID>--ID SESIÓN--</ID>
        </ROClientIDHeader>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
        <v1:WebServService___ConsRecogidaResponse>
            <v1:strRecogida>
                <CONSULTA>
                    <RECOGIDAS
                        V_COD="--CÓDIGO DE RECOGIDA--"
                        V_COD_AGE_ORI="--AGENCIA DE ORIGEN--"
                        V_COD_AGE_SOL="--AGENCIA QUE SOLICITA--"
                        V_COD_AGE_DES="--AGENCIA DE DESTINO--"
                        V_COD_AGE_CARGO="--AGENCIA DE CARGO--"
                        D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
                        D_FEC_REC="--FECHA RECOGIDA en formato mm/dd/yyyy hh:mm:ss--"
                        SD_HORA_REC_INI="--HORA INICIO RECOGIDA en formato mm/dd/yyyy hh:mm:ss--"
                        SD_HORA_REC_FIN="--HORA FIN RECOGIDA en formato mm/dd/yyyy hh:mm:ss--"
                        I_BUL="--BULTOS--"
                        F_PESO="--PESO--"
                        F_VALOR="--VALOR--"
                        F_ANTICIPO="--ANTICIPO--"
                        V_COD_VEH="--CÓDIGO DEL VEHÍCULO--"
```

```

V_NOM_ORI="--NOMBRE REMITENTE--"
V_TIPO_VIA_ORI="--TIPO VÍA REMITENTE--"
V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
V_NUM_ORI="--NÚMERO VÍA REMITENTE--"
V_PISO_ORI="--PISO REMITENTE--"
V_POB_ORI="--POBLACION REMITENTE--"
V_CP_ORI="--COD. POSTAL REMITENTE--"
V_TLF_ORI="--TLFNO. REMITENTE--"
V_COD_PRO_ORI="--PROVINCIA REMITENTE--"
V_NOM_DES="--NOMBRE DESTINATARIO--"
V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
V_NUM_DES="--NÚMERO DESTINATARIO--"
V_PISO_DES="--PISO DESTINATARIO--"
V_POB_DES="--POBLACIÓN DESTINATARIO--"
V_CP_DES="--COD. POSTAL DESTINATARIO--"
V_TLF_DES="--TLFNO DESTINATARIO--"
V_COD_PRO_DES="--PROVINCIA DESTINATARIO--"
V_OBS="--OBSERVACIONES--"
V_COD_CLI="--CÓDIGO DE CLIENTE--"
V_COD_CLI_DEP="--CÓDIGO DE DEPARTAMENTO--"
V_TIPO_REC="--TIPO RECOGIDA--"
V_PERS_CONTACTO="--PERSONA CONTACTO--"
B_AUT_KM="--AUTORIZA KILÓMETROS--"
V_COD_TIPO_SERV="--TIPO SERVICIO--"
B_SABADO="--SABADO--"
V_COD_REP="--CÓDIGO REPARTIDOR--"
V_COD_USU_ALTA="--USUARIO ALTA--"
V_COD_ENV="--CÓDIGO DE ENVÍO--"
SD_HORA_REC_INI_TARDE="--HORA INICIO RECOGIDA TARDE en formato mm/dd/yyyy
hh:mm:ss--"
SD_HORA_REC_FIN_TARDE="--HORA FIN RECOGIDA TARDE en formato mm/dd/yyyy
hh:mm:ss--"
V_COD_AGE_ALTA="--AGENCIA ALTA--"
V_COD_CLI_ALTA="--CLIENTE ALTA--"
V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
V_COD_ENV="--CÓDIGO DE ENVÍO--"
/>
</CONSULTA>
</v1:strRecogida>
</v1:WebServService___ConsRecogidaResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

1.2.2.5 Método GrabaEnvio10

PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodAgeCargo	Entrada	String	Agencia de cargo.
strCodAgeOri	Entrada	String	Agencia de origen.
dtFecha	Entrada	Datetime	Fecha del envío.
strCodAgeDes	Entrada	String	Agencia de destino.
strCodTipoServ	Entrada	String	Tipo de servicio.
strCodCli	Entrada	String	Código de cliente.
strCodCliDep	Entrada	String	Código de departamento.
strNomOri	Entrada	String	Nombre remitente.
strTipoViaOri	Entrada	String	Tipo de vía del remitente (C = calle, PZA = plaza, etc.).
strDirOri	Entrada	String	Nombre de vía del remitente.
strNumOri	Entrada	String	Número de casa del remitente.
strPisoOri	Entrada	String	Número de piso del remitente.
strPobOri	Entrada	String	Población del remitente.
strCPori	Entrada	String	Código postal del remitente.
strCodProOri	Entrada	String	Código provincia del remitente.
strTlfOri	Entrada	String	Teléfono del remitente.
strNomDes	Entrada	String	Nombre de destinatario.
strTipoViaDes	Entrada	String	Tipo de vía del destinatario.
strDirDes	Entrada	String	Nombre de vía del destinatario.
strNumDes	Entrada	String	Número de casa del destinatario.
strPisoDes	Entrada	String	Número de piso del destinatario.
strPobDes	Entrada	String	Población del destinatario.
strCPDes	Entrada	String	Código postal del destinatario (para envíos internacionales puede ser hasta de 10 dígitos, para envíos no internacionales de 5 dígitos)
strCodProDes	Entrada	String	Código de provincia del destinatario.
strTlfDes	Entrada	String	Teléfono del destinatario.
intDoc	Entrada	Integer	Número de documentos.
intPaq	Entrada	Integer	Número de paquetes.
dPesoOri	Entrada	Double	Peso en origen.
dAltoOri	Entrada	Double	Alto del envío en origen.
dAnchoOri	Entrada	Double	Ancho del envío en origen.
dLargoOri	Entrada	Double	Largo del envío en origen.
dReembolso	Entrada	Double	Reembolso.
dValor	Entrada	Double	Valor.
dAnticipo	Entrada	Double	Anticipo.
dCobCli	Entrada	Double	Cobro cliente.
strObs	Entrada	String	Observaciones.
boSabado	Entrada	Boolean	Entrega en sábado.
boRetorno	Entrada	Boolean	Retorno.
boGestOri	Entrada	Boolean	Gestión en origen.
boGestDes	Entrada	Boolean	Gestión en destino.
boAnulado	Entrada	Boolean	Anulado.
boAcuse	Entrada	Boolean	Acuse.
strRef	Entrada	String	Referencia del envío.
strCodSalRuta	Entrada	String	Código de salida de ruta.
dBaselmp	Entrada	Double	Base imponible.

dImpuesto	Entrada	Double	Impuesto.
boPorteDebCli	Entrada	Boolean	Indica (si es de cliente final) si es o no porte debido.
strPersContacto	Entrada	String	Persona de contacto.
boDesSMS	Entrada	Boolean	Enviar SMS al destinatario.
boDesEmail	Entrada	Boolean	Enviar email al destinatario.
strDesMoviles	Entrada	String	Móviles destinatario.
strDesDirEmails	Entrada	String	Direcciones email del destinatario.
dtHoraEntrIni	Entrada	DateTime	Hora de entrega mínima.
dtHoraEntrFin	Entrada	DateTime	Hora de entrega máxima.
strCodPais	Entrada	String	Código de país.
dvalorFactInt	Entrada	Double	Valor factura int.
boInsert	Entrada	Boolean	Indica si se está insertando un nuevo envío
strAlbaran	Entrada	String	En el caso de que se vaya a editar un envío este parámetro indica el número de albarán. Si se está insertando este parámetro no se tiene en cuenta.
strConcAgru	Entrada	String	Concepto de agrupación. Si está vacío no se agrupa, en caso contrario busca un envío que coincide para agrupar.
strAlbaranOut	Salida	String	Albarán.
dPesoVolOriOut	Salida	Double	Peso volumétrico origen devuelto.
dPesoVolpesOut	Salida	Double	Peso volpes devuelto.
dAltoVolpesOut	Salida	Double	Alto volpes devuelto.
dAnchoVolpesOut	Salida	Double	Ancho volpes devuelto.
dLargoVolpesOut	Salida	Double	Largo volpes devuelto.
dPesoVolVolpesOut	Salida	Double	Peso volumétrico volpes devuelto.
dtFecEntrOut	Salida	Datetime	Fecha de entrega.
strTipoEnvOut	Salida	String	Tipo de envío.
dtFecHoraAltaOut	Salida	Datetime	Fecha/Hora de alta.
dKmsManOut	Salida	Double	Número manual de kilómetros.
boTecleDesOut	Salida	Boolean	Tecleado en destino.
strCodAgeDes	Salida	String	Código agencia de destino.
strCodProDesOut	Salida	String	Código provincia de destino.
dPorteDebOut	Salida	Double	Porte debido.
strGuidOut	Salida	Guid	GUID del envío.
strCodSalRutaOut	Salida	String	Salida ruta del envío.
strCPDesOut	Salida	String	CP destinatario.
strCodPaisOut	Salida	String	Código de país.
boEnvioAgrupadoOut	Salida	Boolean	Indica si el envío ha sido agrupado.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

El uso es similar a la anterior.

Ejemplo:

```
C#
/*peso-volumetrico*/ /*peso-volpes*/ /*alto-volpes*/ /*ancho-volpes*/ /*largo-volpes*/
double dPesoVolOriOut, dPesoVolpesOut, dAltoVolpesOut, dAnchoVolpesOut, dLargoVolpesOut;
```

```

/*peso-volumetrico-volpes*/ /*kilometros*/ /*porte-debido*/ /*Valor-factura-interna*/
double dPesoVolVolpesOut, dKmsManOut, dPorteDebOut, dValorFactInt;

/*albaran-env*/ /*tipo-envio*/ /*cod-age-dest*/ /*cod-provin-dest*/ /*GUID env*/ /*salida-ruta*/
string strAlbaranOut, strTipoEnvOut, strCodAgeDesOut, strCodProDesOut, strGuidOut, strCodSalRutaOut;

/*CP-destinatario*/ /*código-país*/
string strCPDesOut, strCodPaisOut;

/*fecha-entrega*/ /*fecha-alta*/
DateTime dtFecEntrOut, dtFecHoraAltaOut;

/*teclado-destino*/ /*envío-agrupado*/
bool boTecleDesOut, boEnvioAgrupadoOut;

WebService.WebService wsWebService = new WebService.WebService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

try
{
    strAlbaranOut = wsWebService.GrabarEnvio10("agencia-cargo", "agencia-ori", "fecha-env",
        "agencia-des", "tipo-serv", "cod-cliente", "cod-departamento", "nombre-remit", "tipo-via-remit",
        "nombre-via-remit", "num-casa-remit", "piso-remit", "pobla-remit", "C.P-remit", "cod-provincia-ori",
        "tlfno-remit", "nombre-dest", "tipo-via-dest", "nombre-via-dest", "num-casa-dest", "piso-dest",
        "pobla-dest", "C.P-dest", "cod-provincia-dest", "tlfno-dest", documentos, paquetes, peso-ori,
        alto-origen, ancho-origen, largo-origen, reembolso, valor, anticipo, cobro-cliente, "observaciones",
        sabado, retorno, gestion-origen, gestion-destino, anulado, acuse, "referencia",
        "cod-sal-ruta", base-imponible, impuesto, porte-debido, "persona-contacto", enviar-SMS-destino,
        enviar-email-destino, "moviles-destino", "emails-destino", "hora-entr-mín", "hora-entr-máx",
        "código-país", "insertando", "albaran", "conc-agrupa",
        out dPesoVolOriOut, out dPesoVolpesOut, out dAltoVolpesOut,
        out dAnchoVolpesOut, out dLargoVolpesOut, out dPesoVolVolpesOut, out dtFecEntrOut,
        out strTipoEnvOut, out dtFecHoraAltaOut, out dKmsManOut, out boTecleDesOut,
        out strCodAgeDesOut, out strCodProDesOut, out dPorteDebOut, out strGuidOut,
        out strCodSalRutaOut, out strCPDesOut, out strCodPaisOut, out boEnvioAgrupadoOut);
}
catch (Exception eExcepcion)
{
    MessageBox.Show(eExcepcion.Message);
}

```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```

PHP

<?php
    $strRespuestaServer = http_post_data('http://--ip servidor--/soap',
    '<?xml version="1.0" encoding="utf-8"?>
    <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        <soap:Header>
            <ROClientIDHeader xmlns="http://tempuri.org/">

```

```

<ID>--id de la sesión establecida--</ID>
</ROClientIDHeader>
</soap:Header>
<soap:Body>
<WebServService___GrabaEnvio10 xmlns="http://tempuri.org/">
  <strCodAgeCargo>--agencia-cargo--</strCodAgeCargo>
  <strCodAgeOri>--agencia-origen--</strCodAgeOri>
  <dtFecha>--fecha-envio en formato yyyy/mm/dd--</dtFecha>
  <strCodAgeDes>--agencia-dest--</strCodAgeDes>
  <strCodTipoServ>--tipo-servicio--</strCodTipoServ>
  <strCodCli>--codigo-cliente--</strCodCli>
  <strCodCliDep>--codigo-departamento--</strCodCliDep>
  <strNomOri>--nombre-remitente--</strNomOri>
  <strTipoViaOri>--tipo-via-remitente--</strTipoViaOri>
  <strDirOri>--nombre-via-remitente--</strDirOri>
  <strNumOri>--numero-casa--</strNumOri>
  <strPisoOri>--piso-remitente--</strPisoOri>
  <strPobOri>--poblacion-remitente--</strPobOri>
  <strCPOri>--C.P.-remitente--</strCPOri>
  <strCodProOri>--cod-provincia-remitente--</strCodProOri>
  <strTlfOri>--telefono-remitente--</strTlfOri>
  <strNomDes>--nombre-destinatario--</strNomDes>
  <strTipoViaDes>--tipo-via-destinatario--</strTipoViaDes>
  <strDirDes>--nombre-via-destinatario--</strDirDes>
  <strNumDes>--numero-casa-dest--</strNumDes>
  <strPisoDes>--piso-destinatario--</strPisoDes>
  <strPobDes>--poblacion-dest--</strPobDes>
  <strCPDes>--C.P. Dest--</strCPDes>
  <strCodProDes>--cod-provincia-dest--</strCodProDes>
  <strTlfDes>--telefono-des--</strTlfDes>
  <intDoc>--numero-documentos--</intDoc>
  <intPaq>--numero-paquetes--</intPaq>
  <dPesoOri>--peso-origen--</dPesoOri>
  <dAltoOri>--alto-origen--</dAltoOri>
  <dAnchoOri>--ancho-origen--</dAnchoOri>
  <dLargoOri>--largo-origen--</dLargoOri>
  <dReembolso>--reembolso--</dReembolso>
  <dValor>--valor--</dValor>
  <dAnticipo>--anticipo--</dAnticipo>
  <dCobCli>--cobro-cliente--</dCobCli>
  <strObs>--observaciones--</strObs>
  <boSabado>--sabado--</boSabado>
  <boRetorno>--retorno--</boRetorno>
  <boGestOri>--gestion-origen--</boGestOri>
  <boGestDes>--gestion-destino--</boGestDes>
  <boAnulado>--anulado--</boAnulado>
  <boAcuse>--acuse--</boAcuse>
  <strRef>--referencia--</strRef>
  <strCodSalRuta>--codigo-sal-ruta--</strCodSalRuta>
  <dBaselmp>--base-imponible--</dBaselmp>
  <dImpuesto>--impuesto--</dImpuesto>
  <boPorteDebCli>--porte-debido-cliente--</boPorteDebCli>
  <strPersContacto>--persona-contacto--</strPersContacto>
  <boDesSMS>--enviar-SMS-destinatario--</boDesSMS>
  <boDesEmail>--enviar-email-destinatario--</boDesEmail>
  <strDesMoviles>--moviles-destinatario--</strDesMoviles>
  <strDesDirEmails>--emails-destinatario--</strDesDirEmails>
  <dtHoraEntrIni>--hora-entr-mín--</dtHoraEntrIni>
  <dtHoraEntrFin>--hora-entr-máx--</dtHoraEntrFin>
  <strCodPais>--código-país--</strCodPais>
  <dValorFactInt>--Valor-fact-int--</dValorFactInt>
  <bolInsert>--insertando--</bolInsert>
  <strAlbaran>--albaran--</strAlbaran>
  <strConcAgru>--conc-agrupa--</strConcAgru>
</WebServService___GrabaEnvio10>

```

```

</soap:Body>
</soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>

```

Una vez ejecutado el `http_post_data` la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```

<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/">
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService__GrabaEnvio10Response>
      <v1:strAlbaranOut>--OUT ALBARAN--</v1:strAlbaranOut>
      <v1:dPesoVolOriOut>--OUT PESO VOLUMETRICO ORIGEN--</v1:dPesoVolOriOut>
      <v1:dPesoVolpesOut>--OUT PESO VOLPES--</v1:dPesoVolpesOut>
      <v1:dAltoVolpesOut>--OUT ALTO VOLPES--</v1:dAltoVolpesOut>
      <v1:dAnchoVolpesOut>--OUT ANCHO VOLPES--</v1:dAnchoVolpesOut>
      <v1:dLargoVolpesOut>--OUT LARGO VOLPES--</v1:dLargoVolpesOut>
      <v1:dPesoVolVolpesOut>--OUT PESO VOLUMETRICO VOLPES--</v1:dPesoVolVolpesOut>
      <v1:dtFecEntrOut>--OUT FECHA ENTREGA en formato yyyy/mm/dd--</v1:dtFecEntrOut>
      <v1:strTipoEnvOut>--OUT TIPO ENVIO--</v1:strTipoEnvOut>
      <v1:dtFecHoraAltaOut>--OUT FECHA HORA ALTA en formato yyyy/mm/dd hh:mm:ss--
    </v1:dtFecHoraAltaOut>
      <v1:dKmsManOut>--OUT KILOMETROS--</v1:dKmsManOut>
      <v1:boTecleDesOut>--OUT TECLEADO DESTINO--</v1:boTecleDesOut>
      <v1:strCodAgeDesOut>--OUT AGENCIA DESTINO--</v1:strCodAgeDesOut>
      <v1:strCodProDesOut>--OUT CODIGO PROVINCIA DESTINO--</v1:strCodProDesOut>
      <v1:dPorteDebOut>--OUT PORTE DEBIDO--</v1:dPorteDebOut>
      <v1:strGuidOut>--OUT GUID ENVIO--</v1:strGuidOut>
      <v1:strCodSalRutaOut>--OUT SALIDA RUTA ENVIO--</v1:strCodSalRutaOut>
      <v1:strCPDesOut>--OUT CP DESTINATARIO--</v1:strCPDesOut>
      <v1:strCodPaisOut>--OUT CÓDIGO PAÍS--</v1:strCodPaisOut>
      <v1:boEnvioAgrupadoOut>--OUT ENVIO AGRUPADO--</v1:boEnvioAgrupadoOut>
    </v1:WebServService__GrabaEnvio10Response>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Excepciones:

Esta función devuelve las siguientes excepciones:

CÓDIGO	DESCRIPCIÓN
--------	-------------

1	El usuario no existe.
2	El usuario no tiene permisos de modificación.
3	El usuario no tiene suficientes permisos.
4	La fecha no es válida.
5	La fecha está fuera del rango permitido.
6	El tipo de servicio no existe o está inactivo.
7	La agencia de cargo no existe o está inactiva.
8	La agencia de origen no existe o está inactiva.
9	La agencia de destino no existe o está inactiva.
10	El nº de albarán ha de tener 10 dígitos.
11	El primer dígito en este nº de albarán está reservado para envíos sin conexión.
12	El nº de albarán indicado ya se encuentra en uso.
13	El cliente no pertenece a la agencia de cargo.
14	El departamento no pertenece al cliente.
15	El cliente no autoriza envíos a su cargo.
16	La salida de ruta no existe.
17	Error en la admisión del envío.
18	No está permitido dar de alta envíos sin conexión.
19	El código postal destino es nulo o no válido.
20	La fecha de entrega no es válida.
21	El usuario sólo puede importar envíos cuya agencia de cargo u origen sea la suya.
22	El envío ha de tener mínimo 1 bulto.
23	No se pueden introducir valores numéricos negativos.
24	La agencia no admite valor asegurado en envíos locales.
25	Los campos obligatorios deben tener valor.
26	El tipo de servicio no es admisible para el cliente seleccionado.
27	El nº de albarán no puede terminar en 000.
28	El tipo de servicio no es admisible para la agencia de destino seleccionada.
29	Agencia de origen no permitida
30	La agencia de cargo ha llegado al límite de envíos mensuales permitidos
31	Ya existe un envío con la misma agencia de cargo, origen y el número de albarán especificado en la configuración de números de albarán para clientes finales. Debe modificar dicha configuración para poder grabar envíos
32	No tiene permisos para grabar envíos con el código de cliente indicado
33	El país no existe

Para saber cómo controlar las excepciones devueltas por favor consulte el apartado 1.2.3, Excepciones.

1.2.2.6 Método GrabaRecogida2			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCod	Entrada	String	Código recogida.
strCodAgeOri	Entrada	String	Código de la agencia de origen.
strCodAgeSol	Entrada	String	Código de la agencia que solicita la recogida.
strCodAgeDes	Entrada	String	Código de la agencia de destino.
strCodAgeCargo	Entrada	String	Código de la agencia de cargo.
dtFecRec	Entrada	Datetime	Fecha de recogida.
dtHoraRecIni	Entrada	Datetime	Inicio de hora de recogida.
dtHoraRecIniTarde	Entrada	Datetime	Inicio de hora de recogida por la tarde.
dtHoraRecFin	Entrada	Datetime	Fin de hora de recogida.
dtHoraRecFinTarde	Entrada	Datetime	Fin de hora de recogida por la tarde.
intBul	Entrada	Integer	Número de bultos.
dPeso	Entrada	Double	Peso de la recogida.
dValor	Entrada	Double	Valor.
dAnticipo	Entrada	Double	Anticipo.
strCodVeh	Entrada	String	Vehículo.
strNomOri	Entrada	String	Nombre remitente.
strTipoViaOri	Entrada	String	Tipo de vía del remitente (C = calle, PZA = plaza, etc.).
strDirOri	Entrada	String	Nombre de vía del remitente.

strNumOri	Entrada	String	Número de casa del remitente.
strPisoOri	Entrada	String	Número de piso del remitente.
strPobOri	Entrada	String	Población del remitente.
strCPOri	Entrada	String	Código postal del remitente.
strTlfOri	Entrada	String	Teléfono del remitente.
strCodProOri	Entrada	String	Código provincia del remitente.
strNomDes	Entrada	String	Nombre de destinatario.
strTipoViaDes	Entrada	String	Tipo de vía del destinatario.
strDirDes	Entrada	String	Nombre de vía del destinatario.
strNumDes	Entrada	String	Número de casa del destinatario.
strPisoDes	Entrada	String	Número de piso del destinatario.
strPobDes	Entrada	String	Población del destinatario.
strCPDes	Entrada	String	Código postal del destinatario.
strTlfDes	Entrada	String	Teléfono del destinatario.
strCodProDes	Entrada	String	Código de provincia del destinatario.
strObs	Entrada	String	Observaciones.
strCodCli	Entrada	String	Código de cliente.
strCodCliDep	Entrada	String	Código de departamento.
strPersContacto	Entrada	String	Persona de contacto.
boAutKM	Entrada	Boolean	Autoriza kilómetros.
strCodTipoServ	Entrada	String	Código de tipo servicio.
boSabado	Entrada	Boolean	Recogida en sábado.
strCodRep	Entrada	String	Código de repartidor.
strCodEnv	Entrada	String	Código de envío.
strRef	Entrada	String	Referencia.
strTipoRecOld	Entrada	String	Tipo de recogida anterior.
strObsDes	Entrada	String	Observaciones de destino.
dReembolso	Entrada	Double	Reembolso.
dCobCli	Entrada	Double	Cobro cliente.
dImpuesto	Entrada	Double	Impuesto.
dBaselmp	Entrada	Double	Base imponible.
boAcuse	Entrada	Boolean	Acuse.
boRetorno	Entrada	Boolean	Retorno.
boGestOri	Entrada	Boolean	Gestión en origen.
boGestDes	Entrada	Boolean	Gestión en destino.
boRemSms	Entrada	Boolean	Avisar al remitente por sms.
boRemEmail	Entrada	Boolean	Avisar al remitente por email.
strRemMoviles	Entrada	String	Móviles para avisos de remitente.
strRemDirEmails	Entrada	String	Direcciones email para avisos de remitente.
boRemAlta	Entrada	Boolean	Avisar al remitente al dar de alta la recogida.
boRemInci	Entrada	Boolean	Avisar al remitente al generarse una incidencia.
boRemResInci	Entrada	Boolean	Avisar al remitente al resolverse una incidencia.
boRemRep	Entrada	Boolean	Avisar al remitente al poner en reparto la recogida.
strCodOut	Salida	String	Código de recogida (cuando no se indica al grabar).
strTipoRecOut	Salida	String	Tipo de recogida.
dtFechaHoraAltaOut	Salida	Datetime	Fecha de alta.
strCodAgeOriOut	Salida	String	Código de agencia de origen.
strCodProOriOt	Salida	String	Código de provincia de origen.
strCodAgeDesOut	Salida	String	Código de agencia de destino.
strCodProDesOut	Salida	String	Código de provincia de destino.
fPorteDebOut	Salida	Double	Porte debido.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

El uso es similar a la anterior.

Ejemplo:

```
C#
    /*cod-rec*/ /*tipo-recogida*/ /*cod-age-ori*/ /*cod-provincia-ori*/
string strCodOut, strTipoRecOut, strCodAgeOriOut, strCodProOriOut;

    /*cod-agencia-dest*/ /*cod-provincia-dest*/
string strCodAgeDesOut, strCodProDesOut;

    /*fecha-hora-alta*/
DateTime dtFecHoraAltaOut;

    /*porte-debido*/
Double dPorteDebOut;

WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = strSesion;

try
{
    strCodOut = wsWebService.GrabarRecogida2("codigo-recogida", "agencia-origen",
        "agencia-solicitada", "agencia-destino", "agencia-cargo", "fecha-recogida", "hora-recogida-ini",
        "hora-recogida-ini-tarde", "hora-recogida-fin", "hora-recogida-fin-tarde", bultos, peso, valor,
        anticipo, "codigo-vehiculo", "nombre-origen", "tipo-via-origen", "nombre-via-origen", "numero",
        "piso", "poblacion-origen", "codigo-postal-origen", "telefono-origen", "codigo-provincia-origen",
        "nombre-dest", "tipo-via-dest", "nombre-via-dest", "numero-dest", "piso-dest", "poblacion-dest",
        "codigo-postal-dest", "telefono-dest", "cod-provincia-dest", "observaciones", "codigo-cliente",
        "codigo-departamento", "persona-contacto", autoriza-km, "codigo-tipo-serv", recogida-sabado,
        "codigo-repartidor", "codigo-envio", "referencia", "tipo-recogida-anterior",
        "observaciones-destino", reembolso, cobro-cliente, impuesto, base-imponible, acuse, retorno,
        gestion-origen, gestion-destino, AvisarRemSMS, AvisarRemEmail, "MovilesRem",
        "EmailsRem", AvisarRemAlta, AvisarRemInci, AvisarRemResInci,
        AvisarRemRecReparto, out strTipoRecOut, out dtFecHoraAltaOut, out strCodAgeOriOut,
        out strCodProOriOut, out strCodAgeDesOut, out strCodProDesOut, out dPorteDebOut);
}
catch (Exception eExcepcion)
{
    MessageBox.Show(eExcepcion.Message);
}
```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```
PHP

<?php
    $strRespuestaServer = http_post_data('http://--ip servidor--/soap',
    '<?xml version="1.0" encoding="utf-8"?>')
```

```

<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Header>
    <ROClientIDHeader xmlns="http://tempuri.org/">
      <ID>--ID-SESION--</ID>
    </ROClientIDHeader>
  </soap:Header>
  <soap:Body>
    <WebServService__GrabaRecogida2 xmlns="http://tempuri.org/">
      <strCod>--codigo-recogida--</strCod>
      <strCodAgeOri>--agencia-origen--</strCodAgeOri>
      <strCodAgeSol>--agencia-solicita--</strCodAgeSol>
      <strCodAgeDes>--agencia-destino--</strCodAgeDes>
      <strCodAgeCargo>--agencia-cargo--</strCodAgeCargo>
      <dtFecRec>--fecha-recogida en formato yyyy/mm/dd--</dtFecRec>
      <dtHoraRecIni>--hora-inicio-rec en formato yyyy/mm/dd hh:mm:ss--</dtHoraRecIni>
      <dtHoraRecIniTarde>--hora-inicio-rec-tarde yyyy/mm/dd hh:mm:ss--</dtHoraRecIniTarde>
      <dtHoraRecFin>--hora-fin-rec en formato yyyy/mm/dd hh:mm:ss--</dtHoraRecFin>
      <dtHoraRecFinTarde>--hora-fin-rec-tarde en formato yyyy/mm/dd hh:mm:ss--</dtHoraRecFinTarde>
      <intBul>--bultos--</intBul>
      <dPeso>--peso--</dPeso>
      <dValor>--valor--</dValor>
      <dAnticipo>--anticipo--</dAnticipo>
      <strCodVeh>--codigo-vehiculo--</strCodVeh>
      <strNomOri>--nombre-origen--</strNomOri>
      <strTipoViaOri>--tipo-via-origen--</strTipoViaOri>
      <strDirOri>--nombre-via-origen--</strDirOri>
      <strNumOri>--numero-origen--</strNumOri>
      <strPisoOri>--piso--</strPisoOri>
      <strPobOri>--poblacion-origen--</strPobOri>
      <strCPOri>--cod-postal-ori--</strCPOri>
      <strTlfOri>--telefono-origen--</strTlfOri>
      <strCodProOri>--codigo-provincia-origen--</strCodProOri>
      <strNomDes>--nombre-destino--</strNomDes>
      <strTipoViaDes>--tipo-via-destino--</strTipoViaDes>
      <strDirDes>--nombre-via-destino--</strDirDes>
      <strNumDes>--numero-destino--</strNumDes>
      <strPisoDes>--piso--</strPisoDes>
      <strPobDes>--poblacion-destino--</strPobDes>
      <strCPDes>--cod-postal-destino--</strCPDes>
      <strTlfDes>--telefono-destino--</strTlfDes>
      <strCodProDes>--cod-provincia-destino--</strCodProDes>
      <strObs>--observaciones--</strObs>
      <strCodCli>--codigo-cliente--</strCodCli>
      <strCodCliDep>--codigo-departamento--</strCodCliDep>
      <strPersContacto>--persona-contacto--</strPersContacto>
      <boAutKM>--autoriza-km--</boAutKM>
      <strCodTipoServ>--tipo-servicio--</strCodTipoServ>
      <boSabado>--recogida-sabado--</boSabado>
      <strCodRep>--codigo-repartidor--</strCodRep>
      <strCodEnv>--codigo-envio--</strCodEnv>
      <strRef>--referencia--</strRef>
      <strTipoRecOld>--tipo-recogida-anterior--</strTipoRecOld>
      <strObsDes>--observaciones-destino--</strObsDes>
      <dReembolso>--reembolso--</dReembolso>
      <dCobCli>--cobro-cliente--</dCobCli>
      <dImpuesto>--impuesto--</dImpuesto>
      <dBaselmp>--base-imponible--</dBaselmp>
      <boAcuse>--acuse--</boAcuse>
      <boRetorno>--retorno--</boRetorno>
      <boGestOri>--gestion-origen--</boGestOri>
      <boGestDes>--gestion-destino--</boGestDes>
      <boRemSms>--gestion-destino--</boRemSms>
    </WebServService__GrabaRecogida2>
  </soap:Body>
</soap:Envelope>
  
```

```

<boRemEmail>--gestion-destino--</boRemEmail>
<strRemMoviles>--gestion-destino--</boGestDes>
<strRemDirEmails>--gestion-destino--</strRemDirEmails>
<boRemAlta>--gestion-destino--</boRemAlta>
<boRemInci>--gestion-destino--</boRemInci>
<boRemResInci>--gestion-destino--</boRemResInci>
<boRemRep>--gestion-destino--</boRemRep>

</WebServService___GrabaRecogida>
</soap:Body>
</soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>

```

Una vez ejecutado el `http_post_data` la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

```

XML
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/">
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID-SESION--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService___GrabaRecogidaResponse>
      <v1:strCodOut>--OUT-CODIGO-RECOGIDA--</v1:strCodOut>
      <v1:strTipoRecOut>--OUT-TIPO-RECOGIDA--</v1:strTipoRecOut>
      <v1:dtFecHoraAltaOut>--OUT-FECHA-HORA-ALTA en formato yyyy/mm/dd hh:mm:ss--
    </v1:dtFecHoraAltaOut>
      <v1:strCodAgeOriOut>--OUT-CODIGO-AGENCIA-ORI--</v1:strCodAgeOriOut>
      <v1:strCodProOriOut>--OUT-CODIGO-PROVINCIA-ORI--</v1:strCodProOriOut>
      <v1:strCodAgeDesOut>--OUT-CODIGO-AGENCIA-DEST--</v1:strCodAgeDesOut>
      <v1:strCodProDesOut>--OUT-CODIGO-PROVINCIA-DEST--</v1:strCodProDesOut>
      <v1:fPorteDebOut>--OUT-PORTE-DEB --</v1:fPorteDebOut>
    </v1:WebServService___GrabaRecogidaResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Excepciones:

Esta función devuelve las siguientes excepciones:

CÓDIGO	DESCRIPCIÓN
-2	Error al intentar realizar la canalización en destino.
-1	Error al intentar realizar la canalización en origen.
1	El usuario no existe.
2	El usuario no tiene suficientes permisos para modificar esta recogida.

3	El usuario no tiene suficientes permisos para realizar esta acción.
4	La fecha no es válida.
5	La fecha está fuera del rango permitido.
6	El tipo de servicio no existe o está inactivo.
7	La agencia de origen no existe o está inactiva.
8	La agencia solicitante no existe o está inactiva.
9	La agencia de destino no existe o está inactiva.
10	La agencia de cargo no existe o está inactiva.
11	El cliente no pertenece a la agencia de cargo.
12	El cliente no autoriza recogidas a su cargo.
13	El código del envío asociado a la recogida no es válido.
14	El departamento no pertenece al cliente.
15	Error en la admisión de la recogida.
16	El código postal origen es nulo o no válido.
17	El código postal destino es nulo o no válido.
18	No se pueden introducir valores numéricos negativos.
19	Los campos obligatorios deben tener valor.
20	El tipo de servicio no es admisible para el cliente seleccionado.

Para saber cómo controlar las excepciones devueltas por favor consulte el apartado 1.2.3, Excepciones.

1.2.2.7 Método InfEnvios			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
dtFecha	Entrada	DateTime	Fecha de salida de los envíos a consultar.
strInfEnvios	Salida	String	XML con los datos de los envíos.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio “[LoginWSService](#)”.

Uso:

El uso es similar a la anterior.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

string xmlResult = wsWebService.InfEnvios("fecha-salida");
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>
<INF_ENVIOS
  V_COD_AGE_CARGO="--AGENCIA DE CARGO --"
  V_COD_AGE_ORI="--AGENCIA DE ORIGEN --"
  V_ALBARAN="--ALBARAN --"
  D_FECHA="--FECHA SALIDA en formato mm/dd/yyyy hh:mm:ss--"
  V_COD_AGE_DES="--AGENCIA DE DESTINO --"
  V_NOM_ORI="--NOMBRE REMITENTE--"
  V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
  V_POB_ORI="--POBLACION REMITENTE--"
  V_CP_ORI="--COD. POSTAL REMITENTE--"
  V_NOM_DES="--NOMBRE DESTINATARIO--"
  V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
  V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
  V_NUM_DES="--NÚMERO DESTINATARIO--"
  V_PISO_DES="--PISO DESTINATARIO--"
  V_POB_DES="--POBLACIÓN DESTINATARIO--"
  V_CP_DES="--COD. POSTAL DESTINATARIO--"
  V_TLF_DES="--TLFNO DESTINATARIO--"
  D_FEC_ENTR="--FECHA ENTREGA en formato mm/dd/yyyy hh:mm:ss--"
  I_BUL="--BULTOS--"
  V_REF="--REFERENCIA--"
  F_REEMBOLSO="--REEMBOLSO--"
  F_PORTE_DEB="--PORTE DEBIDO--"
  F_VALOR="--VALOR--"
  V_COD_CLI="--CODIGO CLIENTE--"
  V_ULT_TIPO_VIA="--ULTIMO TIPO VIA--"
  V_ULT_DIR="--ULTIMO NOMBRE VIA--"
  V_ULT_NUM="--ULTIMO NUMERO--"
  V_ULT_PISO="--ULTIMO PISO--"
  F_PESO_ORI="--PESO ORI--"
  V_NOM_CLI="--NOMBRE CLIENTE--"
  V_COD_CLI_DEP="--CODIGO DEPARTAMENTO--"
  V_NOM_DEP="--NOMBRE DEPARTAMENTO--"
  V_COD_TIPO_EST="--CÓDIGO TIPO DE ESTADO--"
  D_FEC_HORA_ENTR_POD="--FECHA Y HORA DE ENTREGA DEL POD--"
  V_COD_PAIS="--CÓDIGO DEL PAÍS--"
  V_NOM_PAIS="--NOMBRE DEL PAÍS--"
/>

//DEVOLVERÁ UN BLOQUE POR CADA ENVÍO

<INF_ENVIOS
  V_COD_AGE_CARGO="--AGENCIA DE CARGO --"
  V_COD_AGE_ORI="--AGENCIA DE ORIGEN --"
```

```

V_ALBARAN="--ALBARAN N--"
D_FECHA="--ETC en formato mm/dd/yyyy hh:mm:ss--"
V_COD_AGE_DES="--AGENCIA DE DESTINO --"
V_NOM_ORI="--NOMBRE REMITENTE--"
V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
V_POB_ORI="--POBLACION REMITENTE--"
V_CP_ORI="--COD. POSTAL REMITENTE--"
V_NOM_DES="--NOMBRE DESTINATARIO--"
V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
V_NUM_DES="--NÚMERO DESTINATARIO--"
V_PISO_DES="--PISO DESTINATARIO--"
V_POB_DES="--POBLACIÓN DESTINATARIO--"
V_CP_DES="--COD. POSTAL DESTINATARIO--"
V_TLF_DES="--TLFNO DESTINATARIO--"
D_FEC_ENTR="--FECHA ENTREGA en formato mm/dd/yyyy hh:mm:ss--"
I_BUL="--BULTOS--"
V_REF="--REFERENCIA--"
F_REEMBOLSO="--REEMBOLSO--"
F_PORTE_DEB="--PORTE DEBIDO--"
F_VALOR="--VALOR--"
V_COD_CLI="--CODIGO CLIENTE--"
V_ULT_TIPO_VIA="--ULTIMO TIPO VIA--"
V_ULT_DIR="--ULTIMO NOMBRE VIA--"
V_ULT_NUM="--ULTIMO NUMERO--"
V_ULT_PISO="--ULTIMO PISO--"
F_PESO_ORI="--PESO ORI--"
V_NOM_CLI="--NOMBRE CLIENTE--"
V_COD_CLI_DEP="--CODIGO DEPARTAMENTO--"
V_NOM_DEP="--NOMBRE DEPARTAMENTO--"
V_COD_TIPO_EST="--CÓDIGO TIPO DE ESTADO--"
D_FEC_HORA_ENTR_POD="--FECHA Y HORA DE ENTREGA DEL POD--"
V_COD_PAIS="--CÓDIGO DEL PAÍS--"
V_NOM_PAIS="--NOMBRE DEL PAÍS--"
/>

</CONSULTA>

```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```

PHP

<?php
    $strRespuestaServer = http_post_data('http://--ip servidor--/soap',
        '<?xml version="1.0" encoding="utf-8"?>
        <soap:Envelope
            xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema">
            <soap:Header>
                <ROClientIDHeader xmlns="http://tempuri.org/">
                    <ID>--ID de la sesión establecida--</ID>
                </ROClientIDHeader>
            </soap:Header>

```

```

        <soap:Body>
            <WebServService___InfEnvios>
                <dtFecha>-fecha-salida en formato yyyy/mm/dd-</dtFecha>
            </WebServService___InfEnvios>
        </soap:Body>
    </soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>

```

Una vez ejecutado el http_post_data la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

```

XML
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/"
>
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService___InfEnviosResponse>
      <v1:strInfEnvios>
        <CONSULTA>
          <INF_ENVIOS
            V_COD_AGE_CARGO="--AGENCIA DE CARGO --"
            V_COD_AGE_ORI="--AGENCIA DE ORIGEN --"
            V_ALBARAN="--ALBARAN --"
            D_FECHA="--FECHA SALIDA en formato mm/dd/yyyy hh:mm:ss--"
            V_COD_AGE_DES="--AGENCIA DE DESTINO --"
            V_NOM_ORI="--NOMBRE REMITENTE--"
            V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
            V_POB_ORI="--POBLACION REMITENTE--"
            V_CP_ORI="--COD. POSTAL REMITENTE--"
            V_NOM_DES="--NOMBRE DESTINATARIO--"
            V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
            V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
            V_NUM_DES="--NÚMERO DESTINATARIO--"
            V_PISO_DES="--PISO DESTINATARIO--"
            V_POB_DES="--POBLACIÓN DESTINATARIO--"
            V_CP_DES="--COD. POSTAL DESTINATARIO--"
            V_TLF_DES="--TLFNO DESTINATARIO--"
            D_FEC_ENTR="--FECHA ENTREGA en formato mm/dd/yyyy hh:mm:ss--"
            I_BUL="--BULTOS--"
            V_REF="--REFERENCIA--"
            F_REEMBOLSO="--REEMBOLSO--"
            F_PORTE_DEB="--PORTE DEBIDO--"
            F_VALOR="--VALOR--"
            V_COD_CLI="--CODIGO CLIENTE--"
            V_ULT_TIPO_VIA="--ULTIMO TIPO VIA--"

```



```

V_ULT_DIR="--ULTIMO NOMBRE VIA--"
V_ULT_NUM="--ULTIMO NUMERO--"
V_ULT_PISO="--ULTIMO PISO--"
F_PESO_ORI="--PESO ORI--"
V_NOM_CLI="--NOMBRE CLIENTE--"
V_COD_CLI_DEP="--CODIGO DEPARTAMENTO--"
V_NOM_DEP="--NOMBRE DEPARTAMENTO--"
V_COD_TIPO_EST="--CÓDIGO TIPO ESTADO--"
D_FEC_HORA_ENTR_POD="--FECHA Y HORA DE ENTREGA DEL POD--"
V_COD_PAIS="--CÓDIGO DEL PAÍS--"
V_NOM_PAIS="--NOMBRE DEL PAÍS--"
/>

```

//DEVOLVERÁ UN BLOQUE POR CADA ENVÍO

```

<INF_ENVIOS
  V_COD_AGE_CARGO="--AGENCIA DE CARGO N--"
  V_COD_AGE_ORI="--AGENCIA DE ORIGEN N--"
  V_ALBARAN="--ALBARAN N--"
  D_FECHA="--ETC en formato mm/dd/yyyy hh:mm:ss--"
  V_COD_AGE_DES="--AGENCIA DE DESTINO --"
  V_NOM_ORI="--NOMBRE REMITENTE--"
  V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
  V_POB_ORI="--POBLACION REMITENTE--"
  V_CP_ORI="--COD. POSTAL REMITENTE--"
  V_NOM_DES="--NOMBRE DESTINATARIO--"
  V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
  V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
  V_NUM_DES="--NÚMERO DESTINATARIO--"
  V_PISO_DES="--PISO DESTINATARIO--"
  V_POB_DES="--POBLACIÓN DESTINATARIO--"
  V_CP_DES="--COD. POSTAL DESTINATARIO--"
  V_TLF_DES="--TLFNO DESTINATARIO--"
  D_FEC_ENTR="--FECHA ENTREGA en formato mm/dd/yyyy hh:mm:ss--"
  I_BUL="--BULTOS--"
  V_REF="--REFERENCIA--"
  F_REEMBOLSO="--REEMBOLSO--"
  F_PORTE_DEB="--PORTE DEBIDO--"
  F_VALOR="--VALOR--"
  V_COD_CLI="--CODIGO CLIENTE--"
  V_ULT_TIPO_VIA="--ULTIMO TIPO VIA--"
  V_ULT_DIR="--ULTIMO NOMBRE VIA--"
  V_ULT_NUM="--ULTIMO NUMERO--"
  V_ULT_PISO="--ULTIMO PISO--"
  F_PESO_ORI="--PESO ORI--"
  V_NOM_CLI="--NOMBRE CLIENTE--"
  V_COD_CLI_DEP="--CODIGO DEPARTAMENTO--"
  V_NOM_DEP="--NOMBRE DEPARTAMENTO--"
  V_COD_TIPO_EST="--CÓDIGO TIPO ESTADO--"
  D_FEC_HORA_ENTR_POD="--FECHA Y HORA DE ENTREGA DEL POD--"
  V_COD_PAIS="--CÓDIGO DEL PAÍS--"
  V_NOM_PAIS="--NOMBRE DEL PAÍS--"
/>
</CONSULTA>
</v1:strInfEnvios>
</v1:WebServService___InfEnviosResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

1.2.2.8 Método ConsEtiquetaEnvio4

PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodAgeOri	Entrada	String	Agencia de origen, si es vacío se coge la agencia del usuario.
strAlbaran	Entrada	String	Albarán del envío.
intIdRepDet	Entrada	Integer	Id. del report a elegir, 0 para coger el de por defecto.
strFormato	Entrada	String	Formato de salida (PDF o TXT)
strEtiqueta	Salida	String	Etiqueta en pdf codificada en base64.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

El uso es similar a la anterior.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();  
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();  
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;  
  
string strResult = wsWebService.ConsEtiquetaEnvio4("agencia-origen", "albaran", id. report, "PDF");
```

Una vez ejecutada la función la variable `strResult` contendrá la etiqueta en pdf o txt codificada en base64.

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

PHP

```
<?php  
$strRespuestaServer = http_post_data('http://--ip servidor--/soap',  
    '<?xml version="1.0" encoding="utf-8"?>  
    <soap:Envelope  
        xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"  
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```

xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<soap:Header>
  <ROClientIDHeader xmlns="http://tempuri.org/">
    <ID>--ID de la sesión establecida--</ID>
  </ROClientIDHeader>
</soap:Header>
<soap:Body>
  <WebServService___ConsEtiquetaEnvio3>
    <strCodAgeOri>-agencia-origen-</strCodAgeOri>
    <strAlbaran>-albaran-</strAlbaran>
    <intIdRepDet>-id. report-</intIdRepDet>
    <strFormato>-formato-</strFormato>
  </WebServService___ConsEtiquetaEnvio3>
</soap:Body>
</soap:Envelope>');
$strRespuestaServer = http_parse_message ($strRespuestaServer) ->body;
?>

```

Una vez ejecutado el http_post_data la variable `$strRespuestaServer` contendrá la etiqueta en pdf codificada en base64 o una cadena de texto con los códigos de escape de la etiqueta.

1.2.2.9 Método GrabaPedido			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strXmlArticulos	Entrada	String	Xml con los artículos a grabar.
boGrabarPedido	Entrada	Boolean	Indica si se va a grabar el pedido o sólo se van a comprobar los errores.
strCodOut	Salida	String	Código del pedido generado.
dtFecHoraAltaOut	Salida	Datetime	Fecha/Hora de alta.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio “[LoginWSService](#)”.

Uso:

El uso es similar a la anterior.

Ejemplo:

```
C#

/*Código pedido*/
string strCodOut;

/*fecha-alta*/
DateTime dtFecHoraAltaOut;

WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

try
{
    strCodOut = wsWebService.GrabarPedido("datos-artículos", grabar_pedido,
        out strCodOut, out dtFecHoraAltaOut)
}
catch (Exception eExcepcion)
{
    MessageBox.Show(eExcepcion.Message);
}
```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```
PHP

<?php
$xml = '--datos-artículos--';
```

```
// Sustituye caracteres especiales para el xml
$xml = str_replace('<', '&lt;', $xml);
$xml = str_replace('>', '&gt;', $xml);
$xml = str_replace("\'", '&apos;', $xml);
$xml = str_replace('"', '&quot;', $xml);

$strRespuestaServer = http_post_data('http://--ip servidor--/soap',
'<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Header>
    <ROClientIDHeader xmlns="http://tempuri.org/">
      <ID>--id de la sesión establecida--</ID>
    </ROClientIDHeader>
  </soap:Header>
  <soap:Body>
    <WebServService___GrabaPedido xmlns="http://tempuri.org/">
      <strXmlArticulos>'. $xml. '</strXmlArticulos>
      <boGrabarPedido>--grabar-pedido--</boGrabarPedido>
    </WebServService___GrabaPedido>
  </soap:Body>
</soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>
```

El parámetro `strXmlArticulos` debe contener los datos de los artículos dentro de un XML, parseado previamente sustituyendo (<) por (<), (>) por (>), (\) por (') y (") por ("), con la siguiente forma:

```
XML
<strXmlArticulos>
  <Root>
    <Item Arti = 'código-artículo 1' UdSol = "unidades" DesArti = 'descripción-artículo 1'/>
    <Item Arti = 'código-artículo 2' UdSol = "unidades" DesArti = 'descripción-artículo 2'/>
    <Item Arti = 'código-artículo 3' UdSol = "unidades" DesArti = 'descripción-artículo 3'/>
  </Root>
</strXmlArticulos>
```

Una vez ejecutado el `http_post_data` la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/">
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService__GrabaPedidoResponse>
      <v1:strCodOut>--OUT CÓDIGO--</v1:strCodOut>
      <v1:dtFecHoraAltaOut>--OUT FECHA HORA ALTA en formato yyyy/mm/dd hh:mm:ss--
    </v1:dtFecHoraAltaOut>
    </v1:WebServService__GrabaPedidoResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Excepciones:

Esta función devuelve las siguientes excepciones:

CÓDIGO	DESCRIPCIÓN
1	El usuario no existe o está inactivo.
2	La agencia que solicita el pedido no existe o está inactiva.
3	Las unidades mínimas de venta para el artículo xxx son xxx unidades.
4	El código de artículo xxx no existe o está inactivo.
5	El pedido no contiene ningún artículo.
6	El artículo xxx es exclusivo para agencias.

Para saber cómo controlar las excepciones devueltas por favor consulte el apartado 1.2.3, Excepciones.

1.2.2.10 Método ConsEnvEstadosUltimo			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strEnvios	Entrada	String	Envíos a consultar, se manda en formato de xml.

intUltVersion	Entrada	Integer	Rowversion de la última actualización de estados, 0 para traérselos todos.
strEnvEstados	Salida	String	XML con los datos de los últimos estados de los envíos.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

Si disponemos de un lenguaje que interprete los archivos .wsdl (como por ejemplo C#), simplemente tendremos que crear una referencia Web importando el fichero .wsdl y a continuación instanciar esa referencia para poder llamar a las funciones.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

string xmlResult = wsWebService.ConsEnvEstadosUltimo("envíos", rowversion);
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>
  <ENV_ESTADOS_ULTIMO
    V_ALBARAN="--ALBARÁN 1--"
    V_COD_TIPO_EST="--TIPO ESTADO--"
    RV_VERSION="--ROWVERSION--"
  />
  .
  //DEVOLVERÁ UN BLOQUE POR CADA ENVÍO
  .
  <ENV_ESTADOS_ULTIMO
    V_ALBARAN="--ALBARÁN N--"
    V_COD_TIPO_EST="--ETC --"
    RV_VERSION="--ETC--"
  />
</CONSULTA>
```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

PHP

```
<?php
$xml = '-envíos-';

// Sustituye caracteres especiales para el xml
$xml = str_replace('<', '&lt;', $xml);
$xml = str_replace('>', '&gt;', $xml);
$xml = str_replace("'", '&apos;', $xml);
$xml = str_replace('"', '&quot;', $xml);

$strRespuestaServer = http_post_data('http://--ip servidor--/soap',
    '<?xml version="1.0" encoding="utf-8"?>
    <soap:Envelope
        xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        <soap:Header>
            <ROClientIDHeader xmlns="http://tempuri.org/">
                <ID>--ID de la sesión establecida--</ID>
            </ROClientIDHeader>
        </soap:Header>
        <soap:Body>
            <WebServService__ConsEnvEstadosUltimo>
                <strEnvios>'. $xml. '</strEnvios>
                <intUltVersion>--rowversion--</intUltVersion>
            </WebServService__ConsEnvEstadosUltimo>
        </soap:Body>
    </soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>
```

El parámetro **strEnvios** debe contener los albaranes de los envíos a consultar dentro de un XML, parseado previamente sustituyendo (<) por (<), (>) por (>), (\) por (') y (") por ("), con la siguiente forma:

XML

```
<strEnvios>
    <Envios>
        <Envio strAlbaran = albarán 1/>
        <Envio strAlbaran = albarán 2/>
        <Envio strAlbaran = albarán 3/>
    </Envios>
</strEnvios>
```

Una vez ejecutado el http_post_data la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/"
>
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService___ConsEnvEstadosUltimoResponse>
      <v1:strEnvEstados>
        <CONSULTA>
          <ENV_ESTADOS_ULTIMO
            V_ALBARAN="--ALBARÁN 1--"
            V_COD_TIPO_EST="--TIPO ESTADO--"
            RV_VERSION="--ROWVERSION--"
          />
          .
          . //DEVOLVERÁ UN BLOQUE POR CADA ENVÍO
          .
          <ENV_ESTADOS_ULTIMO
            V_ALBARAN="--ALBARÁN N--"
            V_COD_TIPO_EST="--ETC --"
            RV_VERSION="--ETC--"
          />
        </CONSULTA>
      </v1:strEnvEstados>
    </v1:WebServService___ConsEnvEstadosUltimoResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

1.2.2.11 Método ConsEnvIncidencias			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodAgeCargo	Entrada	String	Agencia de cargo.
strCodAgeOri	Entrada	String	Agencia de origen.
strAlbaran	Entrada	String	Albarán.
strEnvIncidencias	Salida	String	XML con los datos de las incidencias del envío.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

Si disponemos de un lenguaje que interprete los archivos .wsdl (como por ejemplo C#), simplemente tendremos que crear una referencia Web importando el fichero .wsdl y a continuación instanciar esa referencia para poder llamar a las funciones.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

string xmlResult = wsWebService.ConsEnvIncidencias("agencia-cargo", "agencia-origen", "albaran");
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>
<ENV_INCIDENCIAS>
<INC
  I_ID="--ID DE LA INCIDENCIA 1--"
  V_COD_TIPO_INC="--TIPO INCIDENCIA--"
  T_OBS="--OBSERVACIONES--"
  D_FEC_HORA_CIERRE="--FECHA/HORA CIERRE en formato mm/dd/yyyy hh:mm:ss--"
  B_RESUELTA="--RESUELTA--"
  D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
  V_COD_USU_ALTA="--USUARIO ALTA--"
  V_COD_USU_RES="--USUARIO RESUELVE--"
  V_COD_AGE_ALTA="--AGENCIA ALTA--"
  V_COD_REP_ALTA="--REPARTIDOR ALTA--"
  V_COD_CLI_ALTA="--CLIENTE ALTA--"
  V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
  SD_HORA_INC="--HORA INCIDENCIA en formato hh:mm--"
/>

<ACT
  I_ID="--ID DE LA ACTUACIÓN 1--"
  V_COD_TIPO_ACTU="--TIPO ACTUACIÓN--"
  V_DES_TIPO_ACTU="--DESCRIPCIÓN ACTUACIÓN--"
  D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
  V_COD_USU_ALTA="--USUARIO ALTA--"
  V_COD_AGE="--AGENCIA--"
  B_LEIDA="--LEÍDA--"
  B_INCL_EMAIL_CLI="--INCLUIR EN EMAIL--"
  V_COD_USU_LEIDA="--USUARIO LEÍDA--"
  D_FEC_HORA_LEIDA="--FECHA/HORA LEIDA en formato mm/dd/yyyy hh:mm:ss--"
  V_TIPO="--TIPO--"
  V_COD_CARGO_A="--CARGO A--"
  V_COD_AGE_ALTA="--AGENCIA ALTA--"
  V_COD_CLI_ALTA="--CLIENTE ALTA--"
  V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
  V_COD_REP_ALTA="--REPARTIDOR ALTA--"
/>
. //DEVOLVERÁ UN BLOQUE POR CADA ACTUACIÓN QUE TENGA ESA INCIDENCIA
.
<ACT
.
.
.
/>
</ENV_INCIDENCIAS>
. //DEVOLVERÁ UN BLOQUE POR CADA INCIDENCIA QUE TENGA ESE ENVÍO
.
<ENV_INCIDENCIAS>
.
.
.
</ENV_INCIDENCIAS>
```

</CONSULTA>

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

PHP

```
<?php
$strRespuestaServer = http_post_data('http://--ip servidor--/soap',
'<?xml version="1.0" encoding="utf-8"?>
  <soap:Envelope
    xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Header>
      <ROClientIDHeader xmlns="http://tempuri.org/">
        <ID>--ID de la sesión establecida--</ID>
      </ROClientIDHeader>
    </soap:Header>
    <soap:Body>
      <WebServService__ConsEnvIncidencias>
        <strCodAgeCargo>-agencia-cargo-</strCodAgeCargo>
        <strCodAgeOri>--agencia-origen--</strCodAgeOri>
        <strAlbaran>--albaran--</strAlbaran>
      </WebServService__ConsEnvIncidencias>
    </soap:Body>
  </soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>
```

Una vez ejecutado el http_post_data la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/"
>
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService__ConsEnvIncidenciasResponse>
      <v1:strEnvIncidencias>
```

```

<CONSULTA>
<ENV_INCIDENCIAS>
  <INC
    I_ID="--ID DE LA INCIDENCIA 1--"
    V_COD_TIPO_INC="--TIPO INCIDENCIA--"
    T_OBS="--OBSERVACIONES--"
    D_FEC_HORA_CIERRE="--FECHA/HORA CIERRE en formato mm/dd/yyyy hh:mm:ss--"
    B_RESUELTA="--RESUELTA--"
    D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
    V_COD_USU_ALTA="--USUARIO ALTA--"
    V_COD_USU_RES="--USUARIO RESUELVE--"
    V_COD_AGE_ALTA="--AGENCIA ALTA--"
    V_COD_REP_ALTA="--REPARTIDOR ALTA--"
    V_COD_CLI_ALTA="--CLIENTE ALTA--"
    V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
    SD_HORA_INC="--HORA INCIDENCIA en formato hh:mm--"
  />

  <ACT
    I_ID="--ID DE LA ACTUACIÓN 1--"
    V_COD_TIPO_ACTU="--TIPO ACTUACIÓN--"
    V_DES_TIPO_ACTU="--DESCRIPCIÓN ACTUACIÓN--"
    D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
    V_COD_USU_ALTA="--USUARIO ALTA--"
    V_COD_AGE="--AGENCIA--"
    B_LEIDA="--LEÍDA--"
    B_INCL_EMAIL_CLI="--INCLUIR EN EMAIL--"
    V_COD_USU_LEIDA="--USUARIO LEÍDA--"
    D_FEC_HORA_LEIDA="--FECHA/HORA LEIDA en formato mm/dd/yyyy hh:mm:ss--"
    V_TIPO="--TIPO--"
    V_COD_CARGO_A="--CARGO A--"
    V_COD_AGE_ALTA="--AGENCIA ALTA--"
    V_COD_CLI_ALTA="--CLIENTE ALTA--"
    V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
    V_COD_REP_ALTA="--REPARTIDOR ALTA--"
  />

  . //DEVOLVERÁ UN BLOQUE POR CADA ACTUACIÓN QUE TENGA ESA INCIDENCIA
  .
  <ACT
    .
    .
    .
  />
</ENV_INCIDENCIAS>

. //DEVOLVERÁ UN BLOQUE POR CADA INCIDENCIA QUE TENGA ESE ENVÍO
.
<ENV_INCIDENCIAS>
.
.
.
</ENV_INCIDENCIAS>
</CONSULTA>
</v1:strEnvIncidencias>
</v1:WebServService___ConsEnvIncidenciasResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

1.2.2.12 Método ConsEnvEstadosFecha			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
dtFecha	Entrada	Datetime	Fecha del estado.
strEnvEstados	Salida	String	XML con los datos de los estados.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

Si disponemos de un lenguaje que interprete los archivos .wsdl (como por ejemplo C#), simplemente tendremos que crear una referencia Web importando el fichero .wsdl y a continuación instanciar esa referencia para poder llamar a las funciones.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();  
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();  
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;  
  
string xmlResult = wsWebService.ConsEnvEstadosFecha("fecha");
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

```
XML
<CONSULTA>
<ENV_ESTADOS
  V_COD_AGE_CARGO="--AGENCIA CARGO ESTADO 1--"
  V_COD_AGE_ORI="--AGENCIA ORIGEN ESTADO 1--"
  V_ALBARAN="--ALBARÁN ESTADO 1--"
  I_ID="--ID DEL ESTADO 1--"
  V_COD_TIPO_EST="--TIPO ESTADO--"
  D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
  V_COD_USU_ALTA="--USUARIO ALTA--"
  V_COD_AGE_ALTA="--AGENCIA ALTA--"
  V_COD_REP_ALTA="--REPARTIDOR ALTA--"
  V_COD_CLI_ALTA="--CLIENTE ALTA--"
  V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
/>
.
. //DEVOLVERÁ UN BLOQUE POR CADA ESTADO QUE TENGA ESA FECHA
.
<ENV_ESTADOS
  V_COD_AGE_CARGO="--AGENCIA CARGO ESTADO N--"
  V_COD_AGE_ORI="--AGENCIA ORIGEN ESTADO N--"
  V_ALBARAN="--ALBARÁN ESTADO N--"
  I_ID="--ID DEL ESTADO N--"
  V_COD_TIPO_EST="--ETC --"
  D_FEC_HORA_ALTA="--ETC en formato mm/dd/yyyy hh:mm:ss--"
  V_COD_USU_ALTA="--ETC --"
  V_COD_AGE_ALTA="--ETC --"
  V_COD_REP_ALTA="--ETC --"
  V_COD_CLI_ALTA="--ETC --"
  V_COD_CLI_DEP_ALTA="--ETC --"
/>
</CONSULTA>
```


Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

PHP

```
<?php
$strRespuestaServer = http_post_data('http://--ip servidor--/soap',
'<?xml version="1.0" encoding="utf-8"?>
    <soap:Envelope
        xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        <soap:Header>
            <ROClientIDHeader xmlns="http://tempuri.org/">
                <ID>--ID de la sesión establecida--</ID>
            </ROClientIDHeader>
        </soap:Header>
        <soap:Body>
            <WebServService___ConsEnvEstadosFecha>
                <dtFecha>--fecha en formato yyyy/mm/dd--</dtFecha>
            </WebServService___ConsEnvEstadosFecha>
        </soap:Body>
    </soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>
```

Una vez ejecutado el http_post_data la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/"
>
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService___ConsEnvEstadosFechaResponse>
      <v1:strEnvEstados>
        <CONSULTA>
          <ENV_ESTADOS
            V_COD_AGE_CARGO="--AGENCIA CARGO DEL ESTADO 1--"
            V_COD_AGE_ORI="--AGENCIA ORIGEN DEL ESTADO 1--"
            V_ALBARAN="--ALBARÁN DEL ESTADO 1--"
            I_ID="--ID DEL ESTADO 1--"
            V_COD_TIPO_EST="--TIPO ESTADO--"
            D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
            V_COD_USU_ALTA="--USUARIO ALTA--"
            V_COD_AGE_ALTA="--AGENCIA ALTA--"
            V_COD_REP_ALTA="--REPARTIDOR ALTA--"
            V_COD_CLI_ALTA="--CLIENTE ALTA--"
            V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
          />
          .
          . //DEVOLVERÁ UN BLOQUE POR CADA ESTADO QUE TENGA ESA FECHA
          .
          <ENV_ESTADOS
            V_COD_AGE_CARGO="--AGENCIA CARGO DEL ESTADO N--"
            V_COD_AGE_ORI="--AGENCIA ORIGEN DEL ESTADO N--"
            V_ALBARAN="--ALBARÁN DEL ESTADO N--"
            I_ID="--ID DEL ESTADO N--"
            V_COD_TIPO_EST="--ETC --"
            D_FEC_HORA_ALTA="--ETC en formato mm/dd/yyyy hh:mm:ss--"
            V_COD_USU_ALTA="--ETC --"
            V_COD_AGE_ALTA="--ETC --"
            V_COD_REP_ALTA="--ETC --"
            V_COD_CLI_ALTA="--ETC --"
            V_COD_CLI_DEP_ALTA="--ETC --"
          />
        </CONSULTA>
      </v1:strEnvEstados>
    </v1:WebServService___ConsEnvEstadosFechaResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

1.2.2.13 Método ConsEnvIncidenciasFecha			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
dtFecha	Entrada	Datetime	Fecha de la incidencia.
strEnvIncidencias	Salida	String	XML con los datos de las incidencias.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

Si disponemos de un lenguaje que interprete los archivos .wsdl (como por ejemplo C#), simplemente tendremos que crear una referencia Web importando el fichero .wsdl y a continuación instanciar esa referencia para poder llamar a las funciones.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();  
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();  
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;  
  
string xmlResult = wsWebService.ConsEnvIncidenciasFecha("fecha");
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>
<ENV_INCIDENCIAS>
  <INC
    V_COD_AGE_CARGO="--AGENCIA CARGO DE LA INCIDENCIA 1--"
    V_COD_AGE_ORI="--AGENCIA ORIGEN DE LA INCIDENCIA 1--"
    V_ALBARAN="--ALBARÁN DE LA INCIDENCIA 1--"
    I_ID="--ID DE LA INCIDENCIA 1--"
    V_COD_TIPO_INC="--TIPO INCIDENCIA--"
    T_OBS="--OBSERVACIONES--"
    D_FEC_HORA_CIERRE="--FECHA/HORA CIERRE en formato mm/dd/yyyy hh:mm:ss--"
    B_RESUELTA="--RESUELTA--"
    D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
    V_COD_USU_ALTA="--USUARIO ALTA--"
    V_COD_USU_RES="--USUARIO RESUELVE--"
    V_COD_AGE_ALTA="--AGENCIA ALTA--"
    V_COD_REP_ALTA="--REPARTIDOR ALTA--"
    V_COD_CLI_ALTA="--CLIENTE ALTA--"
    V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
    SD_HORA_INC="--HORA INCIDENCIA en formato hh:mm--"
  />

  <ACT
    I_ID="--ID DE LA ACTUACIÓN 1--"
    V_COD_TIPO_ACTU="--TIPO ACTUACIÓN--"
    V_DES_TIPO_ACTU="--DESCRIPCIÓN ACTUACIÓN--"
    D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
    V_COD_USU_ALTA="--USUARIO ALTA--"
    V_COD_AGE="--AGENCIA--"
    B_LEIDA="--LEÍDA--"
    B_INCL_EMAIL_CLI="--INCLUIR EN EMAIL--"
    V_COD_USU_LEIDA="--USUARIO LEÍDA--"
    D_FEC_HORA_LEIDA="--FECHA/HORA LEIDA en formato mm/dd/yyyy hh:mm:ss--"
    V_TIPO="--TIPO--"
    V_COD_CARGO_A="--CARGO A--"
    V_COD_AGE_ALTA="--AGENCIA ALTA--"
    V_COD_CLI_ALTA="--CLIENTE ALTA--"
    V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
    V_COD_REP_ALTA="--REPARTIDOR ALTA--"
  />
  .
  . //DEVOLVERÁ UN BLOQUE POR CADA ACTUACIÓN QUE TENGA ESA INCIDENCIA
  .
  <ACT
    .
    .
  />
</ENV_INCIDENCIAS>
.
. //DEVOLVERÁ UN BLOQUE POR CADA INCIDENCIA QUE TENGA ESA FECHA
.
<ENV_INCIDENCIAS>
.
.
```

```
</ENV_INCIDENCIAS>
</CONSULTA>
```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

PHP

```
<?php
$strRespuestaServer = http_post_data('http://--ip servidor--/soap',
'<?xml version="1.0" encoding="utf-8"?>
  <soap:Envelope
    xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Header>
      <ROClientIDHeader xmlns="http://tempuri.org/">
        <ID>--ID de la sesión establecida--</ID>
      </ROClientIDHeader>
    </soap:Header>
    <soap:Body>
      <WebServService___ConsEnvIncidenciasFecha>
        <dtFecha>-fecha en formato yyyy/mm/dd-</dtFecha>
      </WebServService___ConsEnvIncidenciasFecha>
    </soap:Body>
  </soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>
```

Una vez ejecutado el http_post_data la variable \$strRespuestaServer contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/"
>
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService___ConsEnvIncidenciasFechaResponse>
      <v1:strEnvIncidencias>
        <CONSULTA>
          <ENV_INCIDENCIAS>
```

```

<INC
  V_COD_AGE_CARGO="--AGENCIA CARGO DE LA INCIDENCIA 1--"
  V_COD_AGE_ORI="--AGENCIA ORIGEN DE LA INCIDENCIA 1--"
  V_ALBARAN="--ALBARÁN DE LA INCIDENCIA 1--"
  I_ID="--ID DE LA INCIDENCIA 1--"
  V_COD_TIPO_INC="--TIPO INCIDENCIA--"
  T_OBS="--OBSERVACIONES--"
  D_FEC_HORA_CIERRE="--FECHA/HORA CIERRE en formato mm/dd/yyyy hh:mm:ss--"
  B_RESUELTA="--RESUELTA--"
  D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
  V_COD_USU_ALTA="--USUARIO ALTA--"
  V_COD_USU_RES="--USUARIO RESUELVE--"
  V_COD_AGE_ALTA="--AGENCIA ALTA--"
  V_COD_REP_ALTA="--REPARTIDOR ALTA--"
  V_COD_CLI_ALTA="--CLIENTE ALTA--"
  V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
  SD_HORA_INC="--HORA INCIDENCIA en formato hh:mm--"
/>

<ACT
  I_ID="--ID DE LA ACTUACIÓN 1--"
  V_COD_TIPO_ACTU="--TIPO ACTUACIÓN--"
  V_DES_TIPO_ACTU="--DESCRIPCIÓN ACTUACIÓN--"
  D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
  V_COD_USU_ALTA="--USUARIO ALTA--"
  V_COD_AGE="--AGENCIA--"
  B_LEIDA="--LEÍDA--"
  B_INCL_EMAIL_CLI="--INCLUIR EN EMAIL--"
  V_COD_USU_LEIDA="--USUARIO LEÍDA--"
  D_FEC_HORA_LEIDA="--FECHA/HORA LEIDA en formato mm/dd/yyyy hh:mm:ss--"
  V_TIPO="--TIPO--"
  V_COD_CARGO_A="--CARGO A--"
  V_COD_AGE_ALTA="--AGENCIA ALTA--"
  V_COD_CLI_ALTA="--CLIENTE ALTA--"
  V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
  V_COD_REP_ALTA="--REPARTIDOR ALTA--"
/>

. //DEVOLVERÁ UN BLOQUE POR CADA ACTUACIÓN QUE TENGA ESA INCIDENCIA
.
<ACT
.
.
.
/>
</ENV_INCIDENCIAS>

. //DEVOLVERÁ UN BLOQUE POR CADA INCIDENCIA QUE TENGA ESE ENVÍO
.
<ENV_INCIDENCIAS>
.
.
.
</ENV_INCIDENCIAS>
</CONSULTA>
</v1:strEnvIncidencias>
</v1:WebServService___ConsEnvIncidenciasFechaResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

1.2.2.14 Método GrabaEnvIncActuacion			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodAgeCargo	Entrada	String	Agencia de cargo.
strCodAgeOri	Entrada	String	Agencia de origen.
strAlbaran	Entrada	String	Albarán.
intIdEnvInc	Entrada	Integer	Id. de la incidencia, -1 para generar una actuación sin incidencia.
strCodTipoActu	Entrada	String	Código de la actuación.
bolInclEmailCli	Entrada	Boolean	Incluir en email.
strTipo	Entrada	String	Tipo (INT: Interno, AGE: Agencia, CLI: Cliente, REP: Repartidor, RUT: Ruta).

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio ["LoginWSService"](#).

Uso:

El uso es similar a la anterior.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

try
{
    wsWebService.GrabaEnvIncActuacion("agencia-cargo", "agencia-ori", "albarán", id,
        "código-actuación", incluir-en-email, "tipo")
}
catch (Exception eExcepcion)
{
    MessageBox.Show(eExcepcion.Message);
}
```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

PHP

```
<?php
    http_post_data('http://--ip servidor--/soap',
    '<?xml version="1.0" encoding="utf-8"?>
    <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        <soap:Header>
            <ROClientIDHeader xmlns="http://tempuri.org/">
                <ID>--id de la sesión establecida--</ID>
            </ROClientIDHeader>
        </soap:Header>
        <soap:Body>
            <WebServService___GrabaEnvIncActuacion xmlns="http://tempuri.org/">
                <strCodAgeCargo>--agencia-cargo--</strCodAgeCargo>
                <strCodAgeOri>--agencia-origen--</strCodAgeOri>
                <strAlbaran>--albarán--</strAlbaran>
                <intIdEnvInc>--id--</intIdEnvInc>
                <strCodTipoActu>--código-actuación--</strCodTipoActu>
                <bolCnlEmailCli>--incluir-en-email--</bolCnlEmailCli>
                <strTipo>--tipo--</strTipo>
            </WebServService___GrabaEnvIncActuacion>
        </soap:Body>
    </soap:Envelope>');
?>
```

Excepciones:

Esta función devuelve las siguientes excepciones:

CÓDIGO	DESCRIPCIÓN
1	El código de actuación no existe o está inactivo.
2	El tipo de la actuación no es válido.
3	El usuario no tiene permiso para meter actuaciones.
4	El envío no existe o el usuario no tiene permiso para modificarlo.
5	El usuario debe pertenecer a la agencia de cargo.
6	No se puede introducir una actuación sin incidencia si el envío ya tiene incidencias.
7	La incidencia que se genera con la actuación no está configurada.
8	El usuario debe pertenecer a la agencia de cargo o a la agencia de destino.
9	La incidencia indicada no existe.
10	El tipo de servicio del envío no permite grabar estados, incidencias o actuaciones sin incidencias

Para saber cómo controlar las excepciones devueltas por favor consulte el apartado 1.2.3, Excepciones.

1.2.2.15 Método ConsEnvTraz			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodAgeCargo	Entrada	String	Agencia de cargo.
strCodAgeOri	Entrada	String	Agencia de origen.
strAlbaran	Entrada	String	Albarán.
strEnvTraz	Salida	String	XML con los datos de trazabilidad del envío.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio “[LoginWSService](#)”.

Uso:

Si disponemos de un lenguaje que interprete los archivos .wsdl (como por ejemplo C#), simplemente tendremos que crear una referencia Web importando el fichero .wsdl y a continuación instanciar esa referencia para poder llamar a las funciones.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();  
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();  
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;  
  
string xmlResult = wsWebService.ConsEnvTraz("agencia-cargo", "agencia-origen", "albaran");
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>
  <ENV_TRAZ
    V_COD_AGE_CARGO="--AGENCIA CARGO 1--"
    V_COD_AGE_ORI="--AGENCIA ORIGEN--"
    V_ALBARAN="--ALBARÁN--"
    V_BUL="--BULTO--"
    V_COD_PUN_LEC="--PUNTO DE LECTURA--"
    D_FEC_HORA_LEC="--FECHA/HORA LECTURA en formato mm/dd/yyyy hh:mm:ss--"
    V_COD_USU="--USUARIO ALTA--"
    D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
    V_COD_REP_ALTA="--REPARTIDOR ALTA--"
    V_COD_AGE_ALTA="--AGENCIA ALTA--"
  />
  .
  . //DEVOLVERÁ UN BLOQUE POR CADA TRAZA QUE TENGA ESE ENVÍO
  .
  <ENV_TRAZ
    .
    .
    .
  />
</CONSULTA>
```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```
PHP

<?php
    $strRespuestaServer = http_post_data('http://--ip servidor--/soap',
        '<?xml version="1.0" encoding="utf-8"?>
        <soap:Envelope
            xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema">
            <soap:Header>
                <ROClientIDHeader xmlns="http://tempuri.org/">
                    <ID>--ID de la sesión establecida--</ID>
                </ROClientIDHeader>
            </soap:Header>
            <soap:Body>
                <WebServService___ConsEnvTraz>
                    <strCodAgeCargo>--agencia-cargo-</strCodAgeCargo>
                    <strCodAgeOri>--agencia-origen--</strCodAgeOri>
                    <strAlbaran>--albarán--</strAlbaran>
                </WebServService___ConsEnvTraz>
            </soap:Body>
        </soap:Envelope>');
    $strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>
```

Una vez ejecutado el http_post_data la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/"
>
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService___ConsEnvTrazResponse>
      <v1:strEnvTraz>
        <CONSULTA>
          <ENV_TRAZ
            V_COD_AGE_CARGO="--AGENCIA CARGO 1--"
            V_COD_AGE_ORI="--AGENCIA ORIGEN--"
            V_ALBARAN="--ALBARÁN--"
            V_BUL="--BULTO--"
            V_COD_PUN_LEC="--PUNTO DE LECTURA--"
            D_FEC_HORA_LEC="--FECHA/HORA LECTURA en formato mm/dd/yyyy hh:mm:ss--"
            V_COD_USU="--USUARIO ALTA--"
            D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
            V_COD_REP_ALTA="--REPARTIDOR ALTA--"
            V_COD_AGE_ALTA="--AGENCIA ALTA--"
          />
          .
          . //DEVOLVERÁ UN BLOQUE POR CADA TRAZA QUE TENGA ESE ENVÍO
          .
          <ENV_TRAZ
            .
            .
            .
          />
        </CONSULTA>
      </v1:strEnvTraz>
    </v1:WebServService___ConsEnvTrazResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

1.2.2.16 Método ConsEnvTrazFecAge			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
dtFecha	Entrada	Datetime	Fecha de salida del envío.
strCodAge	Entrada	String	Agencia de lectura.
strEnvTraz	Salida	String	XML con los datos de trazabilidad de los envíos.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

Si disponemos de un lenguaje que interprete los archivos .wsdl (como por ejemplo C#), simplemente tendremos que crear una referencia Web importando el fichero .wsdl y a continuación instanciar esa referencia para poder llamar a las funciones.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();  
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();  
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;  
  
string xmlResult = wsWebService.ConsEnvTrazFecAge("fecha", "agencia-lectura");
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>
  <ENV_TRAZ
    V_COD_AGE_CARGO="--AGENCIA CARGO 1--"
    V_COD_AGE_ORI="--AGENCIA ORIGEN--"
    V_ALBARAN="--ALBARÁN--"
    V_BUL="--BULTO--"
    V_COD_PUN_LEC="--PUNTO DE LECTURA--"
    D_FEC_HORA_LEC="--FECHA/HORA LECTURA en formato mm/dd/yyyy hh:mm:ss--"
    V_COD_USU="--USUARIO ALTA--"
    D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
    V_COD_REP_ALTA="--REPARTIDOR ALTA--"
    V_COD_AGE_ALTA="--AGENCIA ALTA--"
  />
  .
  . //DEVOLVERÁ UN BLOQUE POR CADA TRAZA QUE HAYA PARA LA FECHA Y AGENCIA
  INDICADAS
  .
  <ENV_TRAZ
    .
    .
    .
  />
</CONSULTA>
```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

PHP

```
<?php
$strRespuestaServer = http_post_data('http://--ip servidor--/soap',
'<?xml version="1.0" encoding="utf-8"?>
  <soap:Envelope
    xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Header>
      <ROClientIDHeader xmlns="http://tempuri.org/">
        <ID>--ID de la sesión establecida--</ID>
      </ROClientIDHeader>
    </soap:Header>
    <soap:Body>
      <WebServService__ConsEnvTrazFecAge>
        <dtFecha>-fecha en formato yyyy/mm/dd-</dtFecha>
        <strCodAge>--agencia-lectura--</strCodAge>
      </WebServService__ConsEnvTrazFecAge>
    </soap:Body>
  </soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>
```

Una vez ejecutado el http_post_data la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/"
>
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService___ConsEnvTrazFecAgeResponse>
      <v1:strEnvTraz>
        <CONSULTA>
          <ENV_TRAZ
            V_COD_AGE_CARGO="--AGENCIA CARGO 1--"
            V_COD_AGE_ORI="--AGENCIA ORIGEN--"
            V_ALBARAN="--ALBARÁN--"
            V_BUL="--BULTO--"
            V_COD_PUN_LEC="--PUNTO DE LECTURA--"
            D_FEC_HORA_LEC="--FECHA/HORA LECTURA en formato mm/dd/yyyy hh:mm:ss--"
            V_COD_USU="--USUARIO ALTA--"
            D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
            V_COD_REP_ALTA="--REPARTIDOR ALTA--"
            V_COD_AGE_ALTA="--AGENCIA ALTA--"
          />
          .
          . //DEVOLVERÁ UN BLOQUE POR CADA TRAZA QUE HAYA PARA LA FECHA Y AGENCIA
          INDICADAS
          .
          <ENV_TRAZ
            .
            .
          />
        </CONSULTA>
      </v1:strEnvTraz>
    </v1:WebServService___ConsEnvTrazFecAgeResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```


1.2.2.17 Método DigitalizaEnvios			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strEnviosDig	Entrada	String	XML que contiene las digitalizaciones a grabar. Ejemplo de XML que contiene dos digitalizaciones: <EnviosDig><EnvioDig strEnvio = '0080520080520123456789' dtFecHoraDig = '23/10/2013 16:15' /><EnvioDig strEnvioDig = '0020000080520123456789' dtFecHoraDig = '23/10/2013 16:18' /></EnviosDig> Este XML hay que parsearlo, sustituyendo (<) por (<), (>) por (>), (\) por (') y (") por (").
intRegImp	Salida	Integer	Registros bien importados.
intRegNoImp	Salida	Integer	Registros mal importados.
strEnviosDigError	Salida	String	XML que contiene las digitalizaciones que no se han podido grabar con su mensaje de error. Ejemplo de XML que contiene digitalizaciones mal importadas: <CONSULTA><ENV_DIG_ERROR strEnvio="0084948948948948498498" strError="El envío no existe"/></CONSULTA>

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

Si disponemos de un lenguaje que interprete los archivos .wsdl (como por ejemplo C#), simplemente tendremos que crear una referencia Web importando el fichero .wsdl y a continuación instanciar esa referencia para poder llamar a las funciones.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

string xmlResult = wsWebService.DigitalizaEnvios("xml-digitalizaciones", "registros-importados", "registros no importados");
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

```
XML
<CONSULTA>
  <ENV_DIG_ERROR
    strEnvio="--ENVÍO 1--"
    strError="--MENSAJE DE ERROR 1--"
  />
  .
  . //DEVOLVERÁ UN BLOQUE POR CADA DIGITALIZACIÓN MAL IMPORTADA
  .
  <ENV_DIG_ERROR
    .
    .
    .
  />
</CONSULTA>
```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```
PHP
<?php
    $xml = '-xml-digitalizaciones-';

    // Sustituye caracteres especiales para el xml
    $xml = str_replace('<', '&lt;', $xml);
    $xml = str_replace('>', '&gt;', $xml);
    $xml = str_replace('\'', '&apos;', $xml);
    $xml = str_replace('"', '&quot;', $xml);

    $strRespuestaServer = http_post_data('http://--ip servidor--/soap',
        '<?xml version="1.0" encoding="utf-8"?>
        <soap:Envelope
            xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema">
            <soap:Header>
                <ROClientIDHeader xmlns="http://tempuri.org/">
                    <ID>--ID de la sesión establecida--</ID>
                </ROClientIDHeader>
            </soap:Header>
            <soap:Body>
                <WebServService__DigitalizaEnvios>
                    <strEnviosDig>'. $xml .'</dtFecha>
                </WebServService__DigitalizaEnvios>
            </soap:Body>
```

```

    </soap:Envelope>');
    $strRespuestaServer = http_parse_message($strRespuestaServer)->body;
    ?>
  
```

Una vez ejecutado el `http_post_data` la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

```

XML
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/"
>
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService___DigitalizaEnviosResponse>
      <v1:strEnviosDigError>
        <CONSULTA>
          <ENV_DIG_ERROR
            strEnvio="--ENVIO 1--"
            strError="--MENSAJE DE ERROR 1--"
          />
          .
          . //DEVOLVERÁ UN BLOQUE POR CADA DIGITALIZACIÓN MAL IMPORTADA
          .
          <ENV_DIG_ERROR
            .
            .
          />
        </CONSULTA>
      </v1:strEnviosDigError>
      <v1:intRegImp>
        N° de digitalizaciones bien importadas
      </v1:intRegImp>
      <v1:intRegNoImp>
        N° de digitalizaciones mal importadas
      </v1:intRegNoImp>
    </v1:WebServService___DigitalizaEnviosResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
  
```

1.2.2.18 Método ConsEnvEstadosRef			
PARAMETRO	E/S	TIPO	DESCRIPCIÓN
strRef	Entrada	String	Referencia del envío.
strEnvEstadosRef	Salida	String	XML con los datos de los estados del envío.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio ["LoginWSService"](#).

Uso:

Si disponemos de un lenguaje que interprete los archivos .wsdl (como por ejemplo C#), simplemente tendremos que crear una referencia Web importando el fichero .wsdl y a continuación instanciar esa referencia para poder llamar a las funciones.

Ejemplo:
C#

```
WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

string xmlResult = wsWebService.ConsEnvEstadosRef("ref");
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>
<ENV_ESTADOS_REF
  V_SERVICIO="--CARGO+ORIGEN+ALBARÁN DEL ESTADO 1--"
  V_COD_TIPO_EST="--TIPO ESTADO--"
  D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
  B_ULT="--SI ES EL ÚLTIMO ESTADO AÑADIDO--"
/>
```

```

. //DEVOLVERÁ UN BLOQUE POR CADA ESTADO QUE TENGA ESE ENVÍO
.
<ENV_ESTADOS_REF
  V_SERVICIO="--CARGO+ORIGEN+ALBARÁN DEL ESTADO N--"
  V_COD_TIPO_EST="--ETC --"
  D_FEC_HORA_ALTA="--ETC en formato mm/dd/yyyy hh:mm:ss--"
  B_ULT="--ETC --"
/>
</CONSULTA>

```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

PHP

```

<?php
$strRespuestaServer = http_post_data('http://--ip servidor--/soap',
'<?xml version="1.0" encoding="utf-8"?>
  <soap:Envelope
    xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Header>
      <ROClientIDHeader xmlns="http://tempuri.org/">
        <ID>--ID de la sesión establecida--</ID>
      </ROClientIDHeader>
    </soap:Header>
    <soap:Body>
      <WebServService__ConsEnvEstadosRef>
        <strRef>-ref-</strRef>
      </WebServService__ConsEnvEstadosRef>
    </soap:Body>
  </soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>

```

Una vez ejecutado el http_post_data la variable \$strRespuestaServer contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```

<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/"
>
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <WebServService__ConsEnvEstadosRef>
      <strRef>-ref-</strRef>
    </WebServService__ConsEnvEstadosRef>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

```
</ROClientIDHeader>
</SOAP-ENV:Header>
<SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
  <v1:WebServService___ConsEnvEstadosRefResponse>
    <v1:strEnvEstadosRef>
      <CONSULTA>
        <ENV_ESTADOS_REF
          V_SERVICIO="--CARGO+ORIGEN+ALBARÁN DEL ESTADO 1--"
          V_COD_TIPO_EST="--TIPO ESTADO--"
          D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
          B_ULT="--SI ES EL ÚLTIMO ESTADO AÑADIDO--"
        />
        .
        . //DEVOLVERÁ UN BLOQUE POR CADA ESTADO QUE TENGA ESE ENVÍO
        .
        <ENV_ESTADOS_REF
          V_SERVICIO="--CARGO+ORIGEN+ALBARÁN DEL ESTADO N--"
          V_COD_TIPO_EST="--ETC --"
          D_FEC_HORA_ALTA="--ETC en formato mm/dd/yyyy hh:mm:ss--"
          B_ULT="--ETC --"
        />
      </CONSULTA>
    </v1:strEnvEstadosRef>
  </v1:WebServService___ConsEnvEstadosRefResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

1.2.2.19 Método ConsUltimoEstadoEnvio			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodAgeCargo	Entrada	String	Agencia de cargo.
strCodAgeOri	Entrada	String	Agencia de origen.
strAlbaran	Entrada	String	Albarán.
strUltimoEstadoEnvio	Salida	String	XML con los datos del último estado del envío.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio **"LoginWSService"**.

Uso:

Si disponemos de un lenguaje que interprete los archivos .wsdl (como por ejemplo C#), simplemente tendremos que crear una referencia Web importando el fichero .wsdl y a continuación instanciar esa referencia para poder llamar a las funciones.

Ejemplo:
C#

```
WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

string xmlResult = wsWebService.ConsUltimoEstadoEnvio("agencia-cargo", "agencia-origen", "albarán");
```

Una vez ejecutada la función la variable **xmlResult** contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>
  <CONS_ULTIMO_ESTADO_ENVIO
    V_COD_TIPO_EST="--TIPO ESTADO--"
    D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
  />
```

</CONSULTA>

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

PHP

```
<?php
    $strRespuestaServer = http_post_data('http://--ip servidor--/soap',
        '<?xml version="1.0" encoding="utf-8"?>
            <soap:Envelope
                xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
                xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                xmlns:xsd="http://www.w3.org/2001/XMLSchema">
                <soap:Header>
                    <ROClientIDHeader xmlns="http://tempuri.org/">
                        <ID>--ID de la sesión establecida--</ID>
                    </ROClientIDHeader>
                </soap:Header>
                <soap:Body>
                    <WebServService___ConsUltimoEstadoEnvio>
                        <strCodAgeCargo>-agencia-cargo-</strCodAgeCargo>
                        <strCodAgeOri>-agencia-origen-</strCodAgeOri>
                        <strAlbaran>-albarán-</strAlbaran>
                    </WebServService___ConsUltimoEstadoEnvio>
                </soap:Body>
            </soap:Envelope>');
    $strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>
```

Una vez ejecutado el http_post_data la variable \$strRespuestaServer contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:HNS="http://tempuri.org/"
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:v1="http://tempuri.org/"
>
    <SOAP-ENV:Header>
        <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
            <ID>--ID SESIÓN--</ID>
        </ROClientIDHeader>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
        <v1:WebServService___ConsUltimoEstadoEnvioResponse>
            <v1:strUltimoEstadoEnvio>
                <CONSULTA>
                    <CONS_ULTIMO_ESTADO_ENVIO
```



```

V_COD_TIPO_EST="--TIPO ESTADO--"
D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
/>
</CONSULTA>
</v1:strUltimoEstadoEnvio>
</v1:WebServService__ConsUltimoEstadoEnvioResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

1.2.2.20 Método InfEnviosRef

PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
dtFecha	Entrada	DateTime	Fecha de salida de los envíos a consultar.
strRef	Entrada	String	Referencia de los envíos para filtrar
strInfEnvios	Salida	String	XML con los datos de los envíos.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "LoginWSService".

Uso:

El uso es similar a la anterior.

Ejemplo:

C#

```

WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

string xmlResult = wsWebService.InfEnviosRef("fecha-salida", "Referencia");

```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```

<CONSULTA>
<INF_ENVIOS
V_COD_AGE_CARGO="--AGENCIA DE CARGO --"
V_COD_AGE_ORI="--AGENCIA DE ORIGEN --"
V_ALBARAN="--ALBARAN --"
D_FECHA="--FECHA SALIDA en formato mm/dd/yyyy hh:mm:ss--"
V_COD_AGE_DES="--AGENCIA DE DESTINO --"
V_NOM_ORI="--NOMBRE REMITENTE--"

```

```

V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
V_POB_ORI="--POBLACION REMITENTE--"
V_CP_ORI="--COD. POSTAL REMITENTE--"
V_NOM_DES="--NOMBRE DESTINATARIO--"
V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
V_NUM_DES="--NÚMERO DESTINATARIO--"
V_PISO_DES="--PISO DESTINATARIO--"
V_POB_DES="--POBLACIÓN DESTINATARIO--"
V_CP_DES="--COD. POSTAL DESTINATARIO--"
V_TLF_DES="--TLFNO DESTINATARIO--"
D_FEC_ENTR="--FECHA ENTREGA en formato mm/dd/yyyy hh:mm:ss--"
I_BUL="--BULTOS--"
V_REF="--REFERENCIA--"
F_REEMBOLSO="--REEMBOLSO--"
F_PORTE_DEB="--PORTE DEBIDO--"
F_VALOR="--VALOR--"
V_COD_CLI="--CODIGO CLIENTE--"
V_ULT_TIPO_VIA="--ULTIMO TIPO VIA--"
V_ULT_DIR="--ULTIMO NOMBRE VIA--"
V_ULT_NUM="--ULTIMO NUMERO--"
V_ULT_PISO="--ULTIMO PISO--"
F_PESO_ORI="--PESO ORI--"
V_NOM_CLI="--NOMBRE CLIENTE--"
V_COD_CLI_DEP="--CODIGO DEPARTAMENTO--"
V_NOM_DEP="--NOMBRE DEPARTAMENTO--"
V_COD_TIPO_EST="--CÓDIGO TIPO DE ESTADO--"
D_FEC_HORA_ENTR_POD="--FECHA Y HORA DE ENTREGA DEL POD--"
V_COD_PAIS="--CÓDIGO DEL PAÍS--"
V_NOM_PAIS="--NOMBRE DEL PAÍS--"
/>

```

//DEVOLVERÁ UN BLOQUE POR CADA ENVÍO

```

<INF_ENVIOS
V_COD_AGE_CARGO="--AGENCIA DE CARGO --"
V_COD_AGE_ORI="--AGENCIA DE ORIGEN --"
V_ALBARAN="--ALBARAN N--"
D_FECHA="--ETC en formato mm/dd/yyyy hh:mm:ss--"
V_COD_AGE_DES="--AGENCIA DE DESTINO --"
V_NOM_ORI="--NOMBRE REMITENTE--"
V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
V_POB_ORI="--POBLACION REMITENTE--"
V_CP_ORI="--COD. POSTAL REMITENTE--"
V_NOM_DES="--NOMBRE DESTINATARIO--"
V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
V_NUM_DES="--NÚMERO DESTINATARIO--"
V_PISO_DES="--PISO DESTINATARIO--"
V_POB_DES="--POBLACIÓN DESTINATARIO--"
V_CP_DES="--COD. POSTAL DESTINATARIO--"
V_TLF_DES="--TLFNO DESTINATARIO--"
D_FEC_ENTR="--FECHA ENTREGA en formato mm/dd/yyyy hh:mm:ss--"
I_BUL="--BULTOS--"
V_REF="--REFERENCIA--"
F_REEMBOLSO="--REEMBOLSO--"
F_PORTE_DEB="--PORTE DEBIDO--"
F_VALOR="--VALOR--"
V_COD_CLI="--CODIGO CLIENTE--"
V_ULT_TIPO_VIA="--ULTIMO TIPO VIA--"
V_ULT_DIR="--ULTIMO NOMBRE VIA--"
V_ULT_NUM="--ULTIMO NUMERO--"
V_ULT_PISO="--ULTIMO PISO--"
F_PESO_ORI="--PESO ORI--"
V_NOM_CLI="--NOMBRE CLIENTE--"

```

```
V_COD_CLI_DEP="--CODIGO DEPARTAMENTO--"
V_NOM_DEP="--NOMBRE DEPARTAMENTO--"
V_COD_TIPO_EST="--CÓDIGO TIPO DE ESTADO--"
D_FEC_HORA_ENTR_POD="--FECHA Y HORA DE ENTREGA DEL POD--"
V_COD_PAIS="--CÓDIGO DEL PAÍS--"
V_NOM_PAIS="--NOMBRE DEL PAÍS--"
/>

</CONSULTA>
```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```
PHP

<?php
    $strRespuestaServer = http_post_data('http://--ip servidor--/soap',
        '<?xml version="1.0" encoding="utf-8"?>
        <soap:Envelope
            xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema">
            <soap:Header>
                <ROClientIDHeader xmlns="http://tempuri.org/">
                    <ID>--ID de la sesión establecida--</ID>
                </ROClientIDHeader>
            </soap:Header>
            <soap:Body>
                <WebServService__InfEnviosRef>
                    <dtFecha>--fecha-salida en formato yyyy/mm/dd-</dtFecha>
                    <strRef>--Referencia de los envios a filtrar-</strRef>
                </WebServService__InfEnviosRef>
            </soap:Body>
        </soap:Envelope>');
    $strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>
```

Una vez ejecutado el http_post_data la variable \$strRespuestaServer contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/"
>
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService___InfEnviosRefResponse>
      <v1:strInfEnvios>
        <CONSULTA>
          <INF_ENVIOS
            V_COD_AGE_CARGO="--AGENCIA DE CARGO --"
            V_COD_AGE_ORI="--AGENCIA DE ORIGEN --"
            V_ALBARAN="--ALBARAN --"
            D_FECHA="--FECHA SALIDA en formato mm/dd/yyyy hh:mm:ss--"
            V_COD_AGE_DES="--AGENCIA DE DESTINO --"
            V_NOM_ORI="--NOMBRE REMITENTE--"
            V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
            V_POB_ORI="--POBLACION REMITENTE--"
            V_CP_ORI="--COD. POSTAL REMITENTE--"
            V_NOM_DES="--NOMBRE DESTINATARIO--"
            V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
            V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
            V_NUM_DES="--NÚMERO DESTINATARIO--"
            V_PISO_DES="--PISO DESTINATARIO--"
            V_POB_DES="--POBLACIÓN DESTINATARIO--"
            V_CP_DES="--COD. POSTAL DESTINATARIO--"
            V_TLF_DES="--TLFNO DESTINATARIO--"
            D_FEC_ENTR="--FECHA ENTREGA en formato mm/dd/yyyy hh:mm:ss--"
            I_BUL="--BULTOS--"
            V_REF="--REFERENCIA--"
            F_REEMBOLSO="--REEMBOLSO--"
            F_PORTE_DEB="--PORTE DEBIDO--"
            F_VALOR="--VALOR--"
            V_COD_CLI="--CODIGO CLIENTE--"
            V_ULT_TIPO_VIA="--ULTIMO TIPO VIA--"
            V_ULT_DIR="--ULTIMO NOMBRE VIA--"
            V_ULT_NUM="--ULTIMO NUMERO--"
            V_ULT_PISO="--ULTIMO PISO--"
            F_PESO_ORI="--PESO ORI--"
            V_NOM_CLI="--NOMBRE CLIENTE--"
            V_COD_CLI_DEP="--CODIGO DEPARTAMENTO--"
            V_NOM_DEP="--NOMBRE DEPARTAMENTO--"
            V_COD_TIPO_EST="--CÓDIGO TIPO ESTADO--"
            D_FEC_HORA_ENTR_POD="--FECHA Y HORA DE ENTREGA DEL POD--"
            V_COD_PAIS="--CÓDIGO DEL PAÍS--"
            V_NOM_PAIS="--NOMBRE DEL PAÍS--"
          />

          //DEVOLVERÁ UN BLOQUE POR CADA ENVÍO

          <INF_ENVIOS
            V_COD_AGE_CARGO="--AGENCIA DE CARGO N--"
```

```
V_COD_AGE_ORI="--AGENCIA DE ORIGEN N--"
V_ALBARAN="--ALBARAN N--"
D_FECHA="--ETC en formato mm/dd/yyyy hh:mm:ss--"
V_COD_AGE_DES="--AGENCIA DE DESTINO --"
V_NOM_ORI="--NOMBRE REMITENTE--"
V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
V_POB_ORI="--POBLACION REMITENTE--"
V_CP_ORI="--COD. POSTAL REMITENTE--"
V_NOM_DES="--NOMBRE DESTINATARIO--"
V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
V_NUM_DES="--NÚMERO DESTINATARIO--"
V_PISO_DES="--PISO DESTINATARIO--"
V_POB_DES="--POBLACIÓN DESTINATARIO--"
V_CP_DES="--COD. POSTAL DESTINATARIO--"
V_TLF_DES="--TLFNO DESTINATARIO--"
D_FEC_ENTR="--FECHA ENTREGA en formato mm/dd/yyyy hh:mm:ss--"
I_BUL="--BULTOS--"
V_REF="--REFERENCIA--"
F_REEMBOLSO="--REEMBOLSO--"
F_PORTE_DEB="--PORTE DEBIDO--"
F_VALOR="--VALOR--"
V_COD_CLI="--CODIGO CLIENTE--"
V_ULT_TIPO_VIA="--ULTIMO TIPO VIA--"
V_ULT_DIR="--ULTIMO NOMBRE VIA--"
V_ULT_NUM="--ULTIMO NUMERO--"
V_ULT_PISO="--ULTIMO PISO--"
F_PESO_ORI="--PESO ORI--"
V_NOM_CLI="--NOMBRE CLIENTE--"
V_COD_CLI_DEP="--CODIGO DEPARTAMENTO--"
V_NOM_DEP="--NOMBRE DEPARTAMENTO--"
V_COD_TIPO_EST="--CÓDIGO TIPO ESTADO--"
D_FEC_HORA_ENTR_POD="--FECHA Y HORA DE ENTREGA DEL POD--"
V_COD_PAIS="--CÓDIGO DEL PAÍS--"
V_NOM_PAIS="--NOMBRE DEL PAÍS--"
/>
</CONSULTA>
</v1:strInfEnvios>
</v1:WebServService___InfEnviosRefResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

1.2.2.21 Método ConsURLDocAgrupados

PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodAgeCargo	Entrada	String	Agencia de cargo.
strCodAgeOri	Entrada	String	Agencia de origen.
strAlbaran	Entrada	String	Albarán.
strURL	Salida	String	URL de documentos agrupados.

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio **“LoginWSService”**.

Uso:

El uso es similar a la anterior.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();  
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();  
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;  
  
string strResult = wsWebService.ConsURLDocAgrupados("agencia-cargo", "agencia-origen", "albaran");
```

Una vez ejecutada la función la variable **strResult** contendrá la URL de documentos agrupados.

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```
PHP
<?php
$strRespuestaServer = http_post_data('http://--ip servidor--/soap',
'<?xml version="1.0" encoding="utf-8"?>
  <soap:Envelope
    xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Header>
      <ROClientIDHeader xmlns="http://tempuri.org/">
        <ID>--ID de la sesión establecida--</ID>
      </ROClientIDHeader>
    </soap:Header>
    <soap:Body>
      <WebServService___ConsURLDocAgrupados>
        <strCodAgeCargo>-agencia-cargo-</strCodAgeCargo>
        <strCodAgeOri>-agencia-origen-</strCodAgeOri>
        <strAlbaran>-albarán-</strAlbaran>
      </WebServService___ConsURLDocAgrupados>
    </soap:Body>
  </soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>
```

Una vez ejecutado el `http_post_data` la variable `$strRespuestaServer` contendrá la URL de documentos agrupados.

1.2.2.22 Método GrabaEnvIncActuacionLibre			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodAgeCargo	Entrada	String	Agencia de cargo.
strCodAgeOri	Entrada	String	Agencia de origen.
strAlbaran	Entrada	String	Albarán.
intIdEnvInc	Entrada	Integer	Id. de la incidencia, -1 para generar una actuación sin incidencia.
strTexto	Entrada	String	Texto para la descripción de la actuación

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

El uso es similar a la anterior.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

try
{
    wsWebService.GrabEnvIncActuacionLibre("agencia-cargo", "agencia-ori", "albarán", id,
        "texto descripción actuación")
}
catch (Exception eExcepcion)
{
    MessageBox.Show(eExcepcion.Message);
}
```


Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

PHP

```
<?php
    http_post_data('http://--ip servidor--/soap',
    '<?xml version="1.0" encoding="utf-8"?>
    <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
        <soap:Header>
            <ROClientIDHeader xmlns="http://tempuri.org/">
                <ID>--id de la sesión establecida--</ID>
            </ROClientIDHeader>
        </soap:Header>
        <soap:Body>
            <WebServService___GrabaEnvIncActuacionLibre xmlns="http://tempuri.org/">
                <strCodAgeCargo>--agencia-cargo--</strCodAgeCargo>
                <strCodAgeOri>--agencia-origen--</strCodAgeOri>
                <strAlbaran>--albarán--</strAlbaran>
                <intIdEnvInc>--id--</intIdEnvInc>
                <strTexto>--código-actuación--</strTexto>
            </WebServService___GrabaEnvIncActuacionLibre>
        </soap:Body>
    </soap:Envelope>');
?>
```

Excepciones:

Esta función devuelve las siguientes excepciones:

CÓDIGO	DESCRIPCIÓN
1	El código de actuación no existe o está inactivo.
2	El usuario no tiene permiso para meter actuaciones.
3	El envío no existe o el usuario no tiene permiso para modificarlo.
4	El usuario debe pertenecer a la agencia de cargo.
5	No se puede introducir una actuación sin incidencia si el envío ya tiene incidencias.
6	La incidencia que se genera con la actuación no está configurada.
7	El usuario debe pertenecer a la agencia de cargo o a la agencia de destino.
8	La incidencia indicada no existe.
9	El tipo de servicio del envío no permite grabar estados, incidencias o actuaciones sin incidencias

Para saber cómo controlar las excepciones devueltas por favor consulte el apartado 1.2.3, Excepciones.

1.2.2.23 Método GrabaRecIncActuacionLibre			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strCodRec	Entrada	String	Recogida.
intIdRecInc	Entrada	Integer	Id. de la incidencia. -1 para generar una actuación sin incidencia.
strTexto	Entrada	String	Texto para la descripción de la actuación

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

El uso es similar a la anterior.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

try
{
    wsWebService.GrabaRecIncActuacionLibre("recogida", id, "texto descripción actuación")
}
catch (Exception eExcepcion)
{
    MessageBox.Show(eExcepcion.Message);
}
```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

PHP

```
<?php
    http_post_data('http://--ip servidor--/soap',
'<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Header>
        <ROClientIDHeader xmlns="http://tempuri.org/">
            <ID>--id de la sesión establecida--</ID>
        </ROClientIDHeader>
    </soap:Header>
    <soap:Body>
        <WebServService__GrabaReclncActuacionLibre xmlns="http://tempuri.org/">
            <strCodRec>--recogida--</strCodRec>
            <intIdReclnc>--id--</intIdReclnc>
            <strTexto>--código-actuación--</strTexto>
        </WebServService__GrabaReclncActuacionLibre>
    </soap:Body>
</soap:Envelope>');
?>
```

Excepciones:

Esta función devuelve las siguientes excepciones:

CÓDIGO	DESCRIPCIÓN
1	El código de actuación no existe o está inactivo.
2	El usuario no tiene permiso para meter actuaciones.
3	La recogida no existe o el usuario no tiene permiso para modificarla.
4	El usuario debe pertenecer a la agencia de cargo o a la agencia de origen.
5	La incidencia indicada no existe.
6	El usuario debe pertenecer a la agencia de origen para poder añadir la incidencia.
7	La incidencia que se genera con la actuación no está configurada.

Para saber cómo controlar las excepciones devueltas por favor consulte el apartado 1.2.3, Excepciones.

1.2.2.24 Método ConsRecogidasRef			
PARÁMETRO	E/S	TIPO	DESCRIPCIÓN
strRef	Entrada	String	Referencia.
strRecogida	Salida	String	XML con los datos de las recogidas que tengan como referencia el valor pasado por el parámetro anterior

ATENCIÓN:

Este método requiere sesión, así que es necesario que antes de utilizarlo realice el login correspondiente mediante el servicio "[LoginWSService](#)".

Uso:

El uso es similar a la anterior.

Ejemplo:

C#

```
WebService.WebServService wsWebService = new WebService.WebServService();
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();
wsWebService.ROClientIDHeaderValue.ID = --ID DE SESIÓN RECOGIDO EN EL LOGIN--;

string xmlResult = wsWebService.ConsRecogidasRef("referencia");
```

Una vez ejecutada la función la variable `xmlResult` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```
<CONSULTA>
<RECOGIDAS
  V_COD="--CÓDIGO DE RECOGIDA--"
  V_COD_AGE_ORI="--AGENCIA DE ORIGEN--"
  V_COD_AGE_SOL="--AGENCIA QUE SOLICITA--"
  V_COD_AGE_DES="--AGENCIA DE DESTINO--"
  V_COD_AGE_CARGO="--AGENCIA DE CARGO--"
  D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
  D_FEC_REC="--FECHA RECOGIDA en formato mm/dd/yyyy hh:mm:ss--"
  SD_HORA_REC_INI="--HORA INICIO RECOGIDA en formato mm/dd/yyyy hh:mm:ss--"
  SD_HORA_REC_FIN="--HORA FIN RECOGIDA en formato mm/dd/yyyy hh:mm:ss--"
  I_BUL="--BULTOS--"
  F_PESO="--PESO--"
```

```

F_VALOR="--VALOR--"
F_ANTICIPO="--ANTICIPO--"
V_COD_VEH="--CÓDIGO DEL VEHÍCULO--"
V_NOM_ORI="--NOMBRE REMITENTE--"
V_TIPO_VIA_ORI="--TIPO VÍA REMITENTE--"
V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
V_NUM_ORI="--NÚMERO VÍA REMITENTE--"
V_PISO_ORI="--PISO REMITENTE--"
V_POB_ORI="--POBLACION REMITENTE--"
V_CP_ORI="--COD. POSTAL REMITENTE--"
V_TLF_ORI="--TLFNO. REMITENTE--"
V_COD_PRO_ORI="--PROVINCIA REMITENTE--"
V_NOM_DES="--NOMBRE DESTINATARIO--"
V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"
V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
V_NUM_DES="--NÚMERO DESTINATARIO--"
V_PISO_DES="--PISO DESTINATARIO--"
V_POB_DES="--POBLACIÓN DESTINATARIO--"
V_CP_DES="--COD. POSTAL DESTINATARIO--"
V_TLF_DES="--TLFNO DESTINATARIO--"
V_COD_PRO_DES="--PROVINCIA DESTINATARIO--"
V_OBS="--OBSERVACIONES--"
V_COD_CLI="--CÓDIGO DE CLIENTE--"
V_COD_CLI_DEP="--CÓDIGO DE DEPARTAMENTO--"
V_TIPO_REC="--TIPO RECOGIDA--"
V_PERS_CONTACTO="--PERSONA CONTACTO--"
B_AUT_KM="--AUTORIZA KILÓMETROS--"
V_COD_TIPO_SERV="--TIPO SERVICIO--"
B_SABADO="--SABADO--"
V_COD_REP="--CÓDIGO REPARTIDOR--"
V_COD_USU_ALTA="--USUARIO ALTA--"
V_COD_ENV="--CÓDIGO DE ENVÍO--"
SD_HORA_REC_INI_TARDE="--HORA INICIO RECOGIDA TARDE en formato mm/dd/yyyy hh:mm:ss--"
SD_HORA_REC_FIN_TARDE="--HORA FIN RECOGIDA TARDE en formato mm/dd/yyyy hh:mm:ss--"
V_COD_AGE_ALTA="--AGENCIA ALTA--"
V_COD_CLI_ALTA="--CLIENTE ALTA--"
V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
/>
</CONSULTA>

```

Otra alternativa de uso para lenguajes que no soporten .wsdl consiste en lanzar una petición POST a la dirección y puerto deseados pasando un XML que lleve los datos de entrada, y el servidor nos devolverá los datos de salida.

Ejemplo:

```

PHP

<?php
    $strRespuestaServer = http_post_data('http://--ip servidor--/soap',
        '<?xml version="1.0" encoding="utf-8"?>
        <soap:Envelope
            xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema">
            <soap:Header>
                <ROClientIDHeader xmlns="http://tempuri.org/">

```

```

        <ID>--id de la sesión establecida--</ID>
    </ROClientIDHeader>
</soap:Header>
<soap:Body>
    <WebServService___ConsRecogidasRef>
        <strRef>-Referencia-</strRef>
    </WebServService___ConsRecogidasRef>
</soap:Body>
</soap:Envelope>');
$strRespuestaServer = http_parse_message($strRespuestaServer)->body;
?>

```

Una vez ejecutado el `http_post_data` la variable `$strRespuestaServer` contendrá un XML de respuesta con el siguiente formato, que habrá que tratar para extraer los valores:

XML

```

<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:v1="http://tempuri.org/">
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID SESIÓN--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <v1:WebServService___ConsRecogidasRefResponse>
      <v1:strRef>
        <CONSULTA>
          <RECOGIDAS
            V_COD="--CÓDIGO DE RECOGIDA--"
            V_COD_AGE_ORI="--AGENCIA DE ORIGEN--"
            V_COD_AGE_SOL="--AGENCIA QUE SOLICITA--"
            V_COD_AGE_DES="--AGENCIA DE DESTINO--"
            V_COD_AGE_CARGO="--AGENCIA DE CARGO--"
            D_FEC_HORA_ALTA="--FECHA/HORA ALTA en formato mm/dd/yyyy hh:mm:ss--"
            D_FEC_REC="--FECHA RECOGIDA en formato mm/dd/yyyy hh:mm:ss--"
            SD_HORA_REC_INI="--HORA INICIO RECOGIDA en formato mm/dd/yyyy hh:mm:ss--"
            SD_HORA_REC_FIN="--HORA FIN RECOGIDA en formato mm/dd/yyyy hh:mm:ss--"
            I_BUL="--BULTOS--"
            F_PESO="--PESO--"
            F_VALOR="--VALOR--"
            F_ANTICIPO="--ANTICIPO--"
            V_COD_VEH="--CÓDIGO DEL VEHÍCULO--"
            V_NOM_ORI="--NOMBRE REMITENTE--"
            V_TIPO_VIA_ORI="--TIPO VÍA REMITENTE--"
            V_DIR_ORI="--NOMBRE VÍA REMITENTE--"
            V_NUM_ORI="--NÚMERO VÍA REMITENTE--"
            V_PISO_ORI="--PISO REMITENTE--"
            V_POB_ORI="--POBLACION REMITENTE--"
            V_CP_ORI="--COD. POSTAL REMITENTE--"
            V_TLF_ORI="--TLFNO. REMITENTE--"
            V_COD_PRO_ORI="--PROVINCIA REMITENTE--"
            V_NOM_DES="--NOMBRE DESTINATARIO--"
            V_TIPO_VIA_DES="--TIPO VÍA DESTINATARIO--"

```

```

V_DIR_DES="--DIRECCIÓN DESTINATARIO--"
V_NUM_DES="--NÚMERO DESTINATARIO--"
V_PISO_DES="--PISO DESTINATARIO--"
V_POB_DES="--POBLACIÓN DESTINATARIO--"
V_CP_DES="--COD. POSTAL DESTINATARIO--"
V_TLF_DES="--TLFNO DESTINATARIO--"
V_COD_PRO_DES="--PROVINCIA DESTINATARIO--"
V_OBS="--OBSERVACIONES--"
V_COD_CLI="--CÓDIGO DE CLIENTE--"
V_COD_CLI_DEP="--CÓDIGO DE DEPARTAMENTO--"
V_TIPO_REC="--TIPO RECOGIDA--"
V_PERS_CONTACTO="--PERSONA CONTACTO--"
B_AUT_KM="--AUTORIZA KILÓMETROS--"
V_COD_TIPO_SERV="--TIPO SERVICIO--"
B_SABADO="--SABADO--"
V_COD_REP="--CÓDIGO REPARTIDOR--"
V_COD_USU_ALTA="--USUARIO ALTA--"
V_COD_ENV="--CÓDIGO DE ENVÍO--"
SD_HORA_REC_INI_TARDE="--HORA INICIO RECOGIDA TARDE en formato mm/dd/yyyy
hh:mm:ss--"
SD_HORA_REC_FIN_TARDE="--HORA FIN RECOGIDA TARDE en formato mm/dd/yyyy
hh:mm:ss--"
V_COD_AGE_ALTA="--AGENCIA ALTA--"
V_COD_CLI_ALTA="--CLIENTE ALTA--"
V_COD_CLI_DEP_ALTA="--DEPARTAMENTO ALTA--"
/>
</CONSULTA>
</v1:strRef>
</v1:WebServService___ConsRecogidasRefResponse>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

1.2.3 Excepciones

Cuando se produzca una excepción en el servidor éste nos devolverá un XML con el siguiente formato del que tendremos que sacar los valores:

```

XML
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:HNS="http://tempuri.org/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Header>
    <ROClientIDHeader SOAP-ENV:mustUnderstand="0" xmlns="urn:DinaPaq">
      <ID>--ID-SESSION--</ID>
    </ROClientIDHeader>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body xmlns:ro="urn:DinaPaq">
    <SOAP-ENV:Fault>
      <faultcode>Exception</faultcode>
      <faultstring>--DESCRIPCION-EXCEPCION--</faultstring>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

En el caso de que el lenguaje que utilizemos disponga de cliente de servicios Web, es muy probable que la interpretación de este XML sea automática y sea el propio cliente el que lance una excepción nativa en el programa, que deberemos controlar:

La excepción siempre va a estar compuesta por un código (numérico), seguida de dos puntos (:), y a continuación la descripción de la excepción.

Ejemplo:

```
WebService.WebServService wsWebService = new WebService.WebServService();  
wsWebService.ROClientIDHeaderValue = new WebService.ROClientIDHeader();  
wsWebService.ROClientIDHeaderValue.ID = strSesion;
```

```
strCodOut = wsWebService.GrabarRecogida("", "", "000001",
```

