



G VICEPRESIDÈNCIA  
O I CONSELLERIA  
I INNOVACIÓ  
B RECERCA I TURISME  
/ DIRECCIÓ GENERAL  
DESENVOLUPAMENT  
TECNOLÒGIC

una manera de hacer  
**europa** 

Fondo Europeo de  
Desarrollo Regional



Unió Europea

# PINBAL: Manual d'integració

27 de novembre de 2018

**Serveis d'Administració Electrònica en el Govern de les Illes Balears**

Lot 3 (Serveis de tramitació d'expedients electrònics)

**Oficina Tècnica de Direcció de Projecte**

## Control de versions del document

Control de Canvis			
Data	Autor	Versió	Canvis
15/10/18	Límit Technologies	1.0	Versió inicial
27/11/18	Límit Technologies	1.1	Modificació del document segons plantilla enviada per la DGDT.

Revisat per		
Nom	Data	Àrea, departament o empresa

Aprobat per		
Nom	Data	Àrea, departament o empresa

Llista de distribució		
Nom	Àrea, departament o empresa	Correu electrònic

## Índex

<b>Control de versions del document.....</b>	<b>2</b>
<b>1. Objecte.....</b>	<b>4</b>
<b>2. Diagrama de funcionament.....</b>	<b>4</b>
2.1. peticionSincrona.....	4
2.2. peticionAsincrona.....	4
2.3. getRespuesta.....	6
2.4. getJustificante.....	6
<b>3. Model de classes.....</b>	<b>6</b>
<b>4. Descripció del servei.....</b>	<b>7</b>
4.1. peticionSincrona.....	8
4.2. peticionAsincrona.....	8
4.3. getRespuesta.....	9
4.4. getJustificante.....	10
4.5. Gestió d'errors.....	10
<b>5. Consideracions de disseny i ús de l'API.....</b>	<b>10</b>
5.1. Adreça del servei.....	10
5.2. Segurització del servei.....	10
5.3. Dependències Maven.....	11
<b>6. Client d'exemple.....</b>	<b>11</b>
<b>Annex I: Estructures de dades SCSP.....</b>	<b>14</b>
6.1. Peticion.....	14
6.2. Respuesta.....	15
6.3. ConfirmacionPeticion.....	16
6.4. Detall dels elements.....	16
<b>Annex II: Possibles estats de les peticions.....</b>	<b>19</b>
<b>Annex III: WSDL del servei de recobriment.....</b>	<b>20</b>

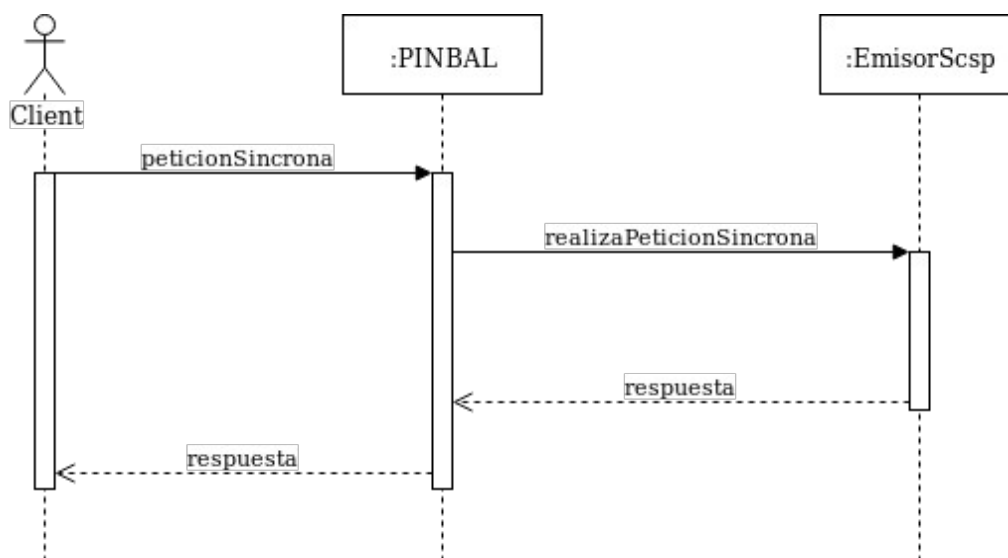
## 1. Objecte

En aquest document es proporciona tota la informació necessària per a la integració d'una aplicació amb el servei de recobriment de PINBAL.

## 2. Diagrama de funcionament

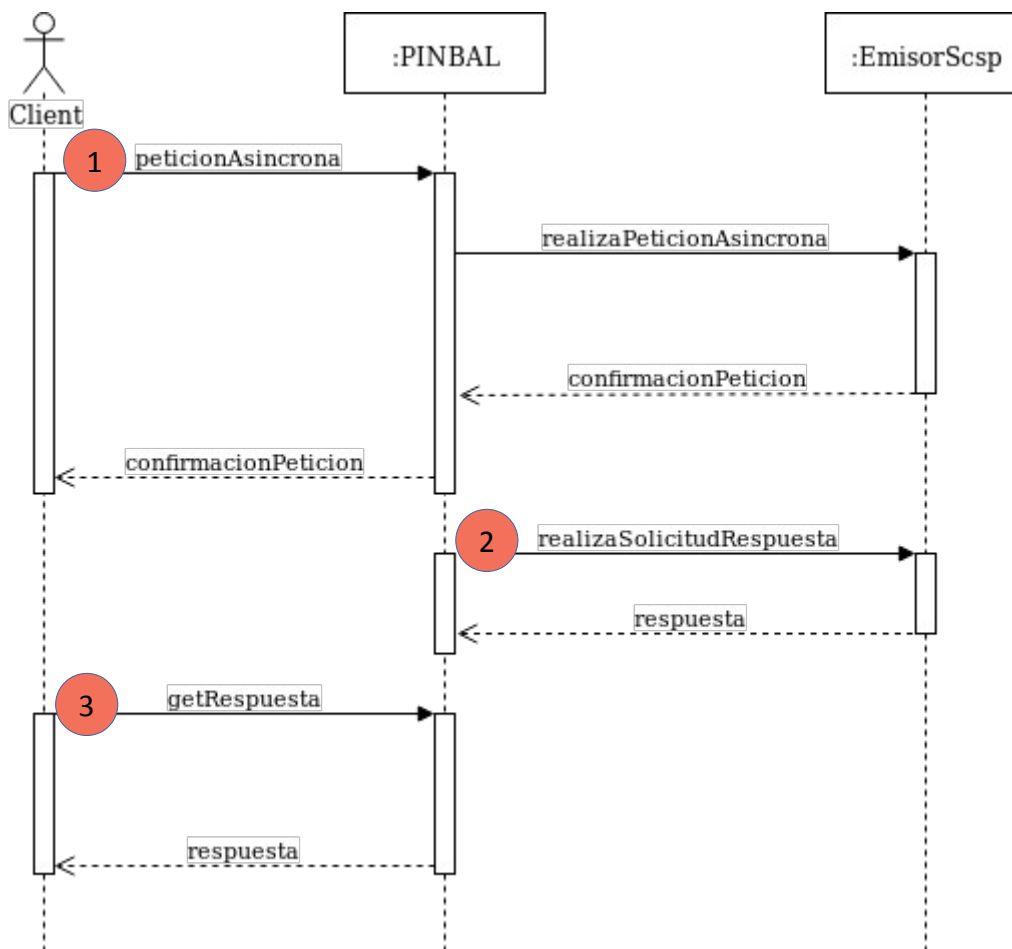
### 2.1. peticionSincrona

Aquest cas d'ús permet fer la consulta a un servei SCSP de forma síncrona. Això vol dir que el client, una vegada feta la petició, es quedarà esperant fins que el servidor retorni la resposta.



### 2.2. peticionAsincrona

Aquest cas d'ús permet fer la consulta a un servei SCSP de forma asíncrona. Això vol dir que el servidor emmagatzemarà la petició i retornarà immediatament un missatge al client informant de que la petició s'ha iniciat correctament però aquest missatge no contendrà la resposta a la petició SCSP.



La seqüència d'accions per a les peticions asíncrones és la següent:

1.- El client envia la petició invocant el mètode `peticionAsincrona` del servei de recobriment. Aquest mètode, en lloc de retornar un objecte de tipus `Respuesta`, retorna un objecte de tipus `ConfirmacionPeticion` amb la següent informació:

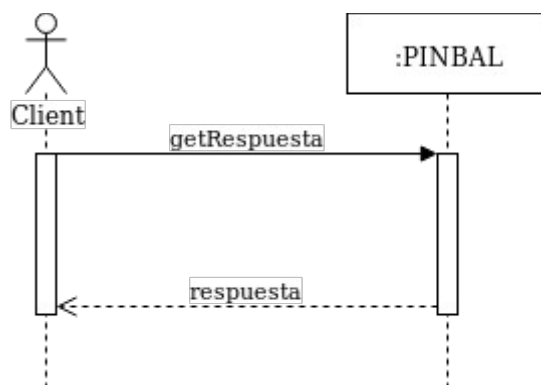
- `idPeticion`: L'identificador assignat a la petició SCSP.
- `estado`: L'estat de la resposta (típicament «0002» que indica que la petició s'ha iniciat correctament i que està pendent d'obtenir la resposta).

2.- De forma periòdica, PINBAL anirà fent peticions a l'emissor SCSP. Si la resposta ja s'ha acabat de processar, l'emissor retornarà un missatge amb l'estat «0003» i la resposta SCSP. Si la resposta encara està pendent de processar, l'emissor retornarà un missatge SCSP amb l'estat «0002». PINBAL continuarà fent peticions a l'emissor SCSP fins a obtenir la resposta.

3.- El client del servei de recobriment pot anar fent peticions a PINBAL per a obtenir informació sobre l'estat de la resposta mitjançant el mètode `getRespuesta`. Si la resposta està disponible es retornarà un missatge amb l'estat «0003» que contendrà la informació retornada pel servei SCSP. Si la resposta encara no està disponible es retornarà un missatge amb l'estat «0002».

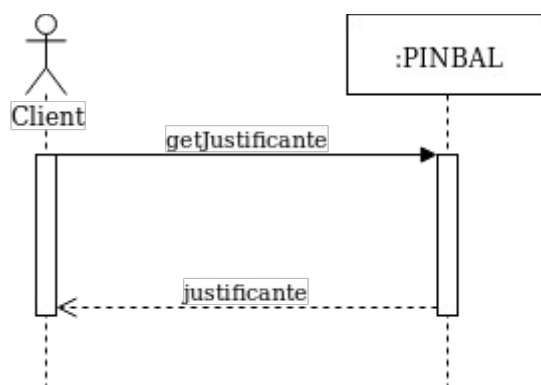
## 2.3. getRespuesta

Aquest cas d'ús permet obtenir la resposta SCSP d'una petició feta amb anterioritat mitjançant el servei de recobriment. Aquest mètode es pot utilitzar indistintament per a obtenir la resposta de peticions síncrones com de peticions asíncrones.



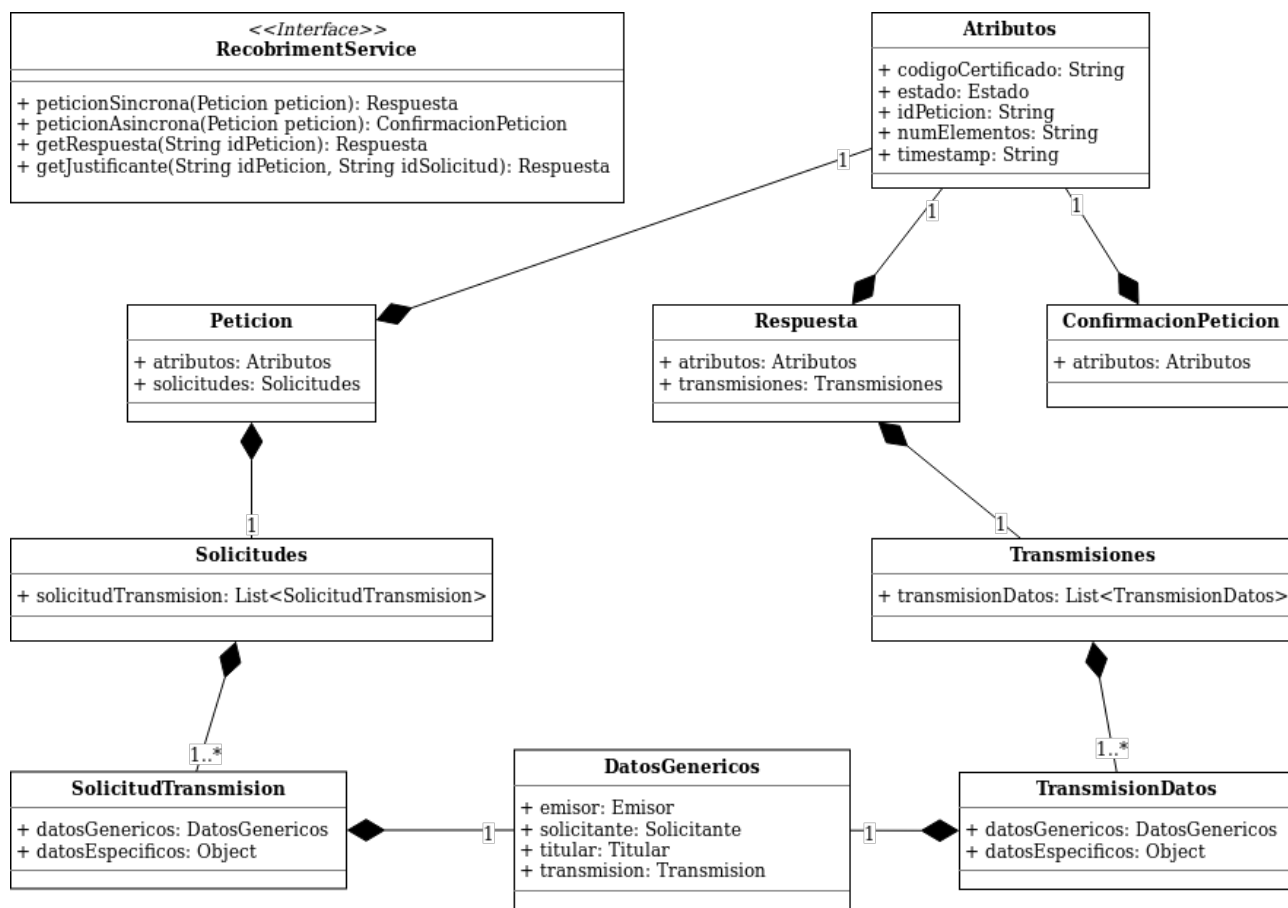
## 2.4. getJustificante

Aquest cas d'ús permet obtenir el justificant d'una sol·licitud feta amb anterioritat mitjançant el servei de recobriment. El justificant només es podrà retornar si la resposta SCSP està en estat de processada (codi d'estat «0003»).



## 3. Model de classes

Aquest és el model de classes del servei de recobriment.



Per a més detall sobre el significat de cada un dels camps i per a veure amb més detall la informació de l'objecte **DatosGenericos** podeu consultar l'Annex I.

## 4. Descripció del servei

El servei de recobriment de PINBAL permet l'enviament de peticions SCSP, la consulta de l'estat d'una petició enviada amb anterioritat i la generació del justificant associat a una sol·licitud.

Els mètodes que formen part d'aquest servei són els següents:

1. **peticionSincrona**: per a realitzar consultes síncrones.
2. **peticionAsincrona**: per a realitzar consultes asíncrones.
3. **getRespuesta**: per a obtenir els detalls d'una consulta feta amb anterioritat.
4. **getJustificante**: per a descarregar el justificant associat a una sol·licitud.

A continuació es descriuen amb més detall cada un d'aquests mètodes.

## 4.1. **peticionSincrona**

Aquesta petició permet l'enviament d'una petició SCSP de forma síncrona. Això vol dir que el servei esperarà fins que la petició s'hagi processat completament abans de retornar la resposta.

Aquest mètode requereix com a paràmetre un objecte de tipus **Peticion** i retorna un objecte de tipus **Respuesta**.

L'objecte de tipus Peticion ha de complir les següents restriccions:

1. Ha d'estar informat l'objecte **Atributos** amb els camps *codigoCertificado* i *numElementos* emplenats.
2. En les peticions síncrones només es pot enviar una sol·licitud a dins cada petició i el valor del camp *numElementos* ha de ser 1.
3. Els altres camps de l'objecte **Atributos** s'ignoraran i es poden deixar en blanc.
4. El camp *solicitudes* ha de contenir una única **SolicitudTransmision** i aquesta s'ha d'emplenar.

Per a més detalls sobre com emplenar la informació de la sol·licitud es pot consultar l'Annex I i la documentació del servei SCSP que es vol invocar. L'estructura del camp **DatosEspecificos** i els camps obligatoris i opcionals poden variar d'un servei a un altre.

La informació resultant serà un objecte de tipus **Respuesta** que contendrà la informació retornada per l'emissor SCSP. Aquesta resposta contendrà un objecte de tipus **Atributos** amb els camps *idPeticion* i *estado* emplenats.

El camp *estado* contendrà l'estat resultant de la petició, que normalment serà Tramitada (codi «0003»).

El camp *idPeticion* conté l'identificador assignat per PINBAL a la petició. Aquest camp es podrà utilitzar per a obtenir de nou la informació de la resposta (mitjançant el mètode **getRespuesta**) o per a generar el justificant de la petició (mitjançant el mètode **getJustificante**).

## 4.2. **peticionAsincrona**

Aquesta petició permet l'enviament d'una petició SCSP de forma asíncrona. Això vol dir que aquest mètode retornarà abans que la resposta a la petició estigui disponible. Per a obtenir la resposta s'haurà d'anar consultant l'estat de la petició mitjançant el mètode **getRespuesta** d'aquest mateix servei.

Aquest mètode requereix com a paràmetre un objecte de tipus **Peticion** i retorna un objecte de tipus **ConfirmacionPeticion**.

L'objecte de tipus Peticion ha de complir les següents restriccions:



1. Ha d'estar informat l'objecte **Atributos** amb els camps *codigoCertificado* i *numElementos* emplenats.
2. En les peticions asíncrones es poden enviar múltiples sol·licituds a dins cada petició i el valor del camp *numElementos* ha de ser major que 1.
3. Els altres camps de l'objecte **Atributos** s'ignoraran i es poden deixar en blanc.
4. El camp *solicitudes* ha de contenir, com a mínim, una **SolicitudTransmision**.

La informació resultant serà un objecte de tipus **ConfirmacionPetición**. Aquesta resposta contendrà un objecte de tipus **Atributos** amb els camps *idPetición* i *estado* emplenats.

El camp *estado* contendrà l'estat resultant de la petició, que normalment serà Tramitada (codi «0003»).

El camp *idPetición* conté l'identificador assignat per PINBAL a la petició. Aquest camp es podrà utilitzar per a obtenir de nou la informació de la resposta (mitjançant el mètode **getRespuesta**) o per a generar el justificant de la petició (mitjançant el mètode **getJustificante**).

Per a obtenir la resposta en les peticions asíncrones s'ha d'anar consultant l'estat de la petició mitjançant el mètode **getRespuesta**. Si l'estat retornat és «0003» això voldrà dir que la petició s'ha processat correctament i la mateixa resposta contendrà la informació retornada per l'emissor SCSP. Si l'estat retornat és diferent de «0003» voldrà dir que la petició encara no ha estat processada.

#### 4.3. **getRespuesta**

Aquest mètode permet consultar l'estat d'una petició i obtenir la resposta si aquesta es troba disponible.

Aquest mètode requereix com a paràmetre un *String* amb l'identificador de la petició que es vol consultar (camp *idPetición* de l'objecte **Atributos**) i retorna un objecte de tipus **Respuesta**.

Aquest objecte de tipus **Respuesta** es retornarà sempre amb el camp *Atributos* emplenat. El camp *Transmisiones* només s'emplenarà si la petició està en estat Tramitada (per a més informació sobre els estats es pot consultar l'Annex I).

#### 4.4. **getJustificante**

Aquest mètode retorna el contingut del justificant d'una sol·licitud en forma de document PDF firmat.

Aquest mètode accepta dos paràmetres (*idPetición* i *idSolicitud*) i retorna un array de bytes que conté un document PDF amb el justificant.

Els justificants es generen a partir d'una de les sol·licituds d'una petició i, per tant, és necessari especificar dos paràmetres: *idPetición* i *idSolicitud*. També es requereix que la petició es trobi en

estat tramitada (codi «0003»). Si la petició no es troba en aquest estat aquest mètode retornarà un error.

En el cas particular de que la petició només contengui una única sol·licitud, només serà necessari emplenar el paràmetre *idPetición* per a generar el justificant (el paràmetre *idSolicitud* es pot deixar amb el valor null).

## 4.5. Gestió d'errors

Si durant el processament d'una petició es produeix algun error de validació o intern de l'aplicació, el servei de recobriment retornarà un missatge de tipus SoapFault que es traduirà en una excepció en el client. En aquest cas, la descripció de l'error contendrà la informació necessària per a identificar la causa de l'error.

# 5. Consideracions de disseny i ús de l'API

## 5.1. Adreça del servei

L'aplicació desplega aquest servei en la següent adreça:

<http://localhost:8080/pinbal/ws/recobriment>

Aquesta adreça és vàlida per a accedir a un servei desplegat a la màquina local. Per a accedir al servei publicat en una màquina remota s'haurà de substituir «localhost» pel nom del servidor remot i el port «8080» pel port del servidor remot.

## 5.2. Segurització del servei

Per a accedir al servei és necessari una autenticació de tipus BASIC amb usuari/contrasenya.

També es requereix que l'usuari utilitzat per a accedir al servei tengui assignat el rol PBL\_WS.

## 5.3. Dependències Maven

Per a desenvolupar un client per al servei de recobriment el més senzill és crear un nou projecte amb Maven i incloent els següents blocs a dins el fitxer pom.xml:

A dins la secció <dependencies>:

```

<dependency>
  <groupId>es.caib.pinbal</groupId>
  <artifactId>pinbal-ws-client</artifactId>
  <version>1.4</version>
</dependency>
  
```

A dins la secció <repositories>:

```

<repository>
  <id>github-governib-maven-repo</id>
  <name>GitHub GovernIB Maven Repository</name>
  <url>http://GovernIB.github.io/maven/maven/</url>
</repository>

```

## 6. Client d'exemple

Aquest és el codi font d'una classe de test que empra el client del recobriment per a realitzar una petició al servei de consulta de dades de residència («VDRSFWS02»):

```

public class RecobrimentClientTest {

    private static final String ENDPOINT_URL =
"http://localhost:8080/pinbal/ws/recobriment";
    private static final String USERNAME = "pinbal";
    private static final String PASSWORD = "pinbal";

    private static final String CODIGO_CERTIFICADO = "VDRSFWS02";
    private static final Consentimiento CONSENTIMIENTO = Consentimiento.LEY;
    private static final String FINALIDAD = "Test recobriment";
    private static final String FUNCIONARIO_NIF = "97669911C";
    private static final String FUNCIONARIO_NOMBRE = "MPR TESTER";
    private static final String SOLICITANTE_ID = "G07896004";
    private static final String SOLICITANTE_NOMBRE = "Fundació BIT";
    private static final String PROCEDIMIENTO_CODIGO =
"IBIT_20101223_PRUEBA";
    private static final String PROCEDIMIENTO_NOMBRE = "PROCEDIMIENTO DE
PRUEBA FUNDACION IBIT";
    private static final String UNIDAD_CODIGO = null;
    private static final String UNIDAD_NOMBRE = "Unitat de test";
    private static final String EXPEDIENTE_ID = null;

    public static void main(String[] args) throws MalformedURLException,
ParserConfigurationException {
        Recobriment client = RecobrimentClientFactory.getWsClient(
            ENDPOINT_URL,
            USERNAME,
            PASSWORD);

        Peticion peticion = new Peticion();
        Atributos atributos = new Atributos();
        peticion.setAtributos(atributos);
        atributos.setNumElementos("1");
        atributos.setCodigoCertificado(CODIGO_CERTIFICADO);
        Solicitudes solicitudes = new Solicitudes();
        SolicitudTransmision solicitudTransmision = new
SolicitudTransmision();
        DatosGenericos datosGenericos = new DatosGenericos();
        Solicitante solicitante = new Solicitante();
        solicitante.setConsentimiento(CONSENTIMIENTO);
        solicitante.setFinalidad(FINALIDAD);
        Funcionario funcionario = new Funcionario();
        funcionario.setNifFuncionario(FUNCIONARIO_NIF);
    }
}

```

```

funcionario.setNombreCompletoFuncionario(FUNCIONARIO_NOMBRE);
solicitante.setFuncionario(funcionario);
solicitante.setIdentificadorSolicitante(SOLICITANTE_ID);
solicitante.setNombreSolicitante(SOLICITANTE_NOMBRE);
Procedimiento procedimiento = new Procedimiento();
procedimiento.setCodProcedimiento(PROCEDIMIENTO_CODIGO);
procedimiento.setNombreProcedimiento(PROCEDIMIENTO_NOMBRE);
solicitante.setProcedimiento(procedimiento);
solicitante.setCodigoUnidadTramitadora(UNIDAD_CODIGO);
solicitante.setUnidadTramitadora(UNIDAD_NOMBRE);
solicitante.setIdExpediente(EXPEDIENTE_ID);
datosGenericos.setSolicitante(solicitante);
Titular titular = new Titular();
datosGenericos.setTitular(titular);
solicitudTransmision.setDatosGenericos(datosGenericos);

solicitudTransmision.setDatosEspecificos(generarDatosEspecificos());
solicitudes.getSolicitudTransmision().add(solicitudTransmision);
peticion.setSolicitudes(solicitudes);
Respuesta respuesta = client.peticionSincrona(peticion);
System.out.println("Petició SCSP enviada (idPeticion=" +
respuesta.getAtributos().getIdPeticion() + ")");
}

private static Element generarDatosEspecificos() throws
ParserConfigurationException {
    DocumentBuilderFactory fac = DocumentBuilderFactory.newInstance();
    fac.setNamespaceAware(true);
    Document doc = fac.newDocumentBuilder().newDocument();
    Element datosEspecificos = doc.createElement("DatosEspecificos");
    Element consulta = doc.createElement("Consulta");
    Element provincia = doc.createElement("CodigoProvincia");
    provincia.setTextContent("07");
    consulta.appendChild(provincia);
    datosEspecificos.appendChild(consulta);
    doc.appendChild(datosEspecificos);
    return doc.getDocumentElement();
}
}

```

## Annex I: Estructures de dades SCSP

### 6.1. Peticion

Aquest missatge s'utilitza per a l'enviament de les peticions SCSP.

Peticion	Atributos	idPeticion			
		numElementos			
		codigoCertificado			
		estado			
		timestamp			
	Solicitudes	SolicitudTransm.	DatosGenericos	Emisor	nifEmisor
				nombreEmisor	
				Solicitante	Consentim.
					Finalidad
					Funcionario
					IdentificadorSol.
					NombreSol.
					idExpediente
					Procedimiento
					UnidadTramit.
		Titular	Nombre		
			Apellido1		
			Apellido2		
			NombreComp.		
			Documentacion		
			TipoDocument.		
		Transmis.	CodigoTrans.		
			FechaGen.		
			idSolicitud		
			idTransmision		
DatosEspecificos					
SolicitudTransm. n					

L'estructura de les peticions consta de dos elements principals:

1. Atributs. Aquest element conté, entre d'altres, la informació sobre el servei que es vol invocar.

2. Sol·licituds. Una petició pot contenir una o varies sol·licituds per a un mateix servei. Cada una d'aquestes sol·licituds conté dos blocs d'informació ben diferenciats:

- Datos genéricos: estructura amb la informació comuna per a qualsevol servei.
- Datos específicos: estructura amb la informació específica per al servei que s'està invocant.

## 6.2. Respuesta

Aquest missatge s'utilitza per a retornar la resposta a les peticions SCSP.

Respuesta	Atributos	idPeticion			
		numElementos			
		codigoCertificado			
		estado			
		timestamp			
	Transmisiones	TransmisionDatos	DatosGenericos	Emisor	nifEmisor
				nombreEmisor	
				Solicitante	Consentim.
					Finalidad
					Funcionario
					IdentificadorSol.
					NombreSol.
					idExpediente
					Procedimiento
					UnidadTramit.
				Titular	Nombre
					Apellido1
					Apellido2
					NombreComp.
					Documentacion
					TipoDocument.
				Transmis.	CodigoTrans.
					FechaGen.
					idSolicitud
					idTransmision
		DatosEspecificos			
		TransmisionDatos n			

L'estructura de les respostes SCSP és anàloga a l'estructura de les peticions SCSP. La única diferència és que en les respostes l'element «Solicitudes» es substitueix per «Transmisiones» i l'element «SolicitudTransmision» es substitueix per «TransmisionDatos».

### 6.3. ConfirmacionPeticion

Aquest missatge es retorna com a resposta a una petició SCSP asíncrona.

ConfirmacionPeticion	Atributos	idPeticion
		numElementos
		codigoCertificado
		estado
		Timestamp

### 6.4. Detall dels elements

Atributos	
idPeticion	Identificador únic per a la petició actual.
numElementos	Nombre de sol·licituds que conté aquesta petició.
codigoCertificado	El codi del servei al qual va dirigida aquesta petició.
estado	L'estat de la petició. Els possibles valors son: <ul style="list-style-type: none"> <li>• 0001: Pendent.</li> <li>• 0002: En procés.</li> <li>• 0003: Tramitada.</li> <li>• 0004: En procés de polling.</li> </ul>
timestamp	La data en la que es va crear aquesta petició.

Emisor	
nifEmisor	NIF de l'entitat que ofereix el servei que es vol consultar.
nombreEmisor	Nom de l'entitat que ofereix el servei que es vol consultar.

Solicitante		
consentimiento	Indica el tipus de consentiment atorgat per a realitzar aquesta petició.	
finalidad	Descripció de la finalitat amb la qual es realitza aquesta petició.	
funcionario	Camp compost amb informació del funcionari que realitza la petició.	
	nifFuncionario	NIF del funcionari

	<table> <tr> <td>nombreFuncionario</td><td>Nom del funcionari.</td></tr> </table>	nombreFuncionario	Nom del funcionari.		
nombreFuncionario	Nom del funcionari.				
identificadorSolicitante	Codi identificatiu de l'administració pública que realitza la petició.				
nombreSolicitante	Nom de l'administració pública que realitza la petició.				
idExpediente	Identificador d'expedient associat a la petició (opcional).				
procedimiento	Camp compost que identifica el procediment a dins el qual s'està realitzant aquesta petició. <table> <tr> <td>codigoProcedimiento</td><td>Codi del procediment.</td></tr> <tr> <td>NombreProcedimiento</td><td>Nom del procediment.</td></tr> </table>	codigoProcedimiento	Codi del procediment.	NombreProcedimiento	Nom del procediment.
codigoProcedimiento	Codi del procediment.				
NombreProcedimiento	Nom del procediment.				
unidadTramitadora	Nom de la unitat o departament a la qual pertany el procediment.				

Titular	
nombre	Nom del titular de la petició.
apellido1	Primer llinatge del titular de la petició.
apellido2	Segon llinatge del titular de la petició.
nombreCompleto	Nom complet del titular de la petició.
documentacion	Nombre del document d'identificació del titular.
tipoDocumentacion	Tipus del document d'identificació del titular.

Transmision	
codigoTransmision	Nom del titular de la petició.
fechaGeneracion	Primer llinatge del titular de la petició.
idSolicitud	Segon llinatge del titular de la petició.
idTransmision	Nom complet del titular de la petició.
documentacion	Nombre del document d'identificació del titular.
tipoDocumentacion	Tipus del document d'identificació del titular.

El bloc de DatosEspecificos es defineix per a cada servei i s'ha de consultar la documentació del servei que es vol consultar per a conèixer l'estructura XML necessària per a fer les peticions.



## Annex II: Possibles estats de les peticions

Codi	Descripció
0001	Pendent
0002	En procés
0003	Tramitada
0004	En procés de polling

## Annex III: WSDL del servei de recobriment

```
<?xml version="1.0" encoding="UTF-8"?>

<wsdl:definitions name="RecobrimentService"
  targetNamespace="http://www.caib.es/pinbal/ws/recobriment"
  xmlns:ns1="http://schemas.xmlsoap.org/wsdl/soap/http"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tns="http://www.caib.es/pinbal/ws/recobriment"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <wsdl:types>
    <xs:schema elementFormDefault="unqualified"
      targetNamespace="http://www.caib.es/pinbal/ws/recobriment"
      version="1.0" xmlns:tns="http://www.caib.es/pinbal/ws/recobriment"
      xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:element name="getRespuesta" type="tns:getRespuesta" />
      <xs:element name="getRespuestaResponse"
        type="tns:getRespuestaResponse" />
      <xs:element name="peticionAsincrona"
        type="tns:peticionAsincrona" />
      <xs:element name="peticionAsincronaResponse"
        type="tns:peticionAsincronaResponse" />
      <xs:element name="peticionSincrona"
        type="tns:peticionSincrona" />
      <xs:element name="peticionSincronaResponse"
        type="tns:peticionSincronaResponse" />
      <xs:complexType name="getRespuesta">
        <xs:sequence>
          <xs:element minOccurs="0" name="idPeticion"
            type="xs:string" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="getRespuestaResponse">
        <xs:sequence>
          <xs:element minOccurs="0" name="return"
            type="tns:respuesta" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="respuesta">
        <xs:sequence>
          <xs:element minOccurs="0" name="atributos"
            type="tns:atributos" />
          <xs:element minOccurs="0" name="transmisiones"
            type="tns:transmisiones" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="atributos">
        <xs:sequence>
          <xs:element minOccurs="0" name="codigoCertificado"
            type="xs:string" />
          <xs:element minOccurs="0" name="estado"
            type="tns:estado" />
          <xs:element minOccurs="0" name="idPeticion"
            type="xs:string" />
          <xs:element minOccurs="0" name="numElementos"
            type="xs:string" />
          <xs:element minOccurs="0" name="timeStamp"
            type="xs:string" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="estado">
        <xs:sequence>
          <xs:element minOccurs="0" name="codigoEstado"
            type="xs:string" />
          <xs:element minOccurs="0" name="codigoEstadoSecundario"
            type="xs:string" />
          <xs:element minOccurs="0" name="literalError"
            type="xs:string" />
          <xs:element minOccurs="0" name="tiempoEstimadoRespuesta"
            type="xs:string" />
        </xs:sequence>
      </xs:complexType>
    </xs:schema>
  </wsdl:types>

```

```

                                type="xs:int" />
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="transmisiones">
        <xs:sequence>
            <xs:element maxOccurs="unbounded" minOccurs="0"
                name="transmisionDatos" nillable="true"
                type="tns:transmisionDatos" />
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="transmisionDatos">
        <xs:sequence>
            <xs:element minOccurs="0" name="datosEspecificos"
                type="xs:anyType" />
            <xs:element minOccurs="0" name="datosGenericos"
                type="tns:datosGenericos" />
            <xs:element minOccurs="0" name="id" type="xs:string" />
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="datosGenericos">
        <xs:sequence>
            <xs:element minOccurs="0" name="emisor"
                type="tns:emisor" />
            <xs:element minOccurs="0" name="solicitante"
                type="tns:solicitante" />
            <xs:element minOccurs="0" name="titular"
                type="tns:titular" />
            <xs:element minOccurs="0" name="transmision"
                type="tns:transmision" />
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="emisor">
        <xs:sequence>
            <xs:element minOccurs="0" name="nifEmisor"
                type="xs:string" />
            <xs:element minOccurs="0" name="nombreEmisor"
                type="xs:string" />
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="solicitante">
        <xs:sequence>
            <xs:element minOccurs="0" name="consentimiento"
                type="tns:consentimiento" />
            <xs:element minOccurs="0" name="finalidad"
                type="xs:string" />
            <xs:element minOccurs="0" name="funcionario"
                type="tns:funcionario" />
            <xs:element minOccurs="0" name="idExpediente"
                type="xs:string" />
            <xs:element minOccurs="0"
                name="identificadorSolicitante" type="xs:string" />
            <xs:element minOccurs="0" name="nombreSolicitante"
                type="xs:string" />
            <xs:element minOccurs="0" name="procedimiento"
                type="tns:procedimiento" />
            <xs:element minOccurs="0" name="unidadTramitadora"
                type="xs:string" />
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="funcionario">
        <xs:sequence>
            <xs:element minOccurs="0" name="nifFuncionario"
                type="xs:string" />
            <xs:element minOccurs="0"
                name="nombreCompletoFuncionario" type="xs:string" />
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="procedimiento">
        <xs:sequence>
            <xs:element minOccurs="0" name="codProcedimiento"
                type="xs:string" />
            <xs:element minOccurs="0" name="nombreProcedimiento"
                type="xs:string" />

```

```

</xs:sequence>
</xs:complexType>
<xs:complexType name="titular">
  <xs:sequence>
    <xs:element minOccurs="0" name="apellido1"
      type="xs:string" />
    <xs:element minOccurs="0" name="apellido2"
      type="xs:string" />
    <xs:element minOccurs="0" name="documentacion"
      type="xs:string" />
    <xs:element minOccurs="0" name="nombre" type="xs:string" />
    <xs:element minOccurs="0" name="nombreCompleto"
      type="xs:string" />
    <xs:element minOccurs="0" name="tipoDocumentacion"
      type="tns:tipoDocumentacion" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="transmision">
  <xs:sequence>
    <xs:element minOccurs="0" name="codigoCertificado"
      type="xs:string" />
    <xs:element minOccurs="0" name="fechaGeneracion"
      type="xs:string" />
    <xs:element minOccurs="0" name="idSolicitud"
      type="xs:string" />
    <xs:element minOccurs="0" name="idTransmision"
      type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType abstract="true"
  name="abstractCollection">
  <xs:sequence />
</xs:complexType>
<xs:complexType abstract="true" name="abstractList">
  <xs:complexContent>
    <xs:extension base="tns:abstractCollection">
      <xs:sequence />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="arrayList">
  <xs:complexContent>
    <xs:extension base="tns:abstractList">
      <xs:sequence />
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="peticionAsincrona">
  <xs:sequence>
    <xs:element minOccurs="0" name="peticion"
      type="tns:peticion" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="peticion">
  <xs:sequence>
    <xs:element minOccurs="0" name="atributos"
      type="tns:atributos" />
    <xs:element minOccurs="0" name="solicitudes"
      type="tns:solicitudes" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="solicitudes">
  <xs:sequence>
    <xs:element maxOccurs="unbounded" minOccurs="0"
      name="solicitudTransmision" nillable="true"
      type="tns:solicitudTransmision" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="solicitudTransmision">
  <xs:sequence>
    <xs:element minOccurs="0" name="datosEspecificos"
      type="xs:anyType" />
    <xs:element minOccurs="0" name="datosGenericos"

```

```

        type="tns:datosGenericos" />
      <xs:element minOccurs="0" name="id" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="peticionAsincronaResponse">
    <xs:sequence>
      <xs:element minOccurs="0" name="return"
        type="tns:confirmacionPetición" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="confirmacionPetición">
    <xs:sequence>
      <xs:element minOccurs="0" name="atributos"
        type="tns:atributos" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="peticiónSincrona">
    <xs:sequence>
      <xs:element minOccurs="0" name="petición"
        type="tns:petición" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="peticiónSincronaResponse">
    <xs:sequence>
      <xs:element minOccurs="0" name="return"
        type="tns:respuesta" />
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="consentimiento">
    <xs:restriction base="xs:string">
      <xs:enumeration value="Si" />
      <xs:enumeration value="Ley" />
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="tipoDocumentacion">
    <xs:restriction base="xs:string">
      <xs:enumeration value="CIF" />
      <xs:enumeration value="NIF" />
      <xs:enumeration value="DNI" />
      <xs:enumeration value="Pasaporte" />
      <xs:enumeration value="NIE" />
    </xs:restriction>
  </xs:simpleType>
</xs:schema>
</wsdl:types>
<wsdl:message name="getResponseResponse">
  <wsdl:part element="tns:getRespuestaResponse"
    name="parameters">
  </wsdl:part>
</wsdl:message>
<wsdl:message name="getRespuesta">
  <wsdl:part element="tns:getRespuesta" name="parameters">
  </wsdl:part>
</wsdl:message>
<wsdl:message name="peticiónSincrona">
  <wsdl:part element="tns:peticiónSincrona" name="parameters">
  </wsdl:part>
</wsdl:message>
<wsdl:message name="peticiónSincronaResponse">
  <wsdl:part element="tns:peticiónSincronaResponse"
    name="parameters">
  </wsdl:part>
</wsdl:message>
<wsdl:message name="peticiónAsincrona">
  <wsdl:part element="tns:peticiónAsincrona" name="parameters">
  </wsdl:part>
</wsdl:message>
<wsdl:message name="peticiónAsincronaResponse">
  <wsdl:part element="tns:peticiónAsincronaResponse"
    name="parameters">
  </wsdl:part>
</wsdl:message>
<wsdl:portType name="Recobrimiento">

```

```
<wsdl:operation name="getRespuesta">
  <wsdl:input message="tns:getRespuesta" name="getRespuesta">
  </wsdl:input>
  <wsdl:output message="tns:getRespuestaResponse"
    name="getRespuestaResponse">
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="peticionAsincrona">
  <wsdl:input message="tns:peticionAsincrona"
    name="peticionAsincrona">
  </wsdl:input>
  <wsdl:output message="tns:peticionAsincronaResponse"
    name="peticionAsincronaResponse">
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="peticionSincrona">
  <wsdl:input message="tns:peticionSincrona"
    name="peticionSincrona">
  </wsdl:input>
  <wsdl:output message="tns:peticionSincronaResponse"
    name="peticionSincronaResponse">
  </wsdl:output>
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="RecobrimientServiceSoapBinding"
  type="tns:Recobrimient">
  <soap:binding style="document"
    transport="http://schemas.xmlsoap.org/soap/http" />
  <wsdl:operation name="getRespuesta">
    <soap:operation soapAction="" style="document" />
    <wsdl:input name="getRespuesta">
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output name="getRespuestaResponse">
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="peticionAsincrona">
    <soap:operation soapAction="" style="document" />
    <wsdl:input name="peticionAsincrona">
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output name="peticionAsincronaResponse">
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="peticionSincrona">
    <soap:operation soapAction="" style="document" />
    <wsdl:input name="peticionSincrona">
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output name="peticionSincronaResponse">
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
<wsdl:service name="RecobrimientService">
  <wsdl:port binding="tns:RecobrimientServiceSoapBinding"
    name="RecobrimientServicePort">
    <soap:address
      location="http://localhost:8080/pinbal/ws/recobrimient" />
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```