



# EBI BROKER

## Guía de Usuario Web Service

Título del documento:	EBI BROKER
Nombre del fichero:	ES_EBIBROKER webServices.odt
Versión:	1.0
Estado:	VIGENTE
Fecha:	10/09/2013.
Autor:	jvalero

**Revisión, Aprobación**

Revisado por:		Fecha:
Aprobado por:		Fecha:

**Historial de cambios**

Versión	Fecha	Descripción de la acción	Páginas
1.0	10/09/2013	Versión inicial	
1.1	17/09/2013	Se añade el método transformAndPublishWS	
1.2	14/03/2014	Se modifican la url que se deben emplear en función del grupo en el que este el cliente	
1.3	15/09/2014	Se añade la información de la función publishDBWS	

# Índice de contenido

<b>1EBIBROKERWS.....</b>	<b>6</b>
1.1Introducción.....	6
1.1.1Acceso al servicio.....	6
1.1.2Funciones del webservice ebibrokerws.....	6
<b>2Funciones EBibrokerws.....</b>	<b>7</b>
2.1Publicación de mensajes.....	7
2.1.1Llamada:.....	7
2.1.2Parámetros:.....	7
2.1.3Respuesta:.....	8
2.1.4Códigos de error:.....	8
2.1.5Ejemplos.....	9
2.2PUBLICACIONES CONFIRMADAS.....	10
2.2.1Llamada:.....	10
2.2.2Parámetros:.....	10
2.2.3Respuesta:.....	10
2.2.4Códigos de error:.....	11
2.2.5Ejemplos.....	11
2.3NOTIFICAR PUBLICACIONES.....	12
2.3.1Llamada:.....	12
2.3.2Parámetros:.....	12
2.3.3Respuesta:.....	13
2.3.4Códigos de error:.....	13
2.3.5Ejemplos.....	13
2.4PUBLICACIONES RECHAZADAS.....	14
2.4.1Llamada:.....	14
2.4.2Parámetros:.....	14
2.4.3Respuesta:.....	15
2.4.4Códigos de error:.....	15
2.4.5Ejemplos.....	15
2.5NOTIFICAR PUBLICACIONES RECHAZADAS.....	16
2.5.1Llamada:.....	16
2.5.2Parámetros:.....	16
2.5.3Respuesta:.....	17
2.5.4Códigos de error:.....	17
2.5.5Ejemplos.....	17
2.6RECOGER SUSCRIPCIONES.....	18
2.6.1Llamada:.....	18
2.6.2Parámetros:.....	18
2.6.3Respuesta:.....	19
2.6.4Códigos de error:.....	19
2.6.5Ejemplos.....	20
2.7CONFIRMAR SUSCRIPCIONES.....	21
2.7.1Llamada:.....	21
2.7.2Parámetros:.....	21
2.7.3Respuesta:.....	22
2.7.4Códigos de error:.....	22
2.7.5Ejemplos.....	22

2.8RECHAZAR SUSCRIPCIONES.....	23
2.8.1Llamada:.....	23
2.8.2Parámetros:.....	24
2.8.3Respuesta:.....	24
2.8.4Códigos de error:.....	24
2.8.5Ejemplos.....	24
<b>3EBIBROKERWSv2.....</b>	<b>26</b>
3.1Introducción.....	26
3.1.1Acceso al servicio.....	26
3.1.2Funciones del webservice ebibrokerwsv2.....	26
<b>4Funciones EBIbrokerwsv2.....</b>	<b>29</b>
4.1Publicación de mensajes.....	29
4.1.1Llamada:.....	29
4.1.2Parámetros:.....	29
4.1.3Respuesta:.....	30
4.1.4Códigos de error:.....	30
4.1.5Ejemplos.....	31
4.2publicacion de mensajes con origen base de datos.....	32
4.2.1Llamada:.....	32
4.2.2Parámetros:.....	32
4.2.3Respuesta:.....	33
4.2.4Códigos de error:.....	34
4.2.5Ejemplos.....	34
4.3TRANSFORMACION Y Publicación de mensajes.....	35
4.3.1Llamada:.....	35
4.3.2Parámetros:.....	36
4.3.3Respuesta:.....	37
4.3.4Códigos de error:.....	37
4.3.5Ejemplos.....	37
4.4PUBLICACIONES CONFIRMADAS.....	39
4.4.1Llamada:.....	39
4.4.2Parámetros:.....	40
4.4.3Respuesta:.....	40
4.4.4Códigos de error:.....	40
4.4.5Ejemplos.....	40
4.5NOTIFICAR PUBLICACIONES.....	42
4.5.1Llamada:.....	42
4.5.2Parámetros:.....	42
4.5.3Respuesta:.....	42
4.5.4Códigos de error:.....	42
4.5.5Ejemplos.....	43
4.6PUBLICACIONES RECHAZADAS.....	43
4.6.1Llamada:.....	43
4.6.2Parámetros:.....	44
4.6.3Respuesta:.....	44
4.6.4Códigos de error:.....	44
4.6.5Ejemplos.....	44
4.7NOTIFICAR PUBLICACIONES RECHAZADAS.....	46
4.7.1Llamada:.....	46

4.7.2Parámetros:	46
4.7.3Respuesta:	46
4.7.4Códigos de error:	47
4.7.5Ejemplos:	47
4.8RECOGER SUSCRIPCIONES:	47
4.8.1Llamada:	48
4.8.2Parámetros:	48
4.8.3Respuesta:	48
4.8.4Códigos de error:	49
4.8.5Ejemplos:	49
4.9CONFIRMAR SUSCRIPCIONES:	50
4.9.1Llamada:	51
4.9.2Parámetros:	51
4.9.3Respuesta:	51
4.9.4Códigos de error:	51
4.9.5Ejemplos:	52
4.10RECHAZAR SUSCRIPCIONES:	53
4.10.1Llamada:	53
4.10.2Parámetros:	53
4.10.3Respuesta:	54
4.10.4Códigos de error:	54
4.10.5Ejemplos:	54
<b>5anexo. Acceso a edicomnet:</b>	<b>56</b>
<b>6Anexo: WSDL ebibroker:</b>	<b>57</b>
6.1 EBIBROKERWS:	57
6.2EBIBROKERWSV2:	57

# 1 EBIBROKERWS

## 1.1 INTRODUCCIÓN

El webService **EBIBrokerWS**, permite el acceso al EBIBROKER, poder realizar operaciones envío y recepción de mensajes.

Este documento ofrece la WSDL del servicio y la información necesaria para que para interactuar con el servicio EBIBrokerWS y desarrollar aplicaciones compatibles.

### 1.1.1 ACCESO AL SERVICIO

La URL de acceso al webservice EBIBrokerWS del EBI-BROKER es la siguiente:

<https://aspXX.sedeb2b.com:9020/?wsdl>

donde XX corresponde con el número de grupo asociado al cliente, para los grupos entre el 35 y el 40 inclusive, se debe utilizar aspebi.sedeb2b.com

### 1.1.2 FUNCIONES DEL WEBSERVICE EBIBROKERWS

Función	Descripción	ver.
TEBIBrokerInterface PublishWS	Esta función permite publicar mensajes en el EBI-BROKER vía web service, utilizando SOAP Attachment, para enviar los fichero a publicar	
TEBIBrokerInterface_ConfirmedPublicationsWS	Devuelve las publicaciones que han sido recuperadas por el destinatario	
TEBIBrokerInterface_NotifiedPublicationWS	Emite un acuse de la publicación recuperada por el destinatario	
TEBIBrokerInterface_RejectedPublicationsWS	Devuelve las publicaciones que han sido rechazadas por el destinatario	
TEBIBrokerInterface_NotifiedRejectedPublicationWS	Emite un acuse de la publicación rechazada	
TEBIBrokerInterface_SubscriptionWS	Permite recuperar suscripciones, aplicándoles o no una transformación	
TEBIBrokerInterface_ConfirmSubscriptionWS	Confirma de forma positiva la entrega de los mensajes que se han recuperado	
TEBIBrokerInterface_RejectSubscriptionWS	Rechaza la suscripción recogida del sistema (es una confirmación negativa).	

## 2 FUNCIONES EBIBROKERWS

### 2.1 PUBLICACIÓN DE MENSAJES

---

#### TEBIBrokerInterface\_PublishWS

Esta función permitirá publicar mensajes en el EBIBROKER

#### 2.1.1 LLAMADA:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_PublishWS>
      <aClientId>?</aClientId>
      <aUser>?</aUser>
      <aPassword>?</aPassword>
      <aDomain>?</aDomain>
      <aApplication>?</aApplication>
      <aSchema>?</aSchema>
      <aDestination>?</aDestination>
      <aReference>?</aReference>
      <aMessage>?</aMessage>
      <aMessageFormat>?</aMessageFormat>
      <aDuplicates>?</aDuplicates>
    </ueb:TEBIBrokerInterface_PublishWS>
  </soapenv:Body>
</soapenv:Envelope>
```

#### 2.1.2 PARÁMETROS:

- **aClientId:** identificador del cliente
- **aUser:** usuario que realiza la publicación
- **aPassword:** contraseña del usuario que realiza la publicación
- **aDomain:** dominio donde se va a publicar el documento
- **aApplication:** aplicación en la que se quiere publicar el documento
- **aSchema:** nombre del esquema con el que se publicará el documento
- **aDestination:** destino de la publicación del documento
- **aReference:** referencia con la que se publicará el documento

- **aMessage:** Content-ID del adjunto SOAP con el mensaje, o mensaje codificado en base 64.
- **aMessageFormat:** Indica el formato con el que se esta enviado el mensaje:
  - 1: Codificado en base64 en la variable aMessage.
  - 10: Como SOAP Attachament.

Content-Type: application/x-gzip

Content-Transfer-Encoding: base64
- **aDuplicates:** Indica que acción se debe tomar ante documentos duplicados
  - 0: dupIgnore. La publicación siempre se genera, incluso si ya existe una ya duplicada en el sistema.
  - 1: dupAccept. En caso de ya existir una publicación anterior similar (con el mismo HASH), no se genera la nueva publicación, ni sus suscripciones correspondientes.
  - 2: dupError. En caso de ya existir una publicación anterior devuelve un código de error
  - 3: dupOverride. En caso de ya existir una publicación, esta y sus suscripciones se borran, y se crea una nueva con el mensaje que se esta publicando.

Por publicación duplicada, se entiende, aquellos documentos con el mismo HASH, dentro del mismo dominio y aplicación.

### 2.1.3 RESPUESTA:

- **Respuesta satisfactoria:** Identificador de la publicación que se ha realizado en el EBIBROKER
- **Respuesta de error:** Si se produce un error, se generará un código de error, que indicará el tipo de error que se ha producido.

### 2.1.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la publicación
-2	Publicación no autorizada
-3	Aplicación destino no encontrada
-4	Documento incorrecto acorde a la validación sintáctica del esquema
-5	La aplicación destino no acepta la suscripción



-7	Usuario no autorizado a la suscripción
-8	Publicación duplicada
-9	El dominio no permite publicaciones BROADCAST
-10	El documento no se ha insertado correctamente en la base de datos
-11	Problema generando las suscripciones
-12	Error registrando la sesión
-16	Tamaño de la publicación excedido

## 2.1.5 EJEMPLOS

### **Petición SOAP:**

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_PublishWS>
      <aClientId>TEST</aClientId>
      <aUser>SUPER</aUser>
      <aPassword>ADMIN1</aPassword>
      <aDomain>PRODUCCION</aDomain>
      <aApplication>EDI5</aApplication>
      <aSchema>TEST MSG</aSchema>
      <aDestination>EDI5</aDestination>
      <aReference></aReference>
      <aMessage>aGVsbG8gd29ybGQK</aMessage>
      <aMessageFormat>1</aMessageFormat>
      <aDuplicates>0</aDuplicates>
    </ueb:TEBIBrokerInterface_PublishWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### **Respuesta SOAP:**

El siguiente ejemplo es una respuesta de publicación **satisfactoria**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:TEBIBrokerInterface_PublishWSResponse
xmlns:ns1="UEBIBrokerInterface">
      <return>1309121115086529616</return>
    </ns1:TEBIBrokerInterface_PublishWSResponse>
  </soap:Body>
</soap:Envelope>
```

El siguiente ejemplo es una respuesta de **error**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
```

```

    <soap:Body>
      <ns1:TEBIBrokerInterface_PublishWSResponse
xmlns:ns1="UEBIBrokerInterface">
        <return>-2</return>
      </ns1:TEBIBrokerInterface_PublishWSResponse>
    </soap:Body>
  </soap:Envelope>

```

## 2.2 PUBLICACIONES CONFIRMADAS

---

**TEBIBrokerInterface\_ConfirmedPublicationsWS**

Recupera las publicaciones que han sido recuperadas por el destinatario

### 2.2.1 LLAMADA:

```

<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_ConfirmedPublicationsWS>
      <aClientId>?</aClientId>
      <aUser>?</aUser>
      <aPassword>?</aPassword>
      <aDomain>?</aDomain>
      <aApplication>?</aApplication>
    </ueb:TEBIBrokerInterface_ConfirmedPublicationsWS>
  </soapenv:Body>
</soapenv:Envelope>

```

### 2.2.2 PARÁMETROS:

- **aClientId:** Identificador del cliente
- **aUser:** usuario que realiza la petición
- **aPassword:** contraseña del usuario que realiza la petición
- **aDomain:** dominio en el cual el documento fue publicado
- **aApplication:** aplicación en la que la cual el documento fue publicado

### 2.2.3 RESPUESTA:

- **Respuesta satisfactoria:** Lista con los identificadores de los mensajes que han sido satisfactoriamente recuperados.

- **Respuesta de error:** Si produce un error, se devolverá una excepción del tipo `EBIBrokerServerException` la cual indicará el código y el motivo del error.

## 2.2.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la publicación
-12	Usuario no válido

## 2.2.5 EJEMPLOS

### Petición SOAP:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_ConfirmedPublicationsWS>
      <aClientId>CLIENTID</aClientId>
      <aUser>SUPER</aUser>
      <aPassword>READMIN</aPassword>
      <aDomain>PRODUCCION</aDomain>
      <aApplication>EDIWIN</aApplication>
    </ueb:TEBIBrokerInterface_ConfirmedPublicationsWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### Respuesta SOAP:

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:TEBIBrokerInterface_ConfirmedPublicationsWSResponse
xmlns:ns1="UEBIBrokerInterface">
      <return>
        <item>1305130810353387103</item>
        <item>1305130807244739416</item>
        <item>1305130806227290424</item>
        <item>1309121136002733867</item>
      </return>
    </ns1:TEBIBrokerInterface_ConfirmedPublicationsWSResponse>
  </soap:Body>
```

El siguiente ejemplo es una respuesta de **error**.

```

<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:-999</faultcode>
      <faultstring>Usuario no valido</faultstring>
      <detail>
        <ns1:EBIBrokerServerException
xmlns:ns1="UEBIBrokerInterface">
          <message>Usuario no valido</message>
          <number xsi:type="xs:int"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">-
12</number>
        </ns1:EBIBrokerServerException>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>

```

## 2.3 NOTIFICAR PUBLICACIONES

---

**TEBIBrokerInterface\_NotifiedPublicationWS**

Emite un acuse de entrega de la confirmación de la publicación

### 2.3.1 LLAMADA:

```

<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_NotifiedRejectedPublicationWS>
      <aClientId>?</aClientId>
      <aUser>?</aUser>
      <aPassword>?</aPassword>
      <aDomain>?</aDomain>
      <aApplication>?</aApplication>
      <aMessageIds>
        <!--Zero or more repetitions:-->
        <item>?</item>
      </aMessageIds>
    </ueb:TEBIBrokerInterface_NotifiedRejectedPublicationWS>
  </soapenv:Body>
</soapenv:Envelope>

```

### 2.3.2 PARÁMETROS:

- **aClientId**: Identificador del cliente
- **aUser**: usuario que realiza la petición
- **aPassword**: contraseña del usuario que realiza la petición
- **aDomain**: dominio donde se encuentra la publicación confirmada
- **aApplication**: aplicación donde se encuentra la publicación confirmada
- **aMessageIds**: lista con los identificadores de la publicaciones confirmadas, a emitir su acuse de entrega

### 2.3.3 RESPUESTA:

- **Respuesta satisfactoria**: *True* si la operación se realiza con éxito
- **Respuesta de error**: *False* si se produce un error, o se devolverá una excepción del tipo `EBIBrokerServerException` la cual indicará el código y el motivo del error.

### 2.3.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la publicación
-12	Usuario no válido

### 2.3.5 EJEMPLOS

#### Petición SOAP:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_NotifiedPublicationWS>
      <aClientId>TEST</aClientId>
      <aUser>SUPER</aUser>
      <aPassword>RADMIN</aPassword>
      <aDomain>PRODUCCION</aDomain>
      <aApplication>EDI5</aApplication>
      <aMessageIds>
        <item>1309121134277330990</item>
      </aMessageIds>
    </ueb:TEBIBrokerInterface_NotifiedPublicationWS>
  </soapenv:Body>
</soapenv:Envelope>
```

**Respuesta SOAP:**

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  <soap:Body>
    <ns1:TEBIBrokerInterface_NotifiedPublicationWSResponse
xmlns:ns1="UEBIBrokerInterface">
      <return>true</return>
    </ns1:TEBIBrokerInterface_NotifiedPublicationWSResponse>
  </soap:Body>
</soap:Envelope>
```

## 2.4 PUBLICACIONES RECHAZADAS

---

**TEBIBrokerInterface\_RejectedPublicationsWS**

Recupera las publicaciones que han sido rechazadas por el destinatario

### 2.4.1 LLAMADA:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_RejectedPublicationsWS>
      <aClientId?></aClientId>
      <aUser?></aUser>
      <aPassword?></aPassword>
      <aDomain?></aDomain>
      <aApplication?></aApplication>
    </ueb:TEBIBrokerInterface_RejectedPublicationsWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.4.2 PARÁMETROS:

- **aClientId**: Identificador del cliente
- **aUser**: usuario que realiza la petición
- **aPassword**: contraseña del usuario que realiza la petición
- **aDomain**: dominio en el cual el documento fue publicado
- **aApplication**: aplicación en la que la cual el documento fue publicado

### 2.4.3 RESPUESTA:

- **Respuesta satisfactoria:** Lista con los identificadores de los mensajes que han sido rechazados.
- **Respuesta de error:** Si produce un error, se devolverá una excepción del tipo `EBIBrokerServerException` la cual indicará el código y el motivo del error.

### 2.4.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la publicación
-12	Usuario no válido

### 2.4.5 EJEMPLOS

#### Petición SOAP:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_RejectedPublicationsWS>
      <aClientId>CLIENTE</aClientId>
      <aUser>SUPER</aUser>
      <aPassword>RADMIN</aPassword>
      <aDomain>PRODUCCION</aDomain>
      <aApplication>EDI5</aApplication>
    </ueb:TEBIBrokerInterface_RejectedPublicationsWS>
  </soapenv:Body>
</soapenv:Envelope>
```

#### Respuesta SOAP:

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:TEBIBrokerInterface_RejectedPublicationsWSResponse
xmlns:ns1="UEBIBrokerInterface">
      <return>
        <item>1309121115086529616</item>
      </return>
    </ns1:TEBIBrokerInterface_RejectedPublicationsWSResponse>
  </soap:Body>
</soap:Envelope>
```

El siguiente ejemplo es una respuesta **errónea**.

```

<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:-999</faultcode>
      <faultstring>Usuario no valido</faultstring>
      <detail>
        <ns1:EBIBrokerServerException
xmlns:ns1="UEBIBrokerInterface">
          <message>Usuario no valido</message>
          <number xsi:type="xs:int"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">-
12</number>
        </ns1:EBIBrokerServerException>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>

```

## 2.5 NOTIFICAR PUBLICACIONES RECHAZADAS

**TEBIBrokerInterface\_NotifiedRejectedPublicationWS**

Emite un acuse de entrega del rechazo de la publicación

### 2.5.1 LLAMADA:

```

<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_NotifiedRejectedPublicationWS>
      <aClientId>?</aClientId>
      <aUser>?</aUser>
      <aPassword>?</aPassword>
      <aDomain>?</aDomain>
      <aApplication>?</aApplication>
      <aMessageIds>
        <item>?</item>
      </aMessageIds>
    </ueb:TEBIBrokerInterface_NotifiedRejectedPublicationWS>
  </soapenv:Body>
</soapenv:Envelope>

```

### 2.5.2 PARÁMETROS:



- **aClientId**: Identificador del cliente
- **aUser**: usuario que realiza la petición
- **aPassword**: contraseña del usuario que realiza la petición
- **aDomain**: dominio donde se encuentra la publicación rechazada
- **aApplication**: aplicación donde se encuentra la publicación rechazada
- **aMessageIds**: lista con los identificadores de la publicaciones rechazadas, a emitir su acuse de entrega

### 2.5.3 RESPUESTA:

- **Respuesta satisfactoria**: *True* si la operación se realiza con éxito
- **Respuesta de error**: *False* si se produce un error, o se devolverá una excepción del tipo `EBIBrokerServerException` la cual indicará el código y el motivo del error.

### 2.5.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la publicación
-12	Usuario no válido

### 2.5.5 EJEMPLOS

#### Petición SOAP:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_NotifiedRejectedPublicationWS>
      <aClientId>CLIENTID</aClientId>
      <aUser>SUPER</aUser>
      <aPassword>RADMIN</aPassword>
      <aDomain>PRODUCCION</aDomain>
      <aApplication>EDI5</aApplication>
      <aMessageIds>
        <item>1309121134277330990</item>
        <item>1309121134323630807</item>
      </aMessageIds>
    </ueb:TEBIBrokerInterface_NotifiedRejectedPublicationWS>
  </soapenv:Body>
</soapenv:Envelope>
```

```
</soapenv:Body>
</soapenv:Envelope>
```

### Respuesta SOAP:

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  <soap:Body>

<ns1:TEBIBrokerInterface_NotifiedRejectedPublicationWSResponse
xmlns:ns1="UEBIBrokerInterface">
  <return>true</return>
</ns1:TEBIBrokerInterface_NotifiedRejectedPublicationWSResponse>
  </soap:Body>
</soap:Envelope>
```

## 2.6 RECOGER SUSCRIPCIONES

**TEBIBrokerInterface\_SubscriptionWS**

Recoge los mensajes que estén en estado suscrito

### 2.6.1 LLAMADA:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_SubscriptionWS>
      <aClientId?></aClientId>
      <aUser?></aUser>
      <aPassword?></aPassword>
      <aDomain?></aDomain>
      <aApplication?></aApplication>
      <aSchema?></aSchema>
      <aTransformation?></aTransformation>
      <aReference?></aReference>
      <aMessage?></aMessage>
      <aMessageFormat?></aMessageFormat>
    </ueb:TEBIBrokerInterface_SubscriptionWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### 2.6.2 PARÁMETROS:

- **aClientId**: identificador del cliente
- **aUser**: usuario que realiza la suscripción

- **aPassword:** contraseña del usuario que realiza la suscripción
- **aDomain:** dominio donde se encuentra la suscripción a recoger
- **aApplication:** aplicación donde se encuentra la suscripción
- **aSchema:** esquema del cual se quiere recoger la suscripción
- **aTransformation:** transformación que se desea aplicar a la suscripción, antes de recogerla. Este valor es opcional, sino se indica transformación se recogerá la suscripción tal cual se haya publicado.
- **aReference:** referencia del mensaje que se quiere recuperar, sino se indica ninguna, se recogerá la primera suscripción que este disponible.
- **aMessage:** este parámetro es de **salida**, y en el caso que el parámetro aMessageFormat se igual a 1, viene la suscripción o el resultado de la transformación (solo el primer fichero) codificado en base 64,
- **aMessageFormat:** indica el formato en el que la respuesta será devuelta al cliente:
  - 1: Codificada en base 64 en la variable aMessage. Si se aplica transformación, solo se devolverá el primer documento resultado de la transformación.
  - 10: Como adjuntos a la respuesta (SAOP Attachment)

### 2.6.3 RESPUESTA:

- **Respuesta satisfactoria:** Si la respuesta es satisfactoria, devolverá como respuesta, el identificador de la suscripción recuperada, así como la suscripción propiamente dicha, bien como adjuntos (SOAP Attachments) o bien codificada en base 64 en la variable aMessage. Si no hay suscripciones, pendiente de recoger, devolverá un identificador vacío.
- **Respuesta de error:** Si se produce algún error devolverá el código de error, indicando el motivo.

### 2.6.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la suscripción
-2	Publicación no autorizada a recoger la suscripción
-3	Transformación no encontrada
-4	Error realizando la transformación

-7	Usuario no autorizado a la suscripción
-12	Problema generando las suscripciones

## 2.6.5 EJEMPLOS

### **Petición SOAP:**

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_SubscriptionWS>
      <aClientId>test</aClientId>
      <aUser>SUPER</aUser>
      <aPassword>RADMIN</aPassword>
      <aDomain>PRODUCCION</aDomain>
      <aApplication>EDI5</aApplication>
      <aSchema>TEST MSG</aSchema>
      <aTransformation></aTransformation>
      <aReference></aReference>
      <aMessage></aMessage>
      <aMessageFormat>10</aMessageFormat>
    </ueb:TEBIBrokerInterface_SubscriptionWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### **Respuesta SOAP:**

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:TEBIBrokerInterface_SubscriptionWSResponse
xmlns:ns1="UEBIBrokerInterface">
      <return>1305061315552372323000</return>
      <aMessage/>
      <aMessageFormat>10</aMessageFormat>
    </ns1:TEBIBrokerInterface_SubscriptionWSResponse>
  </soap:Body>
</soap:Envelope>
```

Otra repuesta correcta con aMessageFormat igual a 1

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:TEBIBrokerInterface_SubscriptionWSResponse
xmlns:ns1="UEBIBrokerInterface">
      <return>1309121115086529616000</return>
```

```

    <aMessage>aGVsbG8gd29ybGQK</aMessage>
    <aMessageFormat>1</aMessageFormat>
  </ns1:TEBIBrokerInterface_SubscriptionWSResponse>
</soap:Body>
</soap:Envelope>

```

El siguiente ejemplo es una respuesta **errónea**.

```

<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  <soap:Body>
    <ns1:TEBIBrokerInterface_SubscriptionWSResponse
xmlns:ns1="UEBIBrokerInterface">
      <return>-12</return>
      <aMessage/>
      <aMessageFormat>10</aMessageFormat>
    </ns1:TEBIBrokerInterface_SubscriptionWSResponse>
  </soap:Body>
</soap:Envelope>

```

## 2.7 CONFIRMAR SUSCRIPCIONES

**TEBIBrokerInterface\_ConfirmSubscriptionWS**

Confirma de forma positiva la entrega de las suscripciones recogidas.

### 2.7.1 LLAMADA:

```

<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_ConfirmSubscriptionWS>
      <aClientId>?</aClientId>
      <aUser>?</aUser>
      <aPassword>?</aPassword>
      <aDomain>?</aDomain>
      <aApplication>?</aApplication>
      <aMessageID>?</aMessageID>
    </ueb:TEBIBrokerInterface_ConfirmSubscriptionWS>
  </soapenv:Body>
</soapenv:Envelope>

```

### 2.7.2 PARÁMETROS:

- **aClientId**: identificador del cliente

- **aUser:** usuario que realiza la confirmación
- **aPassword:** contraseña del usuario que realiza la confirmación
- **aDomain:** dominio donde se encuentra la suscripción a confirmar
- **aApplication:** aplicación donde se encuentra la suscripción a confirmar.
- **aMessageID:** identificador del mensaje que se quiere confirmar. Si se desean confirmar varios mensajes, se indicarán los identificadores de mensaje separados por '|'

### 2.7.3 RESPUESTA:

- **Respuesta satisfactoria:** Si la respuesta es satisfactoria, devolverá 1
- **Respuesta de error:** Si se produce algún error devolverá el código de error, indicando el motivo.

### 2.7.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la confirmación
-2	Suscripción no encontrada
-3	Estado del documento, no valido para la confirmación. Los documentos, solo pueden ser confirmados, cuando se encuentran en estado suscrito, pendiente de confirmación o bloqueado
-4	Documento ya en estado confirmado
-12	Error registrando la sesión

### 2.7.5 EJEMPLOS

#### Petición SOAP:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_ConfirmSubscriptionWS>
      <aClientId>CLIENTID</aClientId>
      <aUser>SUPER</aUser>
      <aPassword>RADMIN</aPassword>
      <aDomain>PRODUCCION</aDomain>
      <aApplication>EDI5</aApplication>
      <aMessageID>1309121115086529616000</aMessageID>
    </ueb:TEBIBrokerInterface_ConfirmSubscriptionWS>
  </soapenv:Body>
</soapenv:Envelope>
```

```
</soapenv:Envelope>
```

### **Respuesta SOAP:**

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  <soap:Body>
    <ns1:TEBIBrokerInterface_ConfirmSubscriptionWSResponse
xmlns:ns1="UEBIBrokerInterface">
      <return>1</return>
    </ns1:TEBIBrokerInterface_ConfirmSubscriptionWSResponse>
  </soap:Body>
</soap:Envelope>
```

El siguiente ejemplo es una respuesta **errónea**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  <soap:Body>
    <ns1:TEBIBrokerInterface_ConfirmSubscriptionWSResponse
xmlns:ns1="UEBIBrokerInterface">
      <return>-3</return>
    </ns1:TEBIBrokerInterface_ConfirmSubscriptionWSResponse>
  </soap:Body>
</soap:Envelope>
```

## **2.8 RECHAZAR SUSCRIPCIONES**

**TEBIBrokerInterface\_RejectSubscriptionWS**

Rechaza la suscripción recogida del sistema (es una confirmación negativa).

### **2.8.1 LLAMADA:**

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ueb="UEBIBrokerInterface">
  <soapenv:Header/>
  <soapenv:Body>
    <ueb:TEBIBrokerInterface_RejectSubscriptionWS>
      <aClientId?></aClientId>
      <aUser?></aUser>
      <aPassword?></aPassword>
      <aDomain?></aDomain>
      <aApplication?></aApplication>
      <aMessageID?></aMessageID>
      <aInfo?></aInfo>
```

```

    </ueb:TEBIBrokerInterface_RejectSubscriptionWS>
  </soapenv:Body>
</soapenv:Envelope>

```

## 2.8.2 PARÁMETROS:

- **aClientId:** identificador del cliente
- **aUser:** usuario que realiza el rechazo
- **aPassword:** contraseña del usuario que realiza el rechazo
- **aDomain:** dominio donde se encuentra la suscripción a rechazar
- **aApplication:** aplicación donde se encuentra la suscripción a rechazar.
- **aMessageID:** identificador del mensaje que se quiere rechazar. Si se desean rechazar varios mensajes, se indicarán los identificadores de mensaje separados por '|'.
- **aInfo:** Información adicional del motivo del rechazo.

## 2.8.3 RESPUESTA:

- **Respuesta satisfactoria:** Si la respuesta es satisfactoria, devolverá 1
- **Respuesta de error:** Si se produce algún error devolverá el código de error, indicando el motivo.

## 2.8.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante el proceso de rechazo
-2	Suscripción no encontrada
-3	Estado del documento, no valido para ser rechazado. Los documentos, solo pueden ser rechazados, cuando se encuentran en estado suscrito, pendiente de confirmación o bloqueado
-4	Documento ya en estado rechazado
-12	Error registrando la sesión

## 2.8.5 EJEMPLOS

### Petición SOAP:

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ueb="UEBIBrokerInterface">

```



```

<soapenv:Header/>
<soapenv:Body>
  <ueb:TEBIBrokerInterface_RejectSubscriptionWS>
    <aClientId>CLIENTID</aClientId>
    <aUser>SUPER</aUser>
    <aPassword>RADMIN</aPassword>
    <aDomain>PRODUCCION</aDomain>
    <aApplication>EDI5</aApplication>
    <aMessageID>1309121115086529616000</aMessageID>
    <aInfo>NO VALID</aInfo>
  </ueb:TEBIBrokerInterface_RejectSubscriptionWS>
</soapenv:Body>
</soapenv:Envelope>

```

### **Respuesta SOAP:**

El siguiente ejemplo es una respuesta **correcta**.

```

<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:TEBIBrokerInterface_RejectSubscriptionWSResponse
xmlns:ns1="UEBIBrokerInterface">
      <return>1</return>
    </ns1:TEBIBrokerInterface_RejectSubscriptionWSResponse>
  </soap:Body>
</soap:Envelope>

```

El siguiente ejemplo es una respuesta **errónea**.

```

<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:TEBIBrokerInterface_RejectSubscriptionWSResponse
xmlns:ns1="UEBIBrokerInterface">
      <return>-4</return>
    </ns1:TEBIBrokerInterface_RejectSubscriptionWSResponse>
  </soap:Body>
</soap:Envelope>

```

## 3 EBIBROKERWSV2

### 3.1 INTRODUCCIÓN

El webService **EBIBrokerWSV2**, permite el acceso al EBIBROKER, para poder realizar operaciones envío y recepción de mensajes.

Este webService desde un punto de vista funcional es similar al anterior, pero el API de acceso, es mas moderno, e incorpora estructuras de acceso mas avanzadas, y mas fáciles de emplear, desde los lenguajes de programación.

Este webServices, utiliza *SOAP Attachments*, y *MTOM (Message Transmission Optimization Mechanism)*, para el envío y recepción de mensajes.

#### 3.1.1 ACCESO AL SERVICIO

La URL de acceso al webservice EBIBrokerWSV2 del EBI-BROKER es la siguiente:

<https://aspXX.sedeb2b.com:9020/services/EBIBrokerWSV2?wsdl>

donde XX corresponde con el número de grupo asociado al cliente, para los grupos entre el 35 y el 40 inclusive, se debe utilizar aspebi.sedeb2b.com

#### 3.1.2 FUNCIONES DEL WEBSERVICE EBIBROKERWSV2

Función	Descripción	ver.
publishWS	Esta función permite publicar mensajes en el EBI-BROKER vía web service, utilizando SOAP Attachment, para enviar los fichero a publicar	
publishDBWS	Esta función permite publicar mensajes en el EBI BROKER, cuyo origen es una base de datos. Se publicarán las filas que cumplan el criterio de búsqueda de la petición, utilizando una representación en XML.	
transformAndPublishWS	Esta función publica mensajes en el EBIBROKER, aplicándoles previamente la transformación que se indica como parámetro. La función devolverá el resultado de la transformación	
confirmedPublicationsWS	Devuelve las publicaciones que han sido recuperadas por el destinatario	
notifiedPublicationWS	Emite un acuse de la publicación recuperada por el destinatario	
rejectedPublicationsWS	Devuelve las publicaciones que han sido rechazadas por el destinatario	
notifiedRejectedPublicationWS	Emite un acuse de la publicación rechazada	

subscriptionWS	Permite recuperar suscripciones, aplicándoles o no una transformación	
confirmSubscriptionWS	Confirma de forma positiva la entrega de los mensajes que se han recuperado	
rejectSubscriptionWS	Rechaza la suscripción recogida del sistema (es una confirmación negativa).	

Tipo complejos que devolverán alguna de las funciones webServices del servicio EBIBrokerWSV2

- **EBIDocumentWS**, representa un mensaje en el EBIBROKER

```
<xs:complexType name="ebiDocumentWS">
  <xs:sequence>
    <xs:element minOccurs="0" name="application" type="xs:string"/>
    <xs:element minOccurs="0" name="applicationdst" type="xs:string"/>
    <xs:element minOccurs="0" name="applicationsub" type="xs:string"/>
    <xs:element minOccurs="0" name="countsub" type="xs:int"/>
    <xs:element minOccurs="0" name="date" type="xs:dateTime"/>
    <xs:element minOccurs="0" name="domain" type="xs:string"/>
    <xs:element minOccurs="0" name="id" type="xs:string"/>
    <xs:element minOccurs="0" name="idsub" type="xs:string"/>
    <xs:element minOccurs="0" name="message" type="xs:string"/>
    <xs:element maxOccurs="unbounded" minOccurs="0" name="messages"
nillable="true" type="tns:ebiMessageServerWS"/>
    <xs:element minOccurs="0" name="reference" type="xs:string"/>
    <xs:element minOccurs="0" name="size" type="xs:int"/>
    <xs:element minOccurs="0" name="type" type="tns:ebiDocumentTypeWS"/>
  </xs:sequence>
</xs:complexType>
```

- **EBIMessageServerWS**, representa el contenido de un mensaje. Un mismo mensaje, puede tener varias repeticiones de esta estructura.

```
<xs:complexType name="ebiMessageServerWS">
  <xs:sequence>
    <xs:element minOccurs="0" name="message" type="xs:base64Binary"/>
    <xs:element minOccurs="0" name="name" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

- **EBIDocumentTypeWS**, contiene un enumerado, con los diferentes estados de un documento.

```
<xs:simpleType name="ebiDocumentTypeWS">
  <xs:restriction base="xs:string">
    <xs:enumeration value="PUBLISHED"/>
    <xs:enumeration value="SUBSCRIPTIONED"/>
  </xs:restriction>
</xs:simpleType>
```

```
<xs:enumeration value="CONFIRMED"/>
<xs:enumeration value="BLOCKED"/>
<xs:enumeration value="PUBLISHPARTLYCONFIRMED"/>
<xs:enumeration value="PUBLISHCONFIRMED"/>
<xs:enumeration value="PUBLISHNOTIFIED"/>
<xs:enumeration value="PENDINGCONFIRMATION"/>
<xs:enumeration value="INEBIMAPFORMAT"/>
<xs:enumeration value="REJECTED"/>
<xs:enumeration value="PUBLISHREJECTED"/>
<xs:enumeration value="PUBLISHREJECTEDNOTIFIED"/>
<xs:enumeration value="PUBLISHPARTLYREJECTED"/>
<xs:enumeration value="PUBLISHDUPLICATED"/>
<xs:enumeration value="DELETED"/>
<xs:enumeration value="DELETEDONLYDOC"/>
<xs:enumeration value="COPIED"/>
<xs:enumeration value="NOCOPIED"/>
<xs:enumeration value="UNKNOWN"/>
</xs:restriction>
</xs:simpleType>
```

## 4 FUNCIONES EBIBROKERWSV2

### 4.1 PUBLICACIÓN DE MENSAJES

---

#### **publishWS**

Esta función permitirá publicar mensajes en el EBIBROKER

#### 4.1.1 LLAMADA:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:publishWS>
      <clientId>?</clientId>
      <user>?</user>
      <password>?</password>
      <domain>?</domain>
      <application>?</application>
      <schema>?</schema>
      <destination>?</destination>
      <reference>?</reference>
      <messages>
        <message>cid:272360061242</message>
        <name>?</name>
      </messages>
      <duplicates>?</duplicates>
    </com:publishWS>
  </soapenv:Body>
</soapenv:Envelope>
```

#### 4.1.2 PARÁMETROS:

- **clientId:** identificador del cliente
- **user:** usuario que realiza la publicación
- **password:** contraseña del usuario que realiza la publicación
- **domain:** dominio donde se va a publicar el documento
- **application:** aplicación en la que se quiere publicar el documento
- **schema:** nombre del esquema con el que se publicará el documento
- **destination:** destino de la publicación del documento
- **reference:** referencia con la que se publicará el documento

- **messages:** Esta estructura contendrá los ficheros que representarán la publicación
  - **message:** contenido del fichero, bien el content-id del adjunto SOAP, bien el contenido codificado en base 64.
  - **name:** nombre del fichero
- **duplicates:** Indica que acción se debe tomar ante documentos duplicados
  - 0: dupIgnore. La publicación siempre se genera, incluso si ya existe una ya duplicada en el sistema.
  - 1: dupAccept. En caso de ya existir una publicación anterior similar (con el mismo HASH), no se genera la nueva publicación, ni sus suscripciones correspondientes.
  - 2: dupError. En caso de ya existir una publicación anterior devuelve un código de error
  - 3: dupOverride. En caso de ya existir una publicación, esta y sus suscripciones se borran, y se crea una nueva con el mensaje que se esta publicando.

Por publicación duplicada, se entiende, aquellos documentos con el mismo HASH, dentro del mismo dominio y aplicación.

### 4.1.3 RESPUESTA:

- **Respuesta satisfactoria:** Identificador de la publicación que se ha realizado en el EBIBROKER
- **Respuesta de error:** Si se produce un error, se generará un código de error, que indicará el tipo de error que se ha producido.

### 4.1.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la publicación
-2	Publicación no autorizada
-3	Aplicación destino no encontrada
-4	Documento incorrecto acorde a la validación sintáctica del esquema
-5	La aplicación destino no acepta la suscripción
-7	Usuario no autorizado a la suscripción
-8	Publicación duplicada
-9	El dominio no permite publicaciones BROADCAST
-10	El documento no se ha insertado correctamente en la base de datos

-11	Problema generando las suscripciones
-12	Error registrando la sesión
-16	Tamaño de la publicación excedido

## 4.1.5 EJEMPLOS

### **Petición SOAP:**

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:publishWS>
      <clientId>TEST</clientId>
      <user>SUPER</user>
      <password>RADMIN</password>
      <domain>PRODUCCION</domain>
      <application>EDI5</application>
      <schema>TEST MSG</schema>
      <destination>EDI5</destination>
      <reference></reference>
      <messages>
        <message>cid:743380743642</message>
        <name>TEST</name>
      </messages>
      <duplicates>0</duplicates>
    </com:publishWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### **Respuesta SOAP:**

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:publishWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
      <return>1309131004542208384</return>
    </ns2:publishWSResponse>
  </soap:Body>
</soap:Envelope>
```

El siguiente ejemplo es una respuesta **errónea**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:publishWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
```

```

        <return>-2</return>
    </ns2:publishWSResponse>
</soap:Body>
</soap:Envelope>

```

## 4.2 PUBLICACION DE MENSAJES CON ORIGEN BASE DE DATOS

### publishDBWS

Esta función permitirá publicar mensajes en el EBI BROKER, utilizando como origen de la publicación, una interfaz, que apunta a una base de datos, que debe ser **alcanzable**, por el EBI BROKER. Las filas que cumplan el criterio de búsqueda, se publicarán utilizando una representación en XML de las mismas. Si la publicación tiene éxito, se borrarán o actualizarán las filas publicadas, en función de los parámetros *delete* y *update* de la petición.

### 4.2.1 LLAMADA:

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:publishDBWS>
      <!--Optional:-->
      <clientId?></clientId>
      <!--Optional:-->
      <user?></user>
      <!--Optional:-->
      <password?></password>
      <!--Optional:-->
      <domain?></domain>
      <!--Optional:-->
      <application?></application>
      <!--Optional:-->
      <schema?></schema>
      <!--Optional:-->
      <destination?></destination>
      <!--Optional:-->
      <reference?></reference>
      <delete?></delete>
      <!--Optional:-->
      <update?></update>
      <!--Optional:-->
      <filter?></filter>
      <duplicates?></duplicates>
    </com:publishDBWS>
  </soapenv:Body>
</soapenv:Envelope>

```



## 4.2.2 PARÁMETROS:

- **clientId**: identificador del cliente
- **user**: usuario que realiza la publicación
- **password**: contraseña del usuario que realiza la publicación
- **domain**: dominio donde se va a publicar el documento
- **application**: aplicación en la que se quiere publicar el documento
- **schema**: nombre del esquema que contiene la estructura y los parámetros de conexión a la base de datos.
- **destination**: destino de la publicación del documento
- **reference**: referencia con la que se publicará el documento
- **delete**: Indica si se deben borrar las filas publicadas o no
- **update**: Indica que columna y con que valor, se deben actualizar las filas duplicadas.
- **filter**: Filtro que se aplicará para localizar las filas a publicar de la base de datos
- **duplicates**: Indica que acción se debe tomar ante documentos duplicados
  - 0: dupIgnore. La publicación siempre se genera, incluso si ya existe una ya duplicada en el sistema.
  - 1: dupAccept. En caso de ya existir una publicación anterior similar (con el mismo HASH), no se genera la nueva publicación, ni sus suscripciones correspondientes.
  - 2: dupError. En caso de ya existir una publicación anterior devuelve un código de error
  - 3: dupOverride. En caso de ya existir una publicación, esta y sus suscripciones se borran, y se crea una nueva con el mensaje que se está publicando.

Por publicación duplicada, se entiende, aquellos documentos con el mismo HASH, dentro del mismo dominio y aplicación.

## 4.2.3 RESPUESTA:

- **Respuesta satisfactoria**: Identificador de la publicación que se ha realizado en el EBIBROKER

- **Respuesta de error:** Si se produce un error, se generará un código de error, que indicará el tipo de error que se ha producido.

## 4.2.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la publicación
-2	Publicación no autorizada
-3	Aplicación destino no encontrada
-4	Documento incorrecto acorde a la validación sintáctica del esquema
-5	La aplicación destino no acepta la suscripción
-7	Usuario no autorizado a la suscripción
-8	Publicación duplicada
-9	El dominio no permite publicaciones BROADCAST
-10	El documento no se ha insertado correctamente en la base de datos
-11	Problema generando las suscripciones
-12	Error registrando la sesión
-16	Tamaño de la publicación excedido

## 4.2.5 EJEMPLOS

### Petición SOAP:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:publishDBWS>
      <clientId>TEST</clientId>
      <user>SUPERVISOR</user>
      <password>xxxxxx</password>
      <domain>PRODUCCION</domain>
      <application>TESTA</application>
      <schema>EBIUSERS</schema>
      <destination>TESTA</destination>
      <reference></reference>
      <delete>>false</delete>
      <update>CADUCA=0</update>
      <filter>ID=1</filter>
      <duplicates>0</duplicates>
    </com:publishDBWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### Respuesta SOAP:

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:publishWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
      <return>1409131004542208384</return>
    </ns2:publishWSResponse>
  </soap:Body>
</soap:Envelope>
```

El siguiente ejemplo es una respuesta **errónea**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:publishWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
      <return>-2</return>
    </ns2:publishWSResponse>
  </soap:Body>
</soap:Envelope>
```

## 4.3 TRANSFORMACION Y PUBLICACIÓN DE MENSAJES

---

### transformAnPublishWS

Esta función permitirá publicar mensajes en el EBIBROKER

### 4.3.1 LLAMADA:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:transformAndPublishWS>
      <clientId>?</clientId>
      <user>?</user>
      <password>?</password>
      <domain>?</domain>
      <application>?</application>
      <transformation>?</transformation>
      <!--Zero or more repetitions:-->
      <profile>
        <key>?</key>
        <value>?</value>
      </profile>
      <filter>?</filter>
      <schema>?</schema>
      <destination>?</destination>
```

```

<reference>?</reference>
<!--Zero or more repetitions:-->
<messages>
  <message>cid:1052811169699</message>
  <name>?</name>
</messages>
<duplicates>?</duplicates>
<mask>?</mask>
</com:transformAndPublishWS>
</soapenv:Body>
</soapenv:Envelope>

```

### 4.3.2 PARÁMETROS:

- **clientId:** identificador del cliente
- **user:** usuario que realiza la transformación y publicación
- **password:** contraseña del usuario que realiza la transformación y publicación
- **domain:** dominio donde se va a publicar el documento
- **application:** aplicación en la que se quiere publicar el documento
- **transformation:** transformación que se va aplicar a los documentos enviados, previo a su publicación.
- **profile:** en el perfil de ejecución, se puede pasar parámetros, para la ejecución de la transformación
  - **key:** clave del parámetro
  - **value:** valor del parámetro
- **schema:** nombre del esquema con el que se publicará el documento
- **destination:** destino de la publicación del documento
- **reference:** referencia con la que se publicará el documento
- **messages:** Esta estructura contendrá los ficheros sobre los que se va aplicar la transformación
  - **message:** contenido del fichero, bien el content-id del adjunto SOAP, bien el contenido codificado en base 64.
  - **name:** nombre del fichero
- **duplicates:** Indica que acción se debe tomar ante documentos duplicados

- 0: dupIgnore. La publicación siempre se genera, incluso si ya existe una ya duplicada en el sistema.
  - 1: dupAccept. En caso de ya existir una publicación anterior similar (con el mismo HASH), no se genera la nueva publicación, ni sus suscripciones correspondientes.
  - 2: dupError. En caso de ya existir una publicación anterior devuelve un código de error
  - 3: dupOverride. En caso de ya existir una publicación, esta y sus suscripciones se borran, y se crea una nueva con el mensaje que se esta publicando.
- **mask:** máscara de que se aplicará a los ficheros resultado de la transformación, y cuyo resultado serán los que se publiquen en el EBIBROKER.

Por publicación duplicada, se entiende, aquellos documentos con el mismo HASH, dentro del mismo dominio y aplicación.

### 4.3.3 RESPUESTA:

- **Respuesta satisfactoria:** Devolverá los ficheros resultado de la transformación que se va publicar. Estos ficheros de devolverán una lista contiene la estructura *EBIMessageServerWS*.
- **Respuesta de error:** Si se produce un error, se generará una excepción que contendrá el código de error.

### 4.3.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la publicación
-2	Publicación no autorizada
-3	Aplicación destino no encontrada
-4	Documento incorrecto acorde a la validación sintáctica del esquema
-5	La aplicación destino no acepta la suscripción
-7	Usuario no autorizado a la suscripción
-8	Publicación duplicada
-9	El dominio no permite publicaciones BROADCAST
-10	El documento no se ha insertado correctamente en la base de datos
-11	Problema generando las suscripciones
-12	Error registrando la sesión
-14	Error durante la transformación
-16	Tamaño de la publicación excedido

## 4.3.5 EJEMPLOS

### **Petición SOAP:**

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:transformAndPublishWS>
      <clientId>CLIENTID</clientId>
      <user>USER</user>
      <password>PASSWORD</password>
      <domain>PRODUCTION</domain>
      <application>APPSOURCE</application>
      <transformation>MAPA_RETORNO</transformation>
      <profile></profile>
      <filter></filter>
      <schema>XML_EDI</schema>
      <destination>APPDST</destination>
      <reference></reference>
      <!--Zero or more repetitions:-->
      <messages>
        <message>PD94bWwgdmVyc2lvbj0iMS4w...</message>
        <name>filename</name>
      </messages>
      <duplicates>0</duplicates>
      <mask>*</mask>
    </com:transformAndPublishWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### **Respuesta SOAP:**

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:transformAndPublishWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
      <return>
        <messages>
          <message>PD94bWwgdmVyc2lvbj0iMS4w...</message>
          <name>filename1</name>
        </messages>
        <messages>
          <message>PD94bWwgdmVyc2lvbj0iMS4w...</message>
          <name>filename2</name>
        </messages>
      </return>
    </ns2:transformAndPublishWSResponse>
  </soap:Body>
</soap:Envelope>
```

```

        </messages>
    </return>
</ns2:transformAndPublishWSWSResponse>
</soap:Body>
</soap:Envelope>

```

El siguiente ejemplo es una respuesta **errónea**.

```

<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:-999</faultcode>
      <faultstring>Usuario no valido</faultstring>
      <detail>
        <ns1:EBIBrokerServerException
xmlns:ns1="http://com.edicom.broker.adapter.services">
          <message
xmlns:ns2="http://com.edicom.broker.adapter.services">Usuario no
valido</message>
          <number xsi:type="xs:int"
xmlns:ns2="http://com.edicom.broker.adapter.services"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">-
12</number>
        </ns1:EBIBrokerServerException>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>

```

## 4.4 PUBLICACIONES CONFIRMADAS

### confirmedPublicationsWS

Recupera las publicaciones que han sido recuperadas por el destinatario

### 4.4.1 LLAMADA:

```

<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:confirmedPublicationsWS>
      <clientId?></clientId>
      <user?></user>
      <password?></password>
      <domain?></domain>
      <application?></application>
    </com:confirmedPublicationsWS>
  </soapenv:Body>
</soapenv:Envelope>

```

```
</soapenv:Body>
</soapenv:Envelope>
```

#### 4.4.2 PARÁMETROS:

- **clientId:** Identificador del cliente
- **user:** usuario que realiza la petición
- **password:** contraseña del usuario que realiza la petición
- **domain:** dominio en el cual el documento fue publicado
- **application:** aplicación en la que la cual el documento fue publicado

#### 4.4.3 RESPUESTA:

- **Respuesta satisfactoria:** Lista de objetos *EBIDocumentWS* con los mensajes que han sido satisfactoriamente recuperados.
- **Respuesta de error:** Si produce un error, se devolverá una excepción del tipo *EBIBrokerServerException* la cual indicará el código y el motivo del error.

#### 4.4.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la publicación
-12	Usuario no válido

#### 4.4.5 EJEMPLOS

##### **Petición SOAP:**

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:confirmedPublicationsWS>
      <clientId>TEST</clientId>
      <user>SUPER</user>
      <password>ADMIN</password>
      <domain>PRODUCCION</domain>
      <application>TESTJ</application>
    </com:confirmedPublicationsWS>
  </soapenv:Body>
</soapenv:Envelope>
```



**Respuesta SOAP:**

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:confirmedPublicationsWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
      <return>
        <application>TESTJ</application>
        <applicationdst>TESTB</applicationdst>
        <date>2010-07-09T11:11:07.927+02:00</date>
        <domain>PRODUCCION</domain>
        <id>1007090938587837629</id>
        <message>PDF</message>
        <reference/>
        <size>21503</size>
      </return>
    </ns2:confirmedPublicationsWSResponse>
  </soap:Body>
</soap:Envelope>
```

El siguiente ejemplo es una respuesta de **error**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:-999</faultcode>
      <faultstring>Usuario no valido</faultstring>
      <detail>
        <ns1:EBIBrokerServerException
xmlns:ns1="http://com.edicom.broker.adapter.services">
          <message
xmlns:ns2="http://com.edicom.broker.adapter.services">Usuario no
valido</message>
          <number xsi:type="xs:int"
xmlns:ns2="http://com.edicom.broker.adapter.services"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">-
12</number>
        </ns1:EBIBrokerServerException>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

## 4.5 NOTIFICAR PUBLICACIONES

---

`notifiedPublicationWS`

Emite un acuse de entrega de la confirmación de la publicación

### 4.5.1 LLAMADA:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:notifiedPublicationWS>
      <clientId>?</clientId>
      <user>?</user>
      <password>?</password>
      <domain>?</domain>
      <application>?</application>
      <!--Zero or more repetitions:-->
      <messageIds>?</messageIds>
    </com:notifiedPublicationWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### 4.5.2 PARÁMETROS:

- **clientId:** Identificador del cliente
- **user:** usuario que realiza la petición
- **password:** contraseña del usuario que realiza la petición
- **domain:** dominio donde se encuentra la publicación confirmada
- **application:** aplicación donde se encuentra la publicación confirmada
- **messageIds:** lista con los identificadores de la publicaciones confirmadas, a emitir su acuse de entrega

### 4.5.3 RESPUESTA:

- **Respuesta satisfactoria:** *True* si la operación se realiza con éxito
- **Respuesta de error:** *False* si se produce un error, o se devolverá una excepción del tipo `EBIBrokerServerException` la cual indicará el código y el motivo del error.

### 4.5.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la publicación
-12	Usuario no válido

## 4.5.5 EJEMPLOS

### Petición SOAP:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:notifiedPublicationWS>
      <clientId>CLIENT</clientId>
      <user>SUPER</user>
      <password>RADMIN</password>
      <domain>PRODUCCION</domain>
      <application>TESTJ</application>
      <!--Zero or more repetitions:-->
      <messageIds>1007090938587837629</messageIds>
      <messageIds>1007090938587837632</messageIds>
    </com:notifiedPublicationWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### Respuesta SOAP:

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:notifiedPublicationWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
      <return>true</return>
    </ns2:notifiedPublicationWSResponse>
  </soap:Body>
</soap:Envelope>
```

## 4.6 PUBLICACIONES RECHAZADAS

### rejectedPublicationsWS

Recupera las publicaciones que han sido rechazadas por el destinatario

### 4.6.1 LLAMADA:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
```

```

<soapenv:Header/>
<soapenv:Body>
  <com:rejectedPublicationsWS>
    <clientId?></clientId>
    <user?></user>
    <password?></password>
    <domain?></domain>
    <application?></application>
  </com:rejectedPublicationsWS>
</soapenv:Body>
</soapenv:Envelope>

```

## 4.6.2 PARÁMETROS:

- **clientId:** Identificador del cliente
- **user:** usuario que realiza la petición
- **password:** contraseña del usuario que realiza la petición
- **domain:** dominio en el cual el documento fue publicado
- **application:** aplicación en la que la cual el documento fue publicado

## 4.6.3 RESPUESTA:

- **Respuesta satisfactoria:** Lista de objetos *EBIDocumentWS* con los mensajes que han sido satisfactoriamente recuperados.
- **Respuesta de error:** Si produce un error, se devolverá una excepción del tipo *EBIBrokerServerException* la cual indicará el código y el motivo del error.

## 4.6.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la publicación
-12	Usuario no válido

## 4.6.5 EJEMPLOS

### Petición SOAP:

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>

```

```

    <com:rejectedPublicationsWS>
      <clientId>CLIENT</clientId>
      <user>SUPER</user>
      <password>RADMIN</password>
      <domain>PRODUCCION</domain>
      <application>TEST</application>
    </com:rejectedPublicationsWS>
  </soapenv:Body>
</soapenv:Envelope>

```

### **Respuesta SOAP:**

El siguiente ejemplo es una respuesta **correcta**.

```

<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:rejectedPublicationsWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
      <return>
        <application>TESTJ</application>
        <applicationdst>TESTB</applicationdst>
        <date>2013-09-13T15:35:32.217+02:00</date>
        <domain>PRODUCCION</domain>
        <id>1203231635593873126</id>
        <message>EDI_INSDES_D_96A_EN_EAN001_1</message>
        <reference>TEST-REF</reference>
        <size>27136</size>
      </return>
    </ns2:rejectedPublicationsWSResponse>
  </soap:Body>
</soap:Envelope>

```

El siguiente ejemplo es una respuesta **errónea**.

```

<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:-999</faultcode>
      <faultstring>Usuario no valido</faultstring>
      <detail>
        <ns1:EBIBrokerServerException
xmlns:ns1="http://com.edicom.broker.adapter.services">
          <message
xmlns:ns2="http://com.edicom.broker.adapter.services">Usuario no
valido</message>
            <number xsi:type="xs:int"
xmlns:ns2="http://com.edicom.broker.adapter.services"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">-
12</number>

```

```

        </ns1:EBIBrokerServerErrorException>
    </detail>
</soap:Fault>
</soap:Body>
</soap:Envelope>

```

## 4.7 NOTIFICAR PUBLICACIONES RECHAZADAS

**notifiedRejectedPublicationWS**

Emite un acuse de entrega del rechazo de la publicación

### 4.7.1 LLAMADA:

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:notifiedRejectedPublicationWS>
      <clientId>?</clientId>
      <user>?</user>
      <password>?</password>
      <domain>?</domain>
      <application>?</application>
      <!--Zero or more repetitions:-->
      <messageIds>?</messageIds>
    </com:notifiedRejectedPublicationWS>
  </soapenv:Body>
</soapenv:Envelope>

```

### 4.7.2 PARÁMETROS:

- **clientId:** Identificador del cliente
- **user:** usuario que realiza la petición
- **password:** contraseña del usuario que realiza la petición
- **domain:** dominio donde se encuentra la publicación rechazada
- **application:** aplicación donde se encuentra la publicación rechazada
- **messageIds:** lista con los identificadores de la publicaciones rechazadas, a emitir su acuse de entrega

### 4.7.3 RESPUESTA:

- **Respuesta satisfactoria:** *True* si la operación se realiza con éxito
- **Respuesta de error:** *False* si se produce un error, o se devolverá una excepción del tipo `EBIBrokerServerException` la cual indicará el código y el motivo del error.

## 4.7.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la publicación
-12	Usuario no válido

## 4.7.5 EJEMPLOS

### Petición SOAP:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:notifiedRejectedPublicationWS>
      <clientId>CLIENT</clientId>
      <user>SUPER</user>
      <password>RADMIN</password>
      <domain>PRODUCCION</domain>
      <application>TESTJ</application>
      <!--Zero or more repetitions:-->
      <messageIds>1203231635593873126</messageIds>
    </com:notifiedRejectedPublicationWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### Respuesta SOAP:

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:notifiedRejectedPublicationWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
      <return>true</return>
    </ns2:notifiedRejectedPublicationWSResponse>
  </soap:Body>
</soap:Envelope>
```

## 4.8 RECOGER SUSCRIPCIONES

`subscriptionWS`

Recoge los mensajes que estén en estado suscrito

### 4.8.1 LLAMADA:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:subscriptionWS>
      <clientId?></clientId>
      <user?></user>
      <password?></password>
      <domain?></domain>
      <application?></application>
      <schema?></schema>
      <transformation?></transformation>
      <reference?></reference>
    </com:subscriptionWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### 4.8.2 PARÁMETROS:

- **clientId**: identificador del cliente
- **user** : usuario que realiza la suscripción
- **password**: contraseña del usuario que realiza la suscripción
- **domain**: dominio donde se encuentra la suscripción a recoger
- **application**: aplicación donde se encuentra la suscripción
- **schema**: esquema del cual se quiere recoger la suscripción
- **transformation**: transformación que se desea aplicar a la suscripción, antes de recogerla. Este valor es opcional, sino se indica transformación se recogerá la suscripción tal cual se haya publicado.
- **reference**: referencia del mensaje que se quiere recuperar, sino se indica ninguna, se recogerá la primera suscripción que este disponible.

### 4.8.3 RESPUESTA:

- **Respuesta satisfactoria**: Si la respuesta es satisfactoria, se devolverá, el objeto *EBIDocumentWS*, que contendrá la suscripción recuperada del EBIBROKER. Este objeto a su vez, contendrá un lista



de *EBIMessageServerWS*, con el contenido de los ficheros que había en la suscripción.

- **Respuesta de error:** Si se produce algún error devolverá el código de error, indicando el motivo.

#### 4.8.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la suscripción
-2	Publicación no autorizada a recoger la suscripción
-3	Transformación no encontrada
-4	Error realizando la transformación
-7	Usuario no autorizado a la suscripción
-12	Problema generando las suscripciones

#### 4.8.5 EJEMPLOS

##### **Petición SOAP:**

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:subscriptionWS>
      <clientId>CLIENT</clientId>
      <user>SUPER</user>
      <password>RADMIN</password>
      <domain>PRODUCCION</domain>
      <application>TESTA</application>
      <schema>TEST</schema>
      <transformation></transformation>
      <reference></reference>
    </com:subscriptionWS>
  </soapenv:Body>
</soapenv:Envelope>
```

##### **Respuesta SOAP:**

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:subscriptionWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
      <return>
        <application>TESTA</application>
        <applicationsub>TESTA_EDIWIN</applicationsub>
```

```

        <countsub>1</countsub>
        <date>2013-09-13T15:49:06.555+02:00</date>
        <domain>PRODUCCION</domain>
        <id>1212111838583307382000</id>
        <idsub>1212111838583307382</idsub>
        <message>TEST</message>
        <messages>
            <message>
                <xop:Include href="cid:64068d22-fafe-4dda-
a5f2-5b1e14e52d5a-1@cxf.apache.org"
xmlns:xop="http://www.w3.org/2004/08/xop/include"/>
            </message>
            <name>INVRPTD96.EDI</name>
        </messages>
        <reference>I121211192157924</reference>
        <size>12569907</size>
    </return>
</ns2:subscriptionWSResponse>
</soap:Body>
</soap:Envelope>

```

El siguiente ejemplo es una respuesta **errónea**.

```

<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:-999</faultcode>
      <faultstring>Error Suscripción del esquema TESTMSG, no
autorizada en la aplicación TESTJ</faultstring>
      <detail>
        <ns1:EBIBrokerServerException
xmlns:ns1="http://com.edicom.broker.adapter.services">
          <message
xmlns:ns2="http://com.edicom.broker.adapter.services">Error
Suscripción del esquema TESTMSG, no autorizada en la aplicación
TESTJ</message>
          <number xsi:type="xs:int"
xmlns:ns2="http://com.edicom.broker.adapter.services"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">-
2</number>
        </ns1:EBIBrokerServerException>
      </detail>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>

```

## 4.9 CONFIRMAR SUSCRIPCIONES

**confirmSubscriptionWS**

Confirma de forma positiva la entrega de las suscripciones recogidas.

### 4.9.1 LLAMADA:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:confirmSubscriptionWS>
      <clientId>?</clientId>
      <user>?</user>
      <password>?</password>
      <domain>?</domain>
      <application>?</application>
      <!--Zero or more repetitions:-->
      <messageIds>?</messageIds>
    </com:confirmSubscriptionWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### 4.9.2 PARÁMETROS:

- **clientId:** identificador del cliente.
- **user:** usuario que realiza la confirmación.
- **password:** contraseña del usuario que realiza la confirmación.
- **domain:** dominio donde se encuentra la suscripción a confirmar.
- **application:** aplicación donde se encuentra la suscripción a confirmar.
- **messageID:** identificador del mensaje que se quiere confirmar. Si se desean confirmar varios mensajes, se repetirá esta sección con nuevos identificadores.

### 4.9.3 RESPUESTA:

- **Respuesta satisfactoria:** Si la respuesta es satisfactoria, devolverá 1
- **Respuesta de error:** Si se produce algún error devolverá el código de error, indicando el motivo.

### 4.9.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante la confirmación
-2	Suscripción no encontrada
-3	Estado del documento, no valido para la confirmación. Los documentos, solo pueden ser confirmados, cuando se encuentran en estado suscrito, pendiente de confirmación o bloqueado
-4	Documento ya en estado confirmado
-12	Error registrando la sesión

## 4.9.5 EJEMPLOS

### Petición SOAP:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:confirmSubscriptionWS>
      <clientId>CLIENT</clientId>
      <user>ASPZZ\SUPERVISOR</user>
      <password>SUPERVISOR</password>
      <domain>PRODUCCION</domain>
      <application>TESTA</application>
      <!--Zero or more repetitions:-->
      <messageIds>1304171320100623838000</messageIds>
    </com:confirmSubscriptionWS>
  </soapenv:Body>
</soapenv:Envelope>
```

### Respuesta SOAP:

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:confirmSubscriptionWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
      <return>1</return>
    </ns2:confirmSubscriptionWSResponse>
  </soap:Body>
</soap:Envelope>
```

El siguiente ejemplo es una respuesta **erronea**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:confirmSubscriptionWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
```

```

        <return>-4</return>
    </ns2:confirmSubscriptionWSResponse>
</soap:Body>
</soap:Envelope>

```

## 4.10 RECHAZAR SUSCRIPCIONES

### rejectSubscriptionWS

Rechaza la suscripción recogida del sistema (es una confirmación negativa).

### 4.10.1 LLAMADA:

```

<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:rejectSubscriptionWS>
      <clientId>?</clientId>
      <user>?</user>
      <password>?</password>
      <domain>?</domain>
      <application>?</application>
      <!--Zero or more repetitions:-->
      <messageIds>?</messageIds>
      <info>?</info>
    </com:rejectSubscriptionWS>
  </soapenv:Body>
</soapenv:Envelope>

```

### 4.10.2 PARÁMETROS:

- **clientId:** identificador del cliente
- **user:** usuario que realiza el rechazo
- **password:** contraseña del usuario que realiza el rechazo
- **domain:** dominio donde se encuentra la suscripción a rechazar
- **application:** aplicación donde se encuentra la suscripción a rechazar.
- **messageID:** identificador del mensaje que se quiere rechazar. Si se desean rechazar varios mensajes, se repetirá esta sección con nuevos identificadores.

- **info:** Información adicional del motivo del rechazo.

### 4.10.3 RESPUESTA:

- **Respuesta satisfactoria:** Si la respuesta es satisfactoria, devolverá 1
- **Respuesta de error:** Si se produce algún error devolverá el código de error, indicando el motivo.

### 4.10.4 CÓDIGOS DE ERROR:

Código	Error
-1	Error general durante el proceso de rechazo
-2	Suscripción no encontrada
-3	Estado del documento, no valido para ser rechazado. Los documentos, solo pueden ser rechazados, cuando se encuentran en estado suscrito, pendiente de confirmación o bloqueado
-4	Documento ya en estado rechazado
-12	Error registrando la sesión

### 4.10.5 EJEMPLOS

#### Petición SOAP:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:com="http://com.edicom.broker.adapter.services">
  <soapenv:Header/>
  <soapenv:Body>
    <com:rejectSubscriptionWS>
      <clientId>CLIENTID</clientId>
      <user>SUPER</user>
      <password>RADMIN</password>
      <domain>PRODUCCION</domain>
      <application>TESTA</application>
      <!--Zero or more repetitions:-->
      <messageIds>1304171402443772807000</messageIds>
      <info>NO VALID</info>
    </com:rejectSubscriptionWS>
  </soapenv:Body>
</soapenv:Envelope>
```

#### Respuesta SOAP:

El siguiente ejemplo es una respuesta **correcta**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
```

```
<ns2:rejectSubscriptionWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
  <return>1</return>
</ns2:rejectSubscriptionWSResponse>
</soap:Body>
</soap:Envelope>
```

El siguiente ejemplo es una respuesta **errónea**.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns2:rejectSubscriptionWSResponse
xmlns:ns2="http://com.edicom.broker.adapter.services">
      <return>-4</return>
    </ns2:rejectSubscriptionWSResponse>
  </soap:Body>
</soap:Envelope>
```

## 5 ANEXO. ACCESO A EDICOMNET

Para acceder a la VAN EdicomNet, que esta soportada por un EBI, los parámetros que se deben utilizar son los siguientes:

- La URL de acceso es <https://adapter.sedeb2b.com:9020> y posteriormente dependiendo de la interfaz de acceso que se utilice será
  - EBIBROKERWS.  
<https://adapters.sedeb2b.com:9020/ebiBroker/services/EBIBrokerWS>
  - EBIBROKERWSV2.  
<https://adapters.sedeb2b.com:9020/ebiBroker/services/EBIBrokerWSV2>
- Como usuario se debe indicar el nombre del buzón. Si este es de la red de SEDEB2B, tendrá la forma [XXXXX@SEDEB2B.COM](mailto:XXXXX@SEDEB2B.COM)
- La contraseña de acceso es la misma que para el buzón.
- Como dominio existen dos posibles valores:
  - SEDEB2B.COM
  - AAPP.EDICOMNET.COM
- Como aplicación se debe indicar el nombre del buzón. Si este es de la red de SEDEB2B, tendrá la forma [XXXXX@SEDEB2B.COM](mailto:XXXXX@SEDEB2B.COM)
- El esquema que se emplea en la red de EdicomNet siempre es SM.
- El resto de parámetros ya son similares a cualquier otra petición genérica.



## 6 ANEXO: WSDL EBIBROKER

### 6.1 EBIBROKERWS

---

La última versión de WSDL esta disponible en :

<https://aspebi.sedeb2b.com:9020/services/EBIBrokerWS?wsdl>

### 6.2 EBIBROKERWSV2

---

La última versión de WSDL esta disponible en :

<https://aspebi.sedeb2b.com:9020/services/EBIBrokerWSV2?wsdl>