JUNTA DE ANDALUCIA

CONSEJERÍA DE HACIENDA Y ADMINISTRACIÓN PÚBLICA

Dirección General de Política Digital

Carpeta Ciudadana de la Junta de Andalucía

Toolkit de integración

Versión: v1.1.1r00
Fecha: 14/03/2018

Queda prohibido cualquier tipo de explotación y, en particular, la reproducción, distribución, comunicación pública y/o transformación, total o parcial, por cualquier medio, de este documento sin el previo consentimiento expreso y por escrito de la Junta de Andalucía.

HOJA DE CONTROL

Título	Toolkit de integración		
Entregable	Toolkit de integración		
Nombre del Fichero	CC_Toolkit_v1.1.1r00.odt		
Autor	OTGP		
Versión/Edición	V01.1r01	**Fecha Versión**	14/03/2018
Aprobado por		**Fecha Aprobación**	
		Nº Total Páginas	

REGISTRO DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Área	Fecha del Cambio
V1.0r00	Versión inicial	OTGP	OTGP	26/01/2018
v1.1r00	Revisión	OTGP	OTGP	06/02/2018
v1.1r01	Revisión	OTGP	OTGP	20/02/2018
v.1.1.1r00	Revisión	OTGP	OTGP	14/03/2018

CONTROL DE DISTRIBUCIÓN

Nombre y Apellidos	Cargo	Área	Nº Copias
Manuel Perera Domínguez	Jefe de servicio	Servicio de Coordinación Administrativa Electrónica	1
Francisco González Guillén	Adjunto	Servicio de Planificación Estratégica	1

ÍNDICE

1 INTRODUCCIÓN 4
2 ARQUITECTURA 6
3 INTEGRACIÓN DE SERVICIOS 7
3.1 Configuración de base de datos 7
3.2 Definición de consultas 8
3.3 Elaboración de esquema XML 8
3.4 Configuración de plantillas 9
4 EJEMPLOS DE INTEGRACIÓN 10
4.1 Configuración de base de datos 10
4.2 Configuración del servicio 10
4.2.1 Ejemplo de servicio de información 11
4.2.2 Ejemplo de servicio de expedientes 14
5 NOTAS AL DESARROLLO DE SERVICIOS 25
5.1 Servicio tipo 'Mi información' ad hoc 25
5.2 Servicio tipo 'Mi información' SCSP 25
5.3 Servicio tipo 'Mis expedientes' 27
6 CONFIGURACIÓN DEL SERVICIO 32
6.1 Despliegue de servicios usando Axis 2 32
6.1.1 Configuración cabeceras de seguridad 35
7 CONFIGURACIÓN DE LA LIBRERÍA 37
7.1 Definición de servicios 37
7.2 Consultas de datos 37
7.3 Mi información 40
7.4 Mis expedientes 42
7.5 Conexiones a bases de datos 45
8 API DEL TOOLKIT 49
9 DEPENDENCIAS 50
10 NOTAS DE LA VERSIÓN 51
11 CAMBIOS DE LA VERSIÓN 51

1 INTRODUCCIÓN

Este documento tiene por objeto describir la funcionalidad y uso del toolkit de integración de Carpeta Ciudadana de la Junta de Andalucía.

Carpeta Ciudadana es una funcionalidad para impulsar, homogeneizar y simplificar la relación electrónica con la ciudadanía, permitiéndole un acceso sencillo y rápido a diferentes tipos de información relacionada con sus trámites. Entre esa información se encuentra:

- Información sobre los expedientes que tiene en las Consejerías de la Junta de Andalucía.
- Información de carácter personal, en especial la relativa a certificados, tarjetas, carnets, etc.

Carpeta Ciudadana no almacena documentos ni datos de los expedientes de la ciudadanía; no almacena información de manera centralizada sino que la recaba de los sistemas de información de las Consejerías y entidades de la Junta de Andalucía que actúan como proveedores de la información. Existen dos posibilidades técnicas de integración: una específica para la información sobre expedientes y otra para la información sobre certificados, tarjetas, carnets, etc.

Carpeta Ciudadana en la sección denominada **Mis expedientes** ofrecerá la información de los expedientes, tanto abiertos como cerrados, agrupados por Consejería. Además también se ofrecerá la posibilidad de consultar la relación de expedientes en una Consejería y en su caso el detalle de un expediente en concreto.

Carpeta Ciudadana a través de la sección denominada **Mi información** ofrecerá la posibilidad de consultar información de carácter personal, entre la que se encuentra la consulta de certificados, tarjetas, carnets, etc.

El toolkit de integración es un componente que pretende simplificar las tareas de desarrollo necesarias para que un organismo pueda ofrecer a Carpeta Ciudadana información para ambas secciones desde sus sistemas de información.

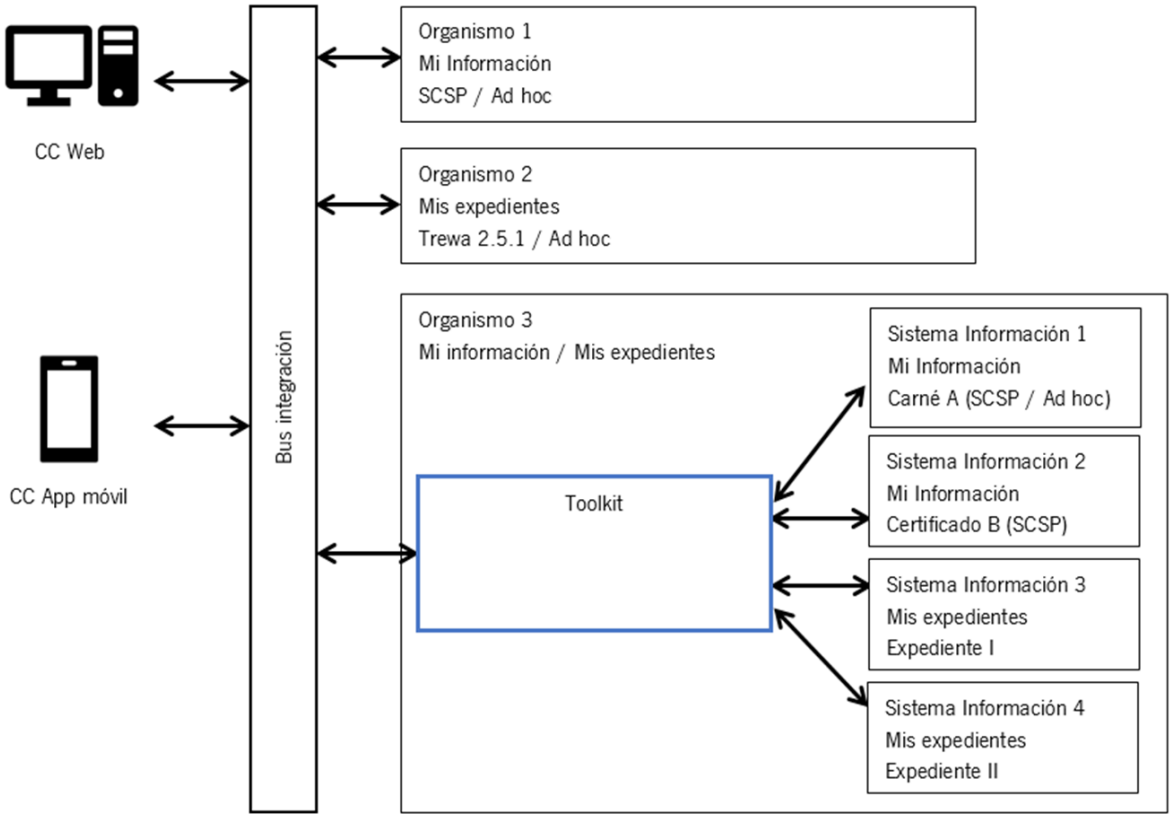
Mediante configuración permite definir distintos tipos de servicios de consulta de datos:

- Consulta de certificados, tarjetas, carnets, etcétera con interfaz ad hoc.
- Consulta de certificados, tarjetas, carnets, etcétera con interfaz SCSP (sustitución de certificados en soporte papel). Información sobre el estándar SCSP en el portal de Administración Electrónica.

  https://administracionelectronica.gob.es/ctt/scsp

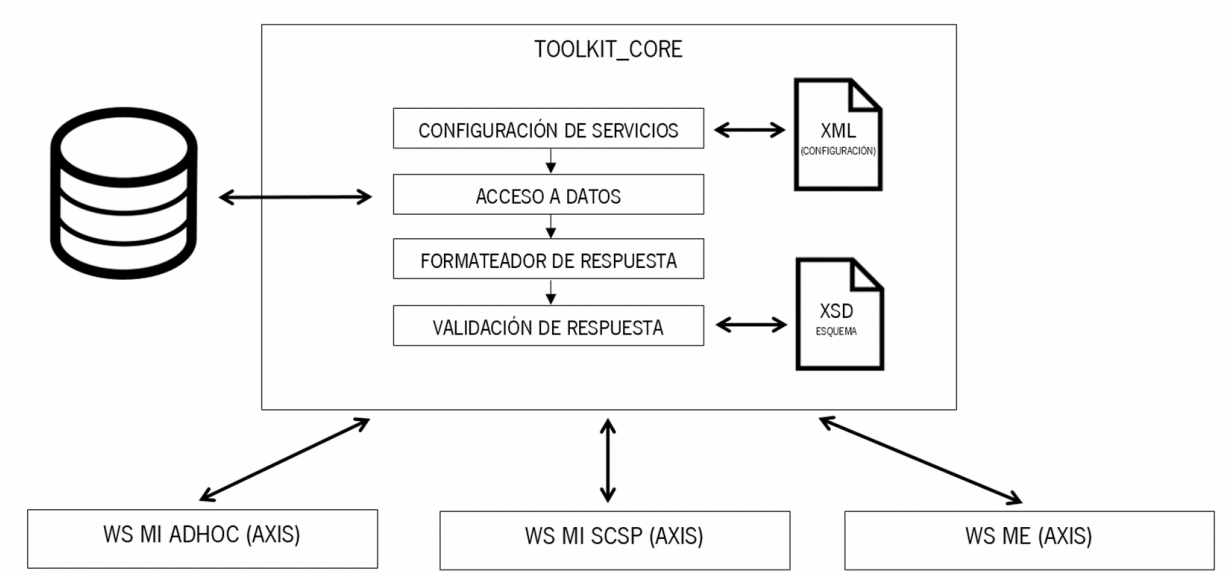
- Consulta de expedientes conforme a contrato de servicio de Carpeta Ciudadana, en el portal de Administración Electrónica de la Junta de Andalucía.

  https://ws024.juntadeandalucia.es/ae/adminelec/areatecnica/carpeta


CC Web
CC App móvil
Bus integración
Organismo 1
Mi Información
SCSP / Ad hoc
Organismo 2
Mis expedientes
Trewa 2.5.1 / Ad hoc
Organismo 3
Mi información / Mis expedientes
Toolkit
Sistema Información 1
Mi Información
Carné A (SCSP / Ad hoc)
Sistema Información 2
Mi Información
Certificado B (SCSP)
Sistema Información 3
Mis expedientes
Expediente I
Sistema Información 4
Mis expedientes
Expediente II

2 ARQUITECTURA

El toolkit de integración se distribuye como una librería Java que implementa funcionalidad básica de acceso a datos y generación de respuestas XML conformes a los contratos de integración de Carpeta Ciudadana y el estándar SCSP. Esta librería deberá ser integrada en cada sistema por el organismo correspondiente y servirá de apoyo para facilitar la implementación de los servicios requeridos por Carpeta Ciudadana. Junto con dicha librería se proporcionan a modo de referencia unos servicios web basados en tecnología Axis 2 para cada una de las modalidades de integración soportadas.



Los elementos incluidos en la distribución del toolkit son los siguientes:

- **toolkit-core:** Librería Java con funcionalidad básica para implementación de servicios de consulta de datos. Habilita la consulta de datos para las secciones 'Mi información' y 'Mis expedientes' de Carpeta Ciudadana a partir de la definición de consulta/s SQL y esquema XML (XSD) necesarios para formar y validar la respuesta de un servicio de consulta.

- **consultamiadhoc.aar:** Ejemplo de integración de servicio de consulta de datos para sección 'Mi información' con interfaz ad hoc, no estandarizada. Se proporciona como servicio web empaquetado para Axis 2.

- **consultamiscsp.aar:** Ejemplo de integración de servicio de consulta de datos para sección 'Mi información' con interfaz conforme a estándar SCSP. Se proporciona como servicio web empaquetado para Axis 2.

- **consultame.aar:** Ejemplo de integración de servicio de consulta de datos para sección 'Mis expedientes' con interfaz conforme al contrato de servicio de Carpeta Ciudadana. Se proporciona como servicio web empaquetado para Axis 2. Este servicio requiere la librería 'toolkit-ws-handler-axis'.

    - **toolkit-ws-handler-axis**: Manejador Axis a configurar en instalaciones del toolkit que proporcionen información de expedientes de múltiples sistemas de información.

3 INTEGRACIÓN DE SERVICIOS

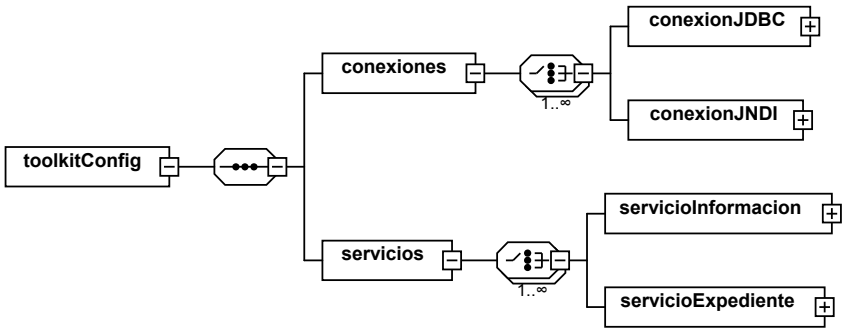
Las tareas básicas que se deben abordar para configurar un servicio del toolkit son las siguientes:

- Configuración de la conexión de base de datos que utilizará el servicio.
- Definición de la/s consulta/s que permitirá la recuperación de los datos del servicio.

Si el servicio es para la sección 'Mi información' será necesario que cada organismo realice las siguientes tareas:

- Elaborar el esquema XML (XSD) para validación de las respuestas generadas por el servicio.
- Configurar las plantillas XML para las distintas consultas SQL que componen la respuesta del servicio.

Toda esta información se proporcionará en el fichero de configuración del toolkit **configuracion-toolkit.xml** y será descrita en detalle en los apartados siguientes.



3.1 Configuración de base de datos

Cada servicio del toolkit requiere una conexión a base de datos. Existen dos posibilidades de configuración:

- Pool de conexiones gestionado por el toolkit.
- Pool de conexiones definido por el servidor de aplicaciones y recuperada por el toolkit vía JNDI.

Toda conexión debe definir un identificador que será utilizado por los servicios que hagan uso de dicha conexión.

3.2 Definición de consultas

Un servicio requiere la configuración de una o varias consultas. Cada consulta está formada por:

- Una sentencia SQL parametrizada.
- Una relación de parámetros de entrada que serán los que se empleen en la ejecución de la consulta. Para cada parámetro será necesario definir su tipo de dato, el orden y un identificador. Este identificador deberá coincidir con el nombre de alguno de los parámetros de la petición del servicio. Por ejemplo si un parámetro corresponde a una consulta del servicio de consulta agrupada de la sección 'Mis expedientes', su nombre deberá tener alguno de estos valores: docIdentidad, apellido1, etcétera definidos en el contrato de integración.
- Una relación de parámetros de salida que serán los usados para formar la respuesta a entregar a Carpeta Ciudadana. Para cada parámetro será necesario definir su tipo de dato y un identificador. Este identificador deberá coincidir con el nombre de alguno de los parámetros de la respuesta del servicio. Por ejemplo si un parámetro corresponde a una consulta del servicio de consulta agrupada de la sección 'Mis expedientes', deberá tener alguno de estos valores: numeroExpedientes, enlace, etcétera definidos en el contrato de integración. Para simplificar la configuración se recomienda que los nombres de los campos devueltos en la consulta SQL coincidan con los identificadores de los parámetros de salida correspondientes. Si no fuera posible se deberá utilizar el atributo 'columna' del parámetro para establecer la relación.

Opcionalmente se pueden definir subconsultas, las cuales son consultas cuyos parámetros de entrada se obtienen de consultas que deben ser ejecutadas con anterioridad.

Será responsabilidad de los organismos optimizar las consultas (mediante vistas materializadas, índices, etc.) de forma que se minimicen los tiempos de respuesta y no se superen los tiempos máximos definidos en los contratos.

3.3 Elaboración de esquema XML

Toda respuesta generada por el toolkit es validada respecto al esquema XML (XSD) configurado para el servicio. Si la respuesta es incorrecta o no se dispone del esquema correspondiente se generará un error informando sobre el problema. Este esquema se deberá proporcionar para servicios 'Mi información' con independencia de su tipo. A continuación se muestra un ejemplo de fichero XSD:

```
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.example.org/prueba1"
        xmlns:tns="http://www.example.org/prueba1" elementFormDefault="qualified">

        <element name="informacion_carnet">
                <complexType>
                        <sequence>
```

```
                        <element name="carnet" type="tns:CarnetType" maxOccurs="1" />
                    </sequence>
                </complexType>
        </element>

        <complexType name="CarnetType">
                <sequence>
                        <element name="nombre" type="string" minOccurs="1" maxOccurs="1"/>
                        <element name="apellido1" type="string" minOccurs="1" maxOccurs="1"/>
                        <element name="apellido2" type="string" minOccurs="0" maxOccurs="1"/>
                        <element name="dni" type="string" minOccurs="1" maxOccurs="1"/>
                        <element name="cb" type="string" minOccurs="1" maxOccurs="1"/>
                        <element name="fecha" type="date" minOccurs="1" maxOccurs="1"/>
                        <element name="nombre_biblioteca" type="string" minOccurs="0"
maxOccurs="1"/>
                </sequence>
        </complexType>

</schema>
```

3.4 Configuración de plantillas

El toolkit utiliza el estándar XML en la generación de las respuestas a las peticiones de los distintos servicios de consulta de datos. En los servicios de tipo 'Mi información' el organismo que se integre con Carpeta Ciudadana vía toolkit deberá proporcionar el esquema XML (XSD) que define las construcciones válidas para el servicio concreto que se está definiendo.

Un servicio puede necesitar varias consultas para crear la respuesta a una petición. Cada consulta generará una estructura XML que puede ser una parte de la respuesta o toda ella. Dicha estructura se especificará para cada consulta en la configuración mediante una plantilla y en ella se marcará la posición en la aparecerá cada dato de la sentencia SQL.

Por ejemplo una consulta que defina un parámetro de salida COL1 y como plantilla el texto <prueba><dato>%COL1%</dato></prueba> generará una línea de XML por cada registro devuelto por la consulta y sustituirá la cadena %COL1% por el valor recuperado para el campo con nombre COL1.

Para agrupar el texto XML generado por las distintas consultas existe una serie de elementos en la configuración del toolkit:

- nodoRaiz: Permite generar un elemento XML raíz bien para la respuesta completa al servicio o bien para los resultados de una consulta.
- nodoTupla: Permite generar un elemento XML que englobe cada tupla devuelta por una consulta SQL.

4 EJEMPLOS DE INTEGRACIÓN

Se describen a continuación ejemplos completos de integración para los distintos tipos de servicio.

Para ello se comentarán todas las modificaciones a realizar al fichero configuracion-toolkit.xml. Este fichero se encuentra ubicado en el directorio raíz de la librería toolkit-core. Su estructura se debe ajustar al esquema disponible en la librería en la ruta xsd/configuracion/configuracion-toolkit.xsd. Este esquema se encuentra descrito en detalle en el apartado "7. Configuración de la librería" de este documento.

4.1 Configuración de base de datos

El primer apartado que será necesario configurar es el relativo a la definición de las conexiones a base de datos del organismo. A continuación se muestra un ejemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<toolkitConfig xmlns="urn:juntadeandalucia:chap:cicc:v1.0:configuracion"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="urn:juntadeandalucia:chap:cicc:v1.0:configuracion configuracion-toolkit.xsd
">
        <conexiones>
                <conexionJDBC>
                        <identificador>ID1</identificador>
                        <descripcion>Conexión para pruebas 1</descripcion>
                        <driver>oracle.jdbc.driver.OracleDriver</driver>
                        <usuario>TOOLKIT</usuario>
                        <clave>TOOLKIT</clave>
                        <url>jdbc:oracle:thin:@localhost:1521:xe</url>
                </conexionJDBC>
                <conexionJNDI>
                        <identificador>ID2</identificador>
                        <descripcion>Conexión JNDI para pruebas 1</descripcion>
                        <datasource>jdbc/toolkit</datasource>
                </conexionJNDI>
        </conexiones>
```

Como se puede apreciar en el ejemplo, se pueden definir múltiples conexiones a bases de datos distintas, ya sea mediante cadenas de conexión JDBC directas, o bien delegando en conexiones JNDI definidas en el servidor de aplicaciones. Cada conexión definirá un identificador que permitirá posteriormente a los servicios especificar la conexión a utilizar en sus consultas.

4.2 Configuración del servicio

El siguiente apartado es el relativo a la definición de servicios. Estos deberán ser definidos dentro de la etiqueta <servicios> y podrán ser de tipo <servicioInformacion> o bien <servicioExpediente> en función de la sección de Carpeta Ciudadana con la que se integre.

4.2.1 Ejemplo de servicio de información

A continuación se detalla el ejemplo de un servicio de información tipo ad hoc.

```
<servicioInformacion idServicio="CARNET_BIBLIOTECA" idConexion="ID2" tipo="adhoc">
        <esquema>/ruta/respuesta_carnet_biblioteca.xsd</esquema>

<namespace>urn:juntadeandalucia:chap:v1.0:servicios:ciudadania:carnetBibliotecaResponse</namespace>
        <nodoRaiz>informacion_carnet</nodoRaiz>
        <consultas>
                <consulta idConsulta="IDC1">
                        <sentenciaSQL>
                        <![CDATA[
                        SELECT NOMBRE, APELL_1, APELL_2, NIF, CB, FECHA_ALTA, DESC_BIBLIOTECA
                          FROM VIEW1
                         WHERE IDENTIFICADOR = ?
                           AND NOMBRE_RAZON = ?
                           AND APELLIDO1 = ?
                        ]]>
                        </sentenciaSQL>
                        <parametros>
                                <entrada>
                                        <parametro nombre="identificador" tipo="string" orden="1" />
                                        <parametro nombre="nombreRazonSocial" tipo="string" orden="2" />
                                        <parametro nombre="apellido1" tipo="string" orden="3" />
                                </entrada>
                                <salida>
                                        <parametro nombre="NOMBRE" tipo="string" />
                                        <parametro nombre="APELL_1" tipo="string" />
                                        <parametro nombre="APELL_2" tipo="string" />
                                        <parametro nombre="NIF" tipo="string" />
                                        <parametro nombre="CB" tipo="integer" />
                                        <parametro nombre="FECHA" columna="FECHA_ALTA" tipo="date" />
                                        <parametro nombre="DESC_BIBLIOTECA" tipo="string" />
                                </salida>
                        </parametros>
                        <plantillaXML>
                        <![CDATA[
                                <carnet>
                                        <nombre>%NOMBRE%</nombre>
                                        <apellido1>%APELL_1%</apellido1>
                                        <apellido2>%APELL_2%</apellido2>
                                        <dni>%NIF%</dni>
                                        <cb>%CB%</cb>
                                        <fecha>%FECHA%</fecha>
                                        <nombre_biblioteca>%DESC_BIBLIOTECA%</nombre_biblioteca>
                                </carnet>
                        ]]>
                        </plantillaXML>
                </consulta>
        </consultas>
</servicioInformacion>
```

Como se puede observar, cada servicio de información deberá tener asociado una serie de atributos: un identificador (idServicio), una conexión de base de datos definida previamente (idConexion), y un tipo de servicio (tipo) que podrá ser “adhoc” o “scsp”.

El primer parámetro, <esquema> es el encargado de definir la ruta externa al fichero xsd encargado de validar la respuesta ofrecida por el organismo. La ruta definida debe ser accesible desde el servidor donde se ejecute el toolkit.

A continuación se deberá definir el namespace que se empleará en los XML generados como respuesta. El namespace a definir deberá seguir el siguiente patrón:

"urn:juntadeandalucia:chap:v1.0:servicios:ciudadania:[nombre_servicio]*"*

Un ejemplo sería:

"*urn:juntadeandalucia:chap:v1.0:servicios:ciudadania:carnetBiblioteca*"

Se definirá el identificador del elemento que se empleará como raíz de la respuesta por medio del elemento de configuración <nodoRaiz>. En este caso concreto se asigna el valor "informacion_carnet" a <nodoRaiz>, por lo tanto la respuesta asociada a la ejecución de este servicio será de la forma <informacion_carnet>xml consultas</informacion_carnet>.

La sección <consultas> permite especificar cada una de las consultas cuya ejecución va a permitir la obtención de la respuesta del servicio. Cuando un servicio se defina con varias consultas el toolkit calculará la respuesta correspondiente concatenando los XML generados para cada consulta atendiendo al orden en el que estas se han definido en la configuración.

Cada consulta tendrá un atributo donde se defina su identificador (idConsulta). Si la consulta requiere de una conexión de base de datos distinta a la definida para el servicio, podrá especificarla mediante el atributo idConexion.

En el elemento <sentenciaSQL> se definirá la consulta en lenguaje SQL que deberá ser lanzada para la obtención de datos. La parametrización se hará mediante el uso del símbolo "?".

```
<sentenciaSQL>
        <![CDATA[
        SELECT NOMBRE, APELL_1, APELL_2, NIF, CB, FECHA_ALTA, DESC_BIBLIOTECA
          FROM VIEW1
         WHERE IDENTIFICADOR = ?
           AND NOMBRE_RAZON = ?
           AND APELLIDO1 = ?
        ]]>
</sentenciaSQL>
```

La parametrización tanto de entrada como de salida deberá ser definida en la sección de parámetros.

```
<parametros>
        <entrada>
                <parametro nombre="identificador" tipo="string" orden="1" />
                <parametro nombre="nombreRazonSocial" tipo="string" orden="2" />
                <parametro nombre="apellido1" tipo="string" orden="3" />
        </entrada>
        <salida>
                <parametro nombre="NOMBRE" tipo="string" />
                <parametro nombre="APELL_1" tipo="string" />
                <parametro nombre="APELL_2" tipo="string" />
                <parametro nombre="NIF" tipo="string" />
                <parametro nombre="CB" tipo="integer" />
                <parametro nombre="FECHA" columna="FECHA_ALTA" tipo="date" />
                <parametro nombre="DESC_BIBLIOTECA" tipo="string" />
        </salida>
</parametros>
```

Los parámetros de entrada deben ser tantos como caracteres "?" haya en la consulta definida previamente. Para cada parámetro se definirá un nombre que debe coincidir con alguno de los identificadores válidos para el tipo de servicio que se está definiendo. En el caso concreto de un servicio para 'Mi información' con interfaz ad hoc los valores posibles son "identificador", "nombreRazonSocial" y "apellido1". Los tipos de parámetros permitidos están indicados en la sección "7. Configuración de la librería".

Los parámetros de salida tendrán definido un nombre que posteriormente será utilizado en la definición de la plantilla XML de respuesta para ubicar los datos recuperados de la consulta. Con objeto de simplificar la configuración se recomienda que los nombres de los campos de la consulta SQL coincidan con los nombres asignados a los parámetros de salida. Caso de no seguirse dicho criterio será necesario definir el atributo "columna" para establecer la relación entre los identificadores.

El elemento <nodoRaiz> permite agrupar los resultados generados por la consulta.

El fragmento XML de una consulta se especificará haciendo uso del elemento <plantillaXML>.

```
<plantillaXML>
        <![CDATA[
                <carnet>
                        <nombre>%NOMBRE%</nombre>
                        <apellido1>%APELL_1%</apellido1>
                        <apellido2>%APELL_2%</apellido2>
                        <dni>%NIF%</dni>
                        <cb>%CB%</cb>
                        <fecha>%FECHA%</fecha>
                        <nombre_biblioteca>%DESC_BIBLIOTECA%</nombre_biblioteca>
                </carnet>
        ]]>
</plantillaXML>
```

Para simplificar el escapado de caracteres XML se definirá la plantilla en un bloque CDATA. En él se deberá definir el fragmento de XML relativo a dicha consulta que se quiere proporcionar como respuesta en los servicios.

En este XML se ubicarán los resultados obtenidos de la consulta a base de datos. Para ello se deberá hacer uso de literales que sigan el siguiente patrón: %NOMBRE_PARÁMETRO_DE_SALIDA%. De esta forma el toolkit se encargará de formar un xml de respuesta siguiendo esa estructura y sustituyendo los parámetros por los obtenidos en la consulta a base de datos.

4.2.2 Ejemplo de servicio de expedientes

Las respuestas ofrecidas por los servicios de expedientes deben respetar el formato definido en el contrato de servicios de Carpeta Ciudadana. El Toolkit se encarga de ello y por tanto se omiten las configuraciones relativas al formato de la respuesta y su validación, que sí eran necesarias para poder definir servicios de información. Para los servicios se omiten los siguientes parámetros: “esquema”, “namespace” y “nodoRaiz”. Para las consultas se omiten: “nodoRaiz” y “plantillaXML”.

Para definir un servicio de expedientes, por tanto, es suficiente con la configuración de las sentencias SQL y los parámetros de entrada/salida de cada consulta del servicio.

Para facilitar la labor a los integradores, los servicios de expedientes se han dividido a su vez en distintas consultas, simplificando así la definición de sentencias SQL.

Se abordará el detalle de cada consulta y sus parámetros en el apartado “5.3 Servicio tipo ‘Mis expedientes’”.

A continuación se ofrece un ejemplo de configuración de servicio de expedientes.

```
<servicioExpediente idServicio="SME1" idConexion="ID1">
                    <consultaAgrupada>
                           <sqlTotales idConsulta="SME1_AGRUP_TOTALES">
                                  <sentenciaSQL>
                                  <![CDATA[
                                     SELECT numeroExpedientes, (CASE total WHEN 0 THEN 'false' ELSE 'true' END) hayMasResultados, 'ENLACE' enlace
                                       FROM (SELECT COUNT(1) numeroExpedientes
                                               FROM EXPEDIENTES E1
                                      LEFT OUTER JOIN INTERESADOS S1 ON E1.SOLICITANTE_ID = S1.ID
                                      LEFT OUTER JOIN INTERESADOS R1 ON E1.REPRESENTANTE_ID = R1.ID
                                               WHERE S1.IDENTIFICADOR = ?
                                       AND R1.IDENTIFICADOR = COALESCE(?, R1.IDENTIFICADOR)
                                       AND E1.QAA <= ?
                                       AND E1.ESTADO = (CASE ? WHEN 'Total' THEN E1.ESTADO WHEN 'Abierto' THEN 'A' WHEN 'Finalizado' THEN 'F' END)
                                       AND E1.DIR3 = ?),
                                           (SELECT COUNT(1) total
                                               FROM EXPEDIENTES E2
                                      LEFT OUTER JOIN INTERESADOS S2 ON E2.SOLICITANTE_ID = S2.ID
                                      LEFT OUTER JOIN INTERESADOS R2 ON E2.REPRESENTANTE_ID = R2.ID
                                               WHERE S2.IDENTIFICADOR = ?
                                       AND S2.IDENTIFICADOR = COALESCE(?, S2.IDENTIFICADOR)
                                       AND E2.QAA > ?
                                       AND E2.ESTADO = (CASE ? WHEN 'Total' THEN E2.ESTADO WHEN 'Abierto' THEN 'A' WHEN 'Finalizado' THEN 'F' END)
                                       AND E2.DIR3 = ?)
                                        ]]>
                                  </sentenciaSQL>
                                  <parametros>
                                         <entrada>
                                                <parametro nombre="docIdentidad" tipo="string" orden="1"/>
                                                <parametro nombre="docIdentidadRep" tipo="string" orden="2" />
                                                <parametro nombre="qaa" tipo="integer" orden="3" />
                                                <parametro nombre="estado" tipo="string" orden="4" />
                                                <parametro nombre="codigoAgrupacion" tipo="string" orden="5" />
                                                <parametro nombre="docIdentidad" tipo="string" orden="6"/>
                                                <parametro nombre="docIdentidadRep" tipo="string" orden="7" />
                                                <parametro nombre="qaa" tipo="integer" orden="8" />
                                                <parametro nombre="estado" tipo="string" orden="9" />
                                                <parametro nombre="codigoAgrupacion" tipo="string" orden="10" />
                                         </entrada>
                                         <salida>
```

```
                        <parametro nombre="numeroExpedientes" tipo="integer" />
                        <parametro nombre="hayMasResultados" tipo="string" />
                        <parametro nombre="enlace" tipo="string" />
                </salida>
        </parametros>
    </sqlTotales>
    <sqlRpas idConsulta="SME1_AGRUP_RPAS">
        <sentenciaSQL>
        <![CDATA[
                SELECT DISTINCT RP.CODIGO codigo, RP.DESCRIPCION nombre, 'ENLACE' enlace,
                        (SELECT COUNT(1)
                           FROM EXPEDIENTES SE
                            LEFT OUTER JOIN RPAS RP ON SE.COD_RPA = RP.CODIGO
            LEFT OUTER JOIN INTERESADOS SS ON SE.SOLICITANTE_ID = SS.ID
            LEFT OUTER JOIN INTERESADOS SR ON SE.REPRESENTANTE_ID = SR.ID
                      WHERE SS.IDENTIFICADOR = ?
             AND SR.IDENTIFICADOR = COALESCE(?, SR.IDENTIFICADOR)
             AND SE.QAA <= ?
             AND SE.ESTADO = 'A'
             AND SE.DIR3 = ?
             AND SE.COD_RPA = E.COD_RPA) numeroExpedientesAbiertos,
                        (SELECT COUNT(1)
                           FROM EXPEDIENTES SE
                            LEFT OUTER JOIN RPAS RP ON SE.COD_RPA = RP.CODIGO
            LEFT OUTER JOIN INTERESADOS SS ON SE.SOLICITANTE_ID = SS.ID
            LEFT OUTER JOIN INTERESADOS SR ON SE.REPRESENTANTE_ID = SR.ID
                      WHERE SS.IDENTIFICADOR = ?
             AND SR.IDENTIFICADOR = COALESCE(?, SR.IDENTIFICADOR)
             AND SE.QAA <= ?
             AND SE.ESTADO = 'F'
             AND SE.DIR3 = ?
             AND SE.COD_RPA = E.COD_RPA) numeroExpedientesCerrados
              FROM EXPEDIENTES E
                     LEFT OUTER JOIN RPAS RP ON E.COD_RPA = RP.CODIGO
LEFT OUTER JOIN INTERESADOS S ON E.SOLICITANTE_ID = S.ID
LEFT OUTER JOIN INTERESADOS R ON E.REPRESENTANTE_ID = R.ID
             WHERE S.IDENTIFICADOR = ?
 AND R.IDENTIFICADOR = COALESCE(?, R.IDENTIFICADOR)
 AND E.QAA <= ?
 AND E.ESTADO = (CASE ? WHEN 'Total' THEN E.ESTADO WHEN 'Abierto' THEN 'A' WHEN 'Finalizado' THEN 'F' END)
```

```
AND E.DIR3 = ?
                ]]>
              </sentenciaSQL>
              <parametros>
                     <entrada>
                            <parametro nombre="docIdentidad" tipo="string" orden="1"/>
                            <parametro nombre="docIdentidadRep" tipo="string" orden="2" />
                            <parametro nombre="qaa" tipo="integer" orden="3" />
                            <parametro nombre="codigoAgrupacion" tipo="string" orden="4" />
                            <parametro nombre="docIdentidad" tipo="string" orden="5"/>
                            <parametro nombre="docIdentidadRep" tipo="string" orden="6" />
                            <parametro nombre="qaa" tipo="integer" orden="7" />
                            <parametro nombre="codigoAgrupacion" tipo="string" orden="8" />
                            <parametro nombre="docIdentidad" tipo="string" orden="9"/>
                            <parametro nombre="docIdentidadRep" tipo="string" orden="10" />
                            <parametro nombre="qaa" tipo="integer" orden="11" />
                            <parametro nombre="estado" tipo="string" orden="12" />
                            <parametro nombre="codigoAgrupacion" tipo="string" orden="13" />
                     </entrada>
                     <salida>
                            <parametro nombre="codigo" tipo="string" />
                            <parametro nombre="nombre" tipo="string" />
                            <parametro nombre="numeroExpedientesAbiertos" tipo="integer" />
                            <parametro nombre="numeroExpedientesCerrados" tipo="integer" />
                            <parametro nombre="enlace" tipo="string" />
                     </salida>
              </parametros>
        </sqlRpas>
        <sqlNumeroExpedientesNoVisibles idConsulta="SME1_AGRUP_NOVIS">
              <sentenciaSQL>
              <![CDATA[
              SELECT COUNT(1) numeroExpedientesNoVisibles
       FROM EXPEDIENTES E
LEFT OUTER JOIN INTERESADOS S ON E.SOLICITANTE_ID = S.ID
LEFT OUTER JOIN INTERESADOS R ON E.REPRESENTANTE_ID = R.ID
               WHERE S.IDENTIFICADOR = ?
 AND S.IDENTIFICADOR = COALESCE(?, S.IDENTIFICADOR)
 AND E.QAA > ?
 AND E.ESTADO = (CASE ? WHEN 'Total' THEN E.ESTADO WHEN 'Abierto' THEN 'A' WHEN 'Finalizado' THEN 'F' END)
 AND E.DIR3 = ?
```

```
]]>
                </sentenciaSQL>
                <parametros>
                        <entrada>
                                <parametro nombre="docIdentidad" tipo="string" orden="1"/>
                                <parametro nombre="docIdentidadRep" tipo="string" orden="2" />
                                <parametro nombre="qaa" tipo="integer" orden="3" />
                                <parametro nombre="estado" tipo="string" orden="4" />
                                <parametro nombre="codigoAgrupacion" tipo="string" orden="5" />

                        </entrada>
                        <salida>
                                <parametro nombre="numeroExpedientesNoVisibles" tipo="integer" />
                        </salida>
                </parametros>
        </sqlNumeroExpedientesNoVisibles>
 </consultaAgrupada>

 <consultaDetalleOrganismo>
        <sqlExpedientes idConsulta="SME1_DETORG_EXP">
                <sentenciaSQL>
                <![CDATA[
                SELECT NUM_EXPEDIENTE identificadorInterno,
                       NUM_EXP_ENI identificadorNti,
                       DENOMINACION denominacion,
                       COD_RPA codRPA,
                       DIR3 codDIR,
                       INICIA_CIUDADANO iniciaCiudadano,
                       FECHA_SOLICITUD fechaPresentacionSolicitud,
                       COD_REGISTRO registroSolicitud,
                       CSV_SOLICITUD csvSolicitud,
                       FECHA_INICIO fechaInicioExpediente,
                       FECHA_ACTUALIZACION,
                       DECODE(ESTADO, 'A', 'Abierto', 'F', 'Finalizado') estado,
                       ESTADO_DETALLADO estadoDetallado,
                       ENLACE enlaceVentanilla
       FROM EXPEDIENTES E
LEFT OUTER JOIN INTERESADOS S ON E.SOLICITANTE_ID = S.ID
LEFT OUTER JOIN INTERESADOS R ON E.REPRESENTANTE_ID = R.ID
                WHERE S.IDENTIFICADOR = ?
```

```
AND R.IDENTIFICADOR = COALESCE(?, R.IDENTIFICADOR)
AND E.QAA <= ?
AND E.ESTADO = (CASE ? WHEN 'Total' THEN E.ESTADO WHEN 'Abierto' THEN 'A' WHEN 'Finalizado' THEN 'F' END)
AND E.DIR3 = ?
AND COD_RPA IN (?)
                ]]>
              </sentenciaSQL>
              <parametros>
                     <entrada>
                            <parametro nombre="docIdentidad" tipo="string" orden="1"/>
                            <parametro nombre="docIdentidadRep" tipo="string" orden="2" />
                            <parametro nombre="qaa" tipo="integer" orden="3" />
                            <parametro nombre="estado" tipo="string" orden="4" />
                            <parametro nombre="codigoAgrupacion" tipo="string" orden="5" />
                            <parametro nombre="codigosRPAs" tipo="array" orden="6" />
                     </entrada>
                     <salida>
                            <parametro nombre="identificadorInterno" tipo="string" />
                            <parametro nombre="identificadorNti" tipo="string" />
                            <parametro nombre="denominacion" tipo="string" />
                            <parametro nombre="codRPA" tipo="string" />
                            <parametro nombre="codDIR" tipo="string" />
                            <parametro nombre="iniciaCiudadano" tipo="string" />
                            <parametro nombre="fechaPresentacionSolicitud" tipo="date" />
                            <parametro nombre="registroSolicitud" tipo="string" />
                            <parametro nombre="csvSolicitud" tipo="string" />
                            <parametro nombre="fechaInicioExpediente" tipo="date" />
                            <parametro nombre="fechaActualizacionEstadoExpediente" columna="FECHA_ACTUALIZACION" tipo="date" />
                            <parametro nombre="estado" tipo="string" />
                            <parametro nombre="estadoDetallado" tipo="string" />
                            <parametro nombre="enlaceVentanilla" tipo="string" />
                     </salida>
              </parametros>
       </sqlExpedientes>

       <sqlNumeroExpedientesNoVisibles idConsulta="SME1_DETORG_NOVIS">
              <sentenciaSQL>
              <![CDATA[
              SELECT COUNT(1) numeroExpedientesNoVisibles
      FROM EXPEDIENTES E
```

```
 LEFT OUTER JOIN INTERESADOS S ON E.SOLICITANTE_ID = S.ID
 LEFT OUTER JOIN INTERESADOS R ON E.REPRESENTANTE_ID = R.ID
                  WHERE S.IDENTIFICADOR = ?
  AND S.IDENTIFICADOR = COALESCE(?, S.IDENTIFICADOR)
  AND E.QAA > ?
  AND E.ESTADO = (CASE ? WHEN 'Total' THEN E.ESTADO WHEN 'Abierto' THEN 'A' WHEN 'Finalizado' THEN 'F' END)
  AND E.DIR3 = ?
  AND COD_RPA IN (?)
]]>
                </sentenciaSQL>
                <parametros>
                        <entrada>
                                <parametro nombre="docIdentidad" tipo="string" orden="1"/>
                                <parametro nombre="docIdentidadRep" tipo="string" orden="2" />
                                <parametro nombre="qaa" tipo="integer" orden="3" />
                                <parametro nombre="estado" tipo="string" orden="4" />
                                <parametro nombre="codigoAgrupacion" tipo="string" orden="5" />
                                <parametro nombre="codigosRPAs" tipo="array" orden="6" />
                        </entrada>
                        <salida>
                                <parametro nombre="numeroExpedientesNoVisibles" tipo="integer" />
                        </salida>
                </parametros>

        </sqlNumeroExpedientesNoVisibles>
 </consultaDetalleOrganismo>

 <consultaDetalleExpediente>
        <sqlExpediente idConsulta="SME1_DETEXP_EXP">
        <sentenciaSQL>
        <![CDATA[
        SELECT NUM_EXPEDIENTE identificadorInterno,
               NUM_EXP_ENI identificadorNti,
               DENOMINACION denominacion,
               COD_RPA codRPA,
               DIR3 codDIR,
               INICIA_CIUDADANO iniciaCiudadano,
               FECHA_SOLICITUD fechaPresentacionSolicitud,
               COD_REGISTRO registroSolicitud,
               CSV_SOLICITUD csvSolicitud,
```

```
            FECHA_INICIO fechaInicioExpediente,
            FECHA_ACTUALIZACION,
            DECODE(ESTADO, 'A', 'Abierto', 'F', 'Finalizado') estado,
            ESTADO_DETALLADO estadoDetallado,
            ENLACE enlaceVentanilla,
            NVL(PENDIENTE, 'Administracion') pendiente,
            ENLACE_PENDIENTE enlacePendiente,
            FECHA_PENDIENTE fechaPendienteInteresado,
            OBSERVACIONES observaciones
FROM EXPEDIENTES E
  WHERE E.NUM_EXPEDIENTE = ?
    ]]>
    </sentenciaSQL>
          <parametros>
            <entrada>
              <parametro nombre="idExpediente" tipo="string" orden="1" />
            </entrada>
            <salida>
              <parametro nombre="identificadorInterno" tipo="string" />
              <parametro nombre="identificadorNti" tipo="string" />
              <parametro nombre="denominacion" tipo="string" />
              <parametro nombre="codRPA" tipo="string" />
              <parametro nombre="codDIR" tipo="string" />
              <parametro nombre="iniciaCiudadano" tipo="string" />
              <parametro nombre="fechaPresentacionSolicitud" tipo="date" />
              <parametro nombre="registroSolicitud" tipo="string" />
              <parametro nombre="csvSolicitud" tipo="string" />
              <parametro nombre="fechaInicioExpediente" tipo="date" />
              <parametro nombre="fechaActualizacionEstadoExpediente" columna="FECHA_ACTUALIZACION" tipo="date" />
              <parametro nombre="estado" tipo="string" />
              <parametro nombre="estadoDetallado" tipo="string" />
              <parametro nombre="enlaceVentanilla" tipo="string" />
              <parametro nombre="pendiente" tipo="string" />
              <parametro nombre="enlacePendiente" tipo="string" />
              <parametro nombre="fechaPendienteInteresado" tipo="date" />
              <parametro nombre="observaciones" tipo="string" />
            </salida>
          </parametros>
    </sqlExpediente>
    <sqlInteresados idConsulta="SME1_DETEXP_INT">
```

```
        <sentenciaSQL>
        <![CDATA[
        SELECT I.IDENTIFICADOR docIdentidad,
               (I.NOMBRE_RAZON || ' ' || I.APELLIDO1 || ' ' || I.APELLIDO2) nombreORazon
          FROM INTERESADOS I
     INNER JOIN EXPEDIENTES E ON E.SOLICITANTE_ID = I.ID
         WHERE E.NUM_EXPEDIENTE = ?
        ]]>
        </sentenciaSQL>
        <parametros>
               <entrada>
                      <parametro nombre="idExpediente" tipo="string" orden="1" />
               </entrada>
               <salida>
                      <parametro nombre="docIdentidad" tipo="string" />
                      <parametro nombre="nombreORazon" tipo="string" />
               </salida>
        </parametros>
</sqlInteresados>
<sqlRepresentantes idConsulta="SME1_DETEXP_REP">
        <sentenciaSQL>
        <![CDATA[
        SELECT R.IDENTIFICADOR docIdentidad,
               (R.NOMBRE_RAZON || ' ' || R.APELLIDO1 || ' ' || R.APELLIDO2) nombreORazon
          FROM INTERESADOS R
     INNER JOIN EXPEDIENTES E ON E.REPRESENTANTE_ID = R.ID
         WHERE E.NUM_EXPEDIENTE = ?
        ]]>
        </sentenciaSQL>
        <parametros>
               <entrada>
                      <parametro nombre="idExpediente" tipo="string" orden="1" />
               </entrada>
               <salida>
                      <parametro nombre="docIdentidad" tipo="string" />
                      <parametro nombre="nombreORazon" tipo="string" />
               </salida>
        </parametros>
</sqlRepresentantes>
<sqlCarpetasIndice idConsulta="SME1_DETEXP_CAR">
```

```
                <sentenciaSQL>
                <![CDATA[
                SELECT D.NOMBRE, D.ORDEN, D.ID idElemento, D.PADRE_ID idPadre
                  FROM DOCUMENTOS D
INNER JOIN EXPEDIENTES E ON E.ID = D.EXPEDIENTE_ID
WHERE D.TIPO = 'CARPETA'
  AND E.NUM_EXPEDIENTE = ?
                 START WITH D.PADRE_ID IS NULL
                 CONNECT BY PRIOR D.ID = D.PADRE_ID
                ]]>
                </sentenciaSQL>
                <parametros>
                        <entrada>
                                <parametro nombre="idExpediente" tipo="string" orden="1" />
                        </entrada>
                        <salida>
                                <parametro nombre="nombre" tipo="string" />
                                <parametro nombre="orden" tipo="integer" />
                                <parametro nombre="idElemento" tipo="string" />
                                <parametro nombre="idPadre" tipo="string" />
                        </salida>
                </parametros>
        </sqlCarpetasIndice>
        <sqlDocumentosIndice idConsulta="SME1_DETEXP_DOC">
                <sentenciaSQL>
                <![CDATA[
                SELECT D.NOMBRE, D.ORDEN, D.IDENI, D.CSV, D.ID idElemento, D.PADRE_ID idPadre
                  FROM DOCUMENTOS D
INNER JOIN EXPEDIENTES E ON E.ID = D.EXPEDIENTE_ID
WHERE D.TIPO = 'DOCUMENTO'
  AND E.NUM_EXPEDIENTE = ?
                 START WITH D.PADRE_ID IS NULL
                 CONNECT BY PRIOR D.ID = D.PADRE_ID
                ]]>
                </sentenciaSQL>
                <parametros>
                        <entrada>
                                <parametro nombre="idExpediente" tipo="string" orden="1" />
                        </entrada>
                        <salida>
```

```
                                    <parametro nombre="orden" tipo="integer" />
                                    <parametro nombre="idEni" tipo="string" />
                                    <parametro nombre="csv" tipo="string" />
                                    <parametro nombre="nombre" tipo="string" />
                                    <parametro nombre="idElemento" tipo="string" />
                                    <parametro nombre="idPadre" tipo="string" />
                                </salida>
                                </parametros>
                            </sqlDocumentosIndice>
                        </consultaDetalleExpediente>
                    </servicioExpediente>
```

5 NOTAS AL DESARROLLO DE SERVICIOS

A continuación se detallará para cada servicio los parámetros de entrada y de salida que serán necesarios configurar.

5.1 Servicio tipo 'Mi información' ad hoc

Para un servicio de este tipo en el toolkit hay que tener en cuenta lo siguiente:

- Parámetros de entrada soportados:
  - identificador: Documento de identidad del titular.
  - nombreRazonSocial: Nombre o razón social del titular.
  - apellido1: Primer apellido del titular.
- Parámetros de salida: Los que determine el organismo.

5.2 Servicio tipo 'Mi información' SCSP

Para un servicio de este tipo en el toolkit hay que tener en cuenta lo siguiente:

1. Identificador de servicio: se especificará como identificador el código de certificado SCSP. Dicho identificador será proporcionado por la plataforma SCSP durante el proceso de alta.

- Parámetros de entrada soportados.

  Datos del titular (dicha información identificativa será suministrada por el proveedor de identidad usado por Carpeta Ciudadana, en este caso Proxy Cl@ve):
  - tipoDocumentacion: Tipo de documento de identidad del titular.
  - documentacion: Documento de identidad del titular.
  - nombreCompleto (opcional): Nombre completo del titular.
  - nombre (opcional): Nombre del titular.
  - apellido1 (opcional): Primer apellido del titular.
  - apellido2 (opcional): Segundo apellido del titular.

Otros datos suministrados en la petición SCSP.

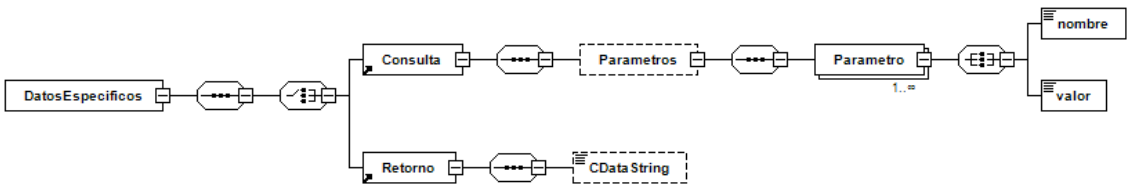
Emisor:

- nifEmisor: NIF del emisor del certificado.
- nombreEmisor: Nombre del emisor del certificado.

Solicitante:

- identificadorSolicitante: NIF del organismo solicitante.
- nombreSolicitante: Nombre del organismo al que pertenece el solicitante de los datos.
- unidadTramitadora (opcional): Nombre de la unidad encargada de la tramitación y destinatario de la información.
- codProcedimiento (opcional): Código RPA del procedimiento para el que se autoriza la consulta.
- nombreProcedimiento (opcional): Nombre del procedimiento para el que se autoriza la consulta.
- finalidad: Contiene el motivo o causa por la que se necesita realizar la consulta.
- consentimiento: Los valores posibles son: "Si", "Ley".
- nombreCompletoFuncionario (opcional): Nombre completo del empleado público que realiza la solicitud.
- nifFuncionario (opcional): NIF del empleado público que realiza la solicitud.
- idExpediente (opcional): identificador del expediente del organismo requirente/gestor que necesita la consulta.

- Parámetros de específicos de entrada: Un servicio SCSP puede definir un conjunto de parámetros de entrada específicos del servicio. Para ello en el elemento 'DatosEspecificos' de la petición se podrá definir un elemento 'Consulta', el cual incluirá un elemento 'Parametros' con el conjunto de parámetros definidos por cada uno por un nombre de parámetro y un valor.



- Parámetros de salida: Los que determine el organismo.

5.3 Servicio tipo 'Mis expedientes'

Para un servicio de este tipo en el toolkit hay que tener en cuenta lo siguiente:

- No es necesario definir los elementos que permiten especificar el formato de la respuesta del servicio, es suficiente con la configuración de las sentencias SQL y parámetros de entrada/salida de cada consulta del servicio.
- Identificador de servicio: se especificará el nombre de sistema indicado en la solicitud de integración con Carpeta Ciudadana.
- consultaAgrupada:
    - Parámetros de entrada soportados:
        - docIdentidad: Documento de identidad del interesado.
        - apellido1: Primer apellido del interesado o razón social.
        - docIdentidadRep (opcional): Documento de identidad del representante.
        - nombreRep (opcional): Nombre del representante.
        - tipoCertificado (opcional): Tipo de certificado con el que se ha autenticado el ciudadano.
        - qaa: Nivel de autenticación con el que se accede a la consulta de expedientes.
        - estado: Parámetro que indica si se devuelven todos los expedientes o solamente los que se encuentren en un estado determinado.
        - tipoAgrupacion: Indica qué tipo de operación se está realizando, si se está realizando la consulta por totales o por código RPA.
        - codigoAgrupacion: Los expedientes de Carpeta Ciudadana se mostrarán organizados por codigoAgrupacion. En principio corresponde con el DIR3 del organismo.
        - origen: Indica el origen de la petición.
    - Parámetros de salida:
        - sqlTotales
            - numeroExpedientes: Número de expedientes del ciudadano.
            - hayMasResultados: Verdadero si existen más resultados.
            - enlace: Enlace.
        - sqlRpas
            - codigo: Código del procedimiento en RPA.

- nombre (opcional): Nombre RPA del procedimiento.
- numeroExpedientesAbiertos (opcional): Número de expedientes abiertos.
- numeroExpedientesCerrados (opcional): Número de expedientes finalizados.
- enlace: Enlace.

▪ sqlNumeroExpedientesNoVisibles

- numeroExpedientesNoVisibles: Número de expedientes que no se pueden visualizar con el QAA indicado.

- consultaDetalleOrganismo:
  - Parámetros de entrada soportados:
    - docIdentidad: Documento de identidad del interesado.
    - apellido1: Primer apellido del interesado o razón social.
    - docIdentidadRep (opcional): Documento de identidad del representante.
    - nombreRep (opcional): Nombre del representante.
    - tipoCertificado (opcional): Tipo de certificado con el que se ha autenticado el ciudadano.
    - qaa: Nivel de autenticación con el que se accede a la consulta de expedientes.
    - estado: Parámetro que indica si se devuelven todos los expedientes o solamente los que se encuentren en un estado determinado.
    - codigoAgrupacion: Los expedientes de Carpeta Ciudadana se mostrarán organizados por codigoAgrupacion. En principio corresponde con el DIR3 del organismo.
    - codigosRPAs: Listado de códigos RPA por el que se quiere realizar la consulta.
    - origen: Indica el origen de la petición.
  - Parámetros de salida:
    - sqlExpedientes
      - identificadorInterno: Identificador de expediente (codificación propia en el órgano que lo tramita).
      - identificadorNti (opcional): Identificador normalizado del expediente.
      - denominacion: Denominación del expediente.
      - codRPA: Código del procedimiento en el RPA.

- codDIR: Código DIR del órgano que tramita el expediente.
- iniciaCiudadano: Expediente iniciado por solicitud del ciudadano (S/N).
- fechaPresentacionSolicitud (opcional): Fecha de presentación de la solicitud a partir de la cual se inicia el expediente.
- registroSolicitud (opcional): Número de registro del documento de presentación de la solicitud.
- csvSolicitud (opcional): Código seguro de verificación en el HCV.
- fechaInicioExpediente: Fecha de inicio del expediente.
- fechaActualizacionEstadoExpediente (opcional): Fecha del último cambio de estado del expediente.
- estado: Estado del expediente. Puede ser abierto o finalizado.
- estadoDetallado: Estado detallado del expediente.
- enlaceVentanilla (opcional): Enlace de consulta para mayor detalle en la sede electrónica / oficina virtual correspondiente.

▪ sqlNumeroExpedientesNoVisibles

- numeroExpedientesNoVisibles: Número de expedientes que no se pueden visualizar con el QAA indicado.

• consultaDetalleExpediente:

○ Parámetros de entrada soportados:

▪ docIdentidad: Documento de identidad del interesado.
▪ apellido1: Primer apellido del interesado o razón social.
▪ docIdentidadRep (opcional): Documento de identidad del representante.
▪ nombreRep (opcional): Nombre del representante.
▪ tipoCertificado (opcional): Tipo de certificado con el que se ha autenticado el ciudadano.
▪ qaa: Nivel de autenticación con el que se accede a la consulta de expedientes.
▪ idExpediente: Identificador del expediente solicitado.
▪ codigoAgrupacion: Los expedientes de Carpeta Ciudadana se mostrarán organizados por codigoAgrupacion. En principio corresponde con el DIR3 del organismo.
▪ origen: Indica el origen de la petición.

- Parámetros de salida:
  - sqlExpediente
    - identificadorInterno: Identificador de expediente (codificación propia en el órgano que lo tramita).
    - identificadorNti (opcional): Identificador normalizado del expediente.
    - denominacion: Denominación del expediente.
    - codRPA: Código del procedimiento en el RPA.
    - codDIR: Código DIR del órgano que tramita el expediente.
    - iniciaCiudadano: Expediente iniciado por solicitud del ciudadano (S/N).
    - fechaPresentacionSolicitud (opcional): Fecha de presentación de la solicitud a partir de la cual se inicia el expediente.
    - registroSolicitud (opcional): Número de registro del documento de presentación de la solicitud.
    - csvSolicitud (opcional): Código seguro de verificación en el HCV.
    - fechaInicioExpediente: Fecha de inicio del expediente.
    - fechaActualizacionEstadoExpediente (opcional): Fecha del último cambio de estado del expediente.
    - estado: Estado del expediente. Puede ser abierto o finalizado.
    - estadoDetallado: Estado detallado del expediente.
    - enlaceVentanilla (opcional): Enlace de consulta para mayor detalle en la sede electrónica / oficina virtual correspondiente.
    - pendiente (opcional): Indica si el expediente se encuentra pendiente de realizar alguna acción por parte de la administración o de la persona interesada.
    - enlacePendiente (opcional): Si el expediente se encuentra pendiente de acción por parte de la persona interesada, dirección en la cual puede realizar dicha acción o está disponible más información sobre la misma.
    - fechaPendienteInteresado (opcional): Fecha límite para efectuar la acción pendiente por parte del interesado. Este campo aplica si el expediente está pendiente del interesado.
    - observaciones (opcional): Observaciones del expediente.

- sqlInteresados
  - docIdentidad: Documento de identidad del interesado.
  - nombreORazon: Nombre o razón del interesado.
- sqlRepresentantes
  - docIdentidad: Documento de identidad del representante.
  - nombreORazon: Nombre o razón del representante.
- sqlCarpetasIndice
  - nombre: Nombre de la carpeta.
  - orden (opcional): Secuencial que marca el orden de visualización de la carpeta.
  - idElemento: Identificador de carpeta. Permite a otras carpetas y documentos vincular esta carpeta como elemento padre en la estructura del índice.
  - idPadre (opcional): Identificador de carpeta padre. Debe coincidir con el idElemento de la carpeta que la contiene.
- sqlDocumentosIndice
  - orden (opcional): Secuencial que marca el orden de visualización del documento dentro de su carpeta.
  - idEni (opcional): Identificador normalizado ENI del documento.
  - csv (opcional): Código seguro de verificación del documento en la HCV.
  - nombre: Título descriptivo del documento.
  - idPadre (opcional): Identificador de carpeta padre. Debe coincidir con el idElemento de la carpeta que la contiene.

6 CONFIGURACIÓN DEL SERVICIO

La librería *toolkit-core* ofrece la funcionalidad básica para facilitar la integración con los servicios requeridos por Carpeta Ciudadana, pero no se encarga de la publicación final de los servicios web. De esta forma el componente es independiente de la tecnología de Web Service que desee implementar cada organismo.

Como complemento al *toolkit-core*, se ofrecen a modo de referencia los componentes *consultamiadhoc*, *consultamiscsp* y *consultame*. Estos componentes, que hacen uso de la librería *toolkit-core*, son ficheros desplegables *aar* que han sido implementados usando la tecnología Axis 2. Mediante su correcto despliegue son los encargados de publicar los Web Services accesibles posteriormente para la comunicación con Carpeta Ciudadana.

6.1 Despliegue de servicios usando Axis 2

A continuación se detallará el proceso para publicar un servicio web mediante Axis 2, haciendo uso de los componentes *aar* suministrados.

Para este ejemplo se empleará la versión de Axis2 1.7.7.

1. Será necesario obtener el empaquetado war de Axis 2 ofrecido por Apache: http://www.apache.org/dyn/closer.lua/axis/axis2/java/core/1.7.7/axis2-1.7.7-war.zip

   Este zip contiene un fichero war con la estructura de la aplicación a desplegar en el servidor de aplicaciones.

```
axis2-web
META-INF
WEB-INF
   classes
   conf
      axis2.xml
   lib
      activation.jar
      ...
      xmlSchema.jar
   modules
      modules.list
      addressing.mar
      ...
      soapmonitor.mar
   services
      services.list
      aservice.aar
      ...
      version.aar
   web.xml
```

2. En esta estructura será necesario realizar una serie de modificaciones. Una de ellas será incluir el módulo encargado de implementar la seguridad de las comunicaciones SOAP mediante uso de autenticación WS-Security con UserNametoken. El módulo que emplear será Rampart en su versión 1.7.1. Para ello será necesario descargarlo desde la web de Apache:

http://www.apache.org/dyn/closer.lua/axis/axis2/java/rampart/1.7.1/rampart-dist-1.7.1-bin.zip

3. En el despliegue de Axis 2 se deberán copiar una serie de ficheros de distribución binaria de Rampart. Los ficheros a copiar son los modulos *mar* (rampart-1.7.1.mar y rahas-1.7.1.mar) ubicados en *rampart-1.7.1/modules.* Deberán ser ubicados dentro de la carpeta <despliegue-axis2>/WEB-INF/modules

   También es necesario copiar todas las librerías ubicadas en *rampart-1.7.1/lib* en la carpeta <despliegue-axis2>/WEB-INF/lib

4. Despliegue de los servicios relativos al toolkit. Como todos los *aar* suministrados son dependientes de la librería toolkit-core, previamente se va a incorporar dicho jar como una librería común de Axis 2. Para ello habrá que copiar el fichero *toolkit-core-x.x.x.jar* en la carpeta <despliegue-axis2>/WEB-INF/lib.

   Dicho fichero jar deberá haber sido configurado previamente según las necesidades del organismo (Ver sección 7. Configuración de la librería).

5. Será necesario que bien, en el servidor de aplicaciones, o en la aplicación axis 2, se incluya la librería que se desea usar para las comunicaciones con base de datos. Por ejemplo, si se está usando una base de datos Oracle 11g, se podría incluir la librería ojdbc6.jar en la carpeta <despliegue-axis2>/WEB-INF/lib.

6. A continuación se procederá con el despliegue de los servicios web del toolkit. Para ello será necesario copiar los ficheros *consultamiadhoc.aar, consultamiscsp.aar* y *consultame.aar* en <despliegue-axis2>/WEB-INF/services.

   Hay que tener en cuenta que las configuraciones de seguridad en las comunicaciones SOAP están definidas en los ficheros "aar". Dichas configuraciones se detallan en el apartado "*6.1.1. Configuración cabeceras de seguridad*".

7. Por último, para poder gestionar la creación y destrucción de los pools de conexiones a base de datos, la librería toolkit-core ofrece una clase (*GestorPoolListener*) que debe ser definida como listener de la aplicación web donde se integre. Como en este caso la integración es en Axis 2, se deberá definir en el fichero <despliegue-axis2>/WEB-INF/web.xml.

   Para ello bastaría con incorporar en el fichero el siguiente código:

```
<listener>
        <description>Listener para la gestion del pool de conexiones.</description>
         <display-name>GestorPoolListener</display-name>
        <listener-class>es.juntadeandalucia.cc.toolkit.sql.GestorPoolListener</listener-class>
 </listener>
```

8. En instalaciones del toolkit que definan múltiples servicios de tipo 'Mi expedientes' (instalación multisistema) es necesario que el servicio web correspondiente sea capaz de procesar la

cabecera SOAP de la petición de Carpeta Ciudadana. Esta cabecera incluye el parámetro 'codigoSistema' el cual se requiere para identificar el servicio a invocar. Para poder tratar esta información de la cabecera SOAP será necesario definir un handler en Axis 2. Para ello se requieren dos actuaciones.

Primero, se incorporará el fichero suministrado *toolkit-ws-handler-axis.jar* en la carpeta <despliegue-axis2>/WEB-INF/lib.

Y segundo, se modificará el fichero <despliegue-axis2>/WEB-INF/conf/axis2.xml de la siguiente forma:

Se identificará el bloque: "`<phaseOrder type="InFlow">`"

En dicho bloque, como último elemento se incorporará el siguiente código:

```
<phase name="HeaderPhase">
   <handler name="HeaderHandler"
            class="es.juntadeandalucia.cc.toolkit.me.CCHeaderHandler">
       <order phase="HeaderPhase"/>
   </handler>
 </phase>
```

9. Para el caso en el que se quieran habilitar los servicios web vía https, será necesario, además de realizar las configuraciones pertinentes al servidor de aplicaciones, modificar el fichero <despliegue-axis2>/WEB-INF/conf/axis2.xml. Por defecto tiene definido un elemento *<transportReceiver name="http"...>*. Para poder publicar por https será necesario añadir un nuevo *transportReceiver* o bien sustituirlo, según se quiera tener publicado el servicio por ambos protocolos o sólo por https. Para el caso en el que se quieran publicar ambos, será necesario obligatoriamente indicar los puertos a usar por cada protocolo:

```
<transportReceiver name="http"
class="org.apache.axis2.transport.http.AxisServletListener">
        <parameter name="port">8080</parameter>
    </transportReceiver>
    <transportReceiver name="https"
class="org.apache.axis2.transport.http.AxisServletListener">
        <parameter name="port">8443</parameter>
    </transportReceiver>
```

10. Una vez realizados todos los pasos, bastaría con desplegar la estructura actualizada de la aplicación web Axis2 en el servidor de aplicaciones correspondiente. Una vez arrancado, se podrá confirmar su correcto despliegue comprobando la publicación de los WSDL de cada servicio. Para ello, desde una navegador se accederá a los distintos endpoints publicados incorporando al final la cadena "?wsdl". Las rutas serán del tipo:

    http://<ruta_publicada>/axis2/services/consultamiadhoc?wsdl

    http://<ruta_publicada>/axis2/services/ConsultaMIService?wsdl

    http://<ruta_publicada>/axis2/services/ConsultaCCService?wsdl

6.1.1 Configuración cabeceras de seguridad

Dependiendo del tipo de servicio se implementarán distintos tipos de cabeceras de seguridad, cumpliendo el protocolo WS-Security.

- Para los servicios de información de tipo ad-hoc y para los servicios de expediente se hará uso de autenticación mediante **UsernameToken**.
- Para los servicios de información tipo SCSP se hará uso de autenticación mediante **BinarySecurityToken**.

Configuración de UsernameToken

Esta configuración aplica para los servicios de información de tipo ad-hoc (consultamiadhoc.aar) y para los servicios de expediente (consultame.aar).

La configuración de los pares usuario/clave válidos para la autenticación con UserNametoken de cada servicio se realiza mediante el fichero *configuracion.properties* ubicado en el raiz de los ficheros aar. Por ello, para cambiar los valores por defecto será necesario editar dicho fichero de properties en cada empaquetado aar.

Deberán ser modificados los siguientes valores:

- *ws.security.user.name*: Se indicará el nombre de usuario usado en la autenticación.
- *ws.security.user.password*: Se indicará la clave usada en la autenticación.

Configuración de BinarySecurityToken

Esta configuración aplica para los servicios de información tipo SCSP (consultamiscsp.aar).

Para esta configuración será necesario que el organismo disponga de:

- Un certificado digital encargado de realizar la firma (mediante su clave privada) de las respuestas SOAP generadas por el servicio.
- Que haya recibido la parte pública del certificado con el que se validarán las peticiones al servicio de información tipo SCSP.

Ambos certificados (el usado para la firma y la parte pública para validar las peticiones SCSP) deberán incluirse en un único almacén de certificados (keystore) que deberá estar ubicado en una ruta de disco accesible por el servidor de aplicaciones donde se despliegue el servicio.

A continuación deberá realizar las siguientes configuraciones en el empaquetado consultamiscsp.aar:

- Fichero *META-INF/services.xml*:

En dicho fichero será necesario configurar el siguiente fragmento:

```
<ramp:RampartConfig xmlns:ramp="http://ws.apache.org/rampart/policy">
                                    <ramp:user>ALIAS_CERT_FIRMA_RESPUESTA</ramp:user>

<ramp:passwordCallbackClass>es.juntadeandalucia.cc.toolkit.mi.util.PWCBHandler</ramp:passwordCal
lbackClass>
                                    <ramp:signatureCrypto>
                                            <ramp:crypto
provider="org.apache.ws.security.components.crypto.Merlin">
                                                    <ramp:property

        name="org.apache.ws.security.crypto.merlin.keystore.type">JKS</ramp:property>
                                                    <ramp:property
name="org.apache.ws.security.crypto.merlin.file">/ruta_keystore/keystore.jks</ramp:property>
                                                    <ramp:property

name="org.apache.ws.security.crypto.merlin.keystore.password">7J32qhPSy61R</ramp:property>
                                            </ramp:crypto>
                                    </ramp:signatureCrypto>
                            </ramp:RampartConfig>
```

Deberán ser modificados los siguientes valores:

- *<ramp:user>:* Se indicará el alias del certificado usado para firmar las respuestas del servicio.
- <ramp:property name="org.apache.ws.security.crypto.merlin.keystore.type">: Se indicará el tipo de keystore usado: JKS o PKCS12.
- <ramp:property name="org.apache.ws.security.crypto.merlin.file">: Se indicará la ruta al keystore usado.
- <ramp:property name="org.apache.ws.security.crypto.merlin.keystore.password">: Se indicará la clave del keystore.

- Fichero *configuracion.properties*:

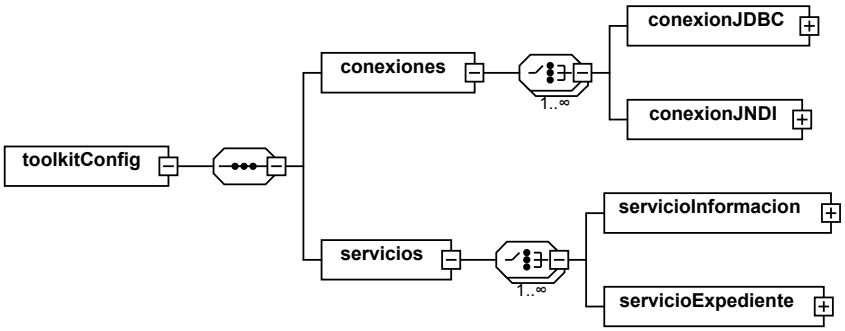
  Deberán ser modificados los siguientes valores:

  - *ws.security.keystore.cert.alias*: Se indicará el alias del certificado usado para firmar las respuestas del servicio.
  - *ws.security.keystore.cert.password*: Se indicará la clave de la parte privada del certificado usado para firmar las respuestas del servicio.

Adicionalmente a estas configuraciones, para evitar problemas de cifrado es necesario asegurarse que la JDK con la que se ejecute el servidor de aplicaciones tenga instalada la extensión ofrecida por Oracle: Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files. Si no es así, a través de la web de Oracle se puede obtener la versión de la extensión acorde a la JDK usada, junto a las instrucciones de su instalación.

7 CONFIGURACIÓN DE LA LIBRERÍA

7.1 Definición de servicios

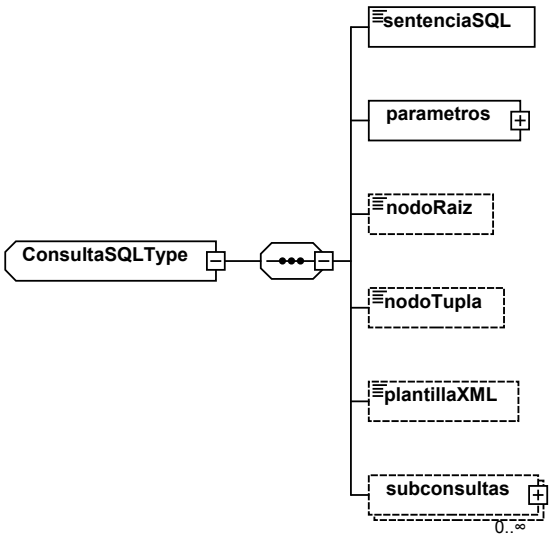


El fichero de configuración del toolkit (configuracion-toolkit.xm) permite definir las conexiones a base de datos a utilizar por los distintos servicios y la configuración específica de cada tipo de servicio.

Cada elemento del fichero de configuración será descrito en la sección correspondiente de este documento.

7.2 Consultas de datos

Todo servicio del toolkit, con independencia de la sección de Carpeta Ciudadana con la que se integre, define una o varias consultas como fuente de datos del servicio.

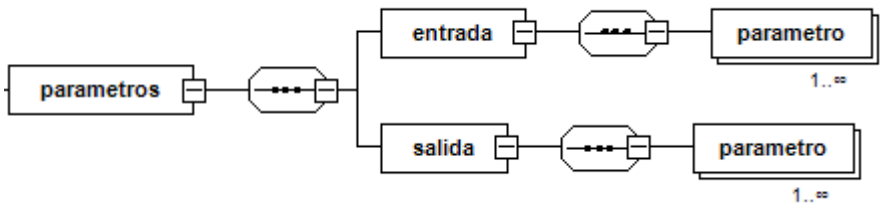


ConsultaSQLType

Nombre	ConsultaSQLType
Tipo	Bloque
Atributos	**idConsulta**: Identificador de consulta. **idConexion**: Identificador de conexión de base de datos. **esSubconsulta**: Indica si la consulta se debe evaluar como subconsulta de otra.
Obligatorio	Sí
Descripción	Configuración de consulta.

Nombre	sentenciaSQL
Tipo	Cadena
Obligatorio	Sí
Descripción	Sentencia SQL a ejecutar para la consulta.

Nombre	parametros
Tipo	Bloque
Obligatorio	Sí
Descripción	Relación de parámetros de entrada y salida de la consulta.



Nombre	parametros/entrada
Tipo	Bloque
Obligatorio	Sí
Descripción	Relación de parámetros de entrada de la consulta.

Nombre	parametros/entrada/parametro
Tipo	ParametroEntradaType
Atributos	**orden**: Orden en la consulta parametrizada. **tipo**: Tipo de dato del parámetro. Valores permitidos: "integer", "string", "date", "double", "float", "array". **subtipo**: Para tipo de dato "array" permite indicar el tipo de sus elementos. Valores permitidos: "integer", "string". **nombre**: Nombre de la variable que se utilizará en la consulta. Se pueden emplear variables de entrada del servicio web y valores generados en otras consultas.
Obligatorio	Si
Descripción	Parámetro de entrada de la consulta. Los parámetros definidos con tipo "array" se deberán corresponder en SQL con una cláusula de tipo "IN (?)". Los parámetros definidos como tipo "date" se deberán indicar en formato ISO 8601 (AAAA-MM-DDTHH:MM:SS). **Nota:** En bases de datos Oracle, si la columna por la que se va a filtrar es de tipo CHAR, se deberá realizar en la consulta SQL una conversión de formato antes de la comparación. Por ejemplo, en la siguiente consulta si la columna identificador está definida como CHAR(10): SELECT * FROM PERSONAS WHERE IDENTIFICADOR = ? deberá escribirse como SELECT * FROM PERSONAS WHERE IDENTIFICADOR = CAST(? AS CHAR(10))

Nombre	parametros/salida
Tipo	Bloque
Obligatorio	Sí
Descripción	Relación de parámetros de salida de la consulta.

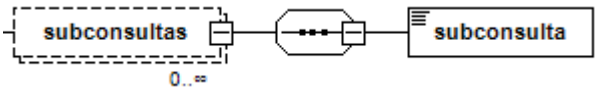
Nombre	parametros/salida/parametro
Tipo	ParametroSalidaType
Atributos	**nombre:** Nombre de variable en la que se almacenará la respuesta. **columna**: Nombre de la columna en la consulta de la que se extraerá el dato. Si no se indica este atributo se tomará como nombre de la columna el valor indicado en el parámetro "nombre". **tipo**: Tipo de dato del parámetro. Valores permitidos: "blob", "clob", "date", "double", "float", "integer", "string", "binary".
Obligatorio	Si
Descripción	Parámetro de entrada de la consulta. Los parámetros definidos como tipo "date" se deberán indicar en formato ISO 8601 (AAAA-MM-DDTHH:MM:SS).

Nombre	nodoRaiz
Tipo	Cadena
Obligatorio	No
Descripción	Nombre de elemento raíz que agrupa los resultados de la consulta.

Nombre	nodoTupla
Tipo	Cadena
Obligatorio	No
Descripción	Nombre de elemento que agrupa cada tupla devuelta por la consulta.

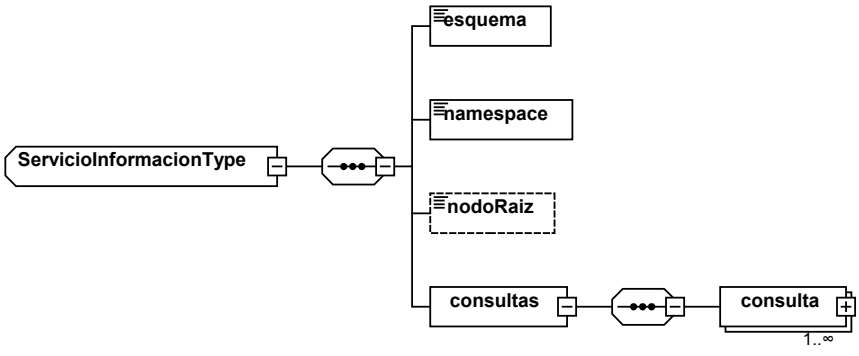
Nombre	plantillaXML
Tipo	Cadena
Obligatorio	No
Descripción	Plantilla xml que se utilizará para la carga de los datos de la consulta SQL. A definir en consultas tipo 'Mi información'.

Nombre	subconsultas
Tipo	Bloque
Obligatorio	No
Descripción	Relación de subconsultas a ejecutar con esta consulta.



Nombre	subconsultas/subconsulta
Tipo	Cadena
Obligatorio	Sí
Descripción	Se indica el identificador de la subconsulta. Una subconsulta podrá hacer referencia a parámetros de salida de la consulta padre.

7.3 Mi información



ServicioInformacionType

Nombre	ServicioInformacionType
Tipo	Bloque
Atributos	**idServicio**: Identificador de servicio. **idConexion**: Identificador de la conexión de base de datos a utilizar en las consultas del servicio. **tipo**: Tipo de servicio. Valores permitidos: 'adhoc', 'scsp'.
Obligatorio	Sí
Descripción	Configuración de servicio tipo 'Mi información'.

Nombre	esquema
Tipo	Cadena
Obligatorio	Sí
Descripción	Ruta del esquema XSD empleado en validación de la respuesta de la consulta.

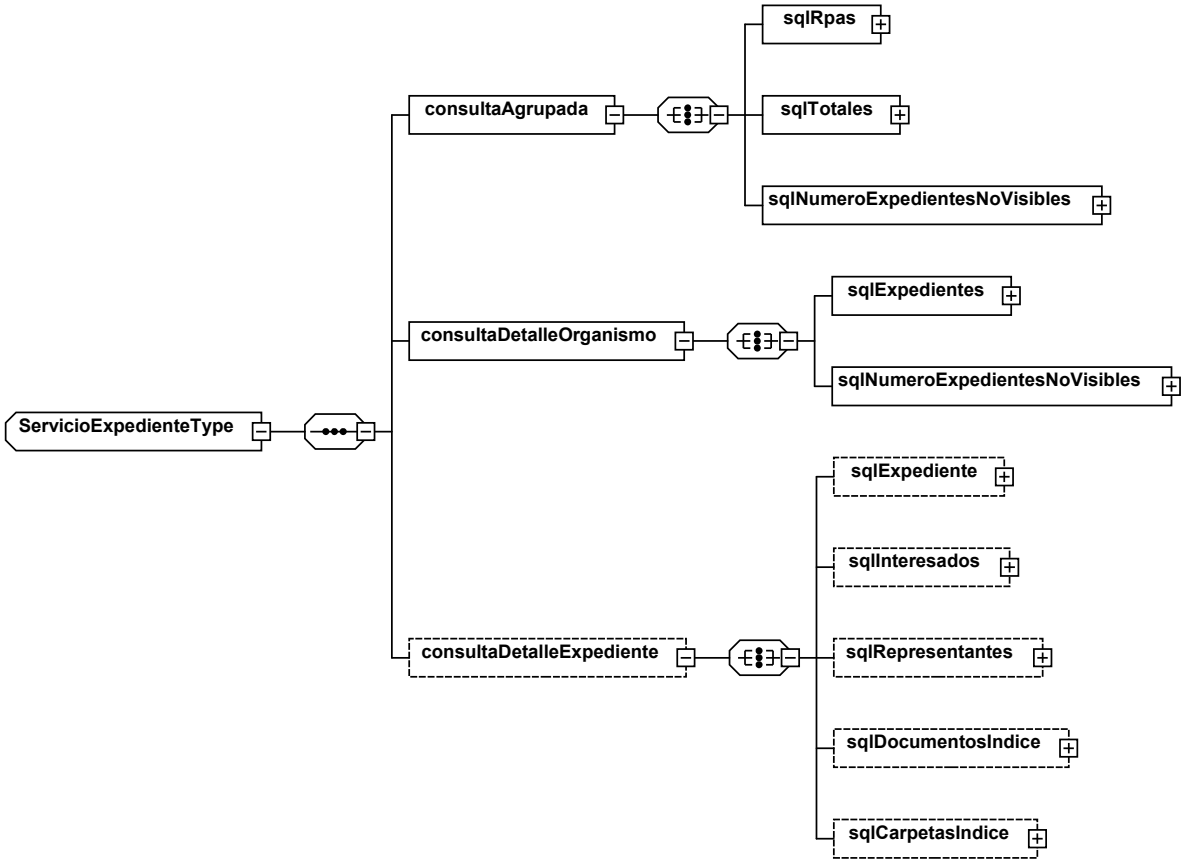
Nombre	namespace
Tipo	Cadena
Obligatorio	Sí
Descripción	Namespace de los elementos XML de la respuesta. Deberá seguir el siguiente patrón: "urn:juntadeandalucia:[siglas_organismo]:[versión_contrato_de_servicio]:servicios:ciudadania:[nombre_servicio]Response" Un ejemplo sería: "urn:juntadeandalucia:chap:v1.0:servicios:ciudadania:carnetBibliotecaResponse"

Nombre	nodoRaiz
Tipo	Cadena
Obligatorio	No
Descripción	Nombre de elemento raíz que agrupa los resultados del servicio.

Nombre	consultas
Tipo	Bloque
Obligatorio	Sí
Descripción	Configuración de consultas para sección Mi información.

Nombre	consultas/consulta
Tipo	ConsultaSQLType
Obligatorio	Sí
Descripción	Consulta que servirá de fuente de datos.

7.4 Mis expedientes



ServicioExpedienteType

Nombre	ServicioExpedienteType
Tipo	Bloque
Atributos	**idServicio**: Identificador de servicio. **idConexion**: Identificador de la conexión de base de datos a utilizar en las consultas del servicio.
Obligatorio	Sí
Descripción	Configuración de servicio tipo 'Mis expedientes'.

Nombre	consultaAgrupada
Tipo	Bloque
Obligatorio	Sí
Descripción	Configuración de consultas necesarias para generar la respuesta del servicio 'consultaAgrupada'.

Nombre	consultaAgrupada/sqlRpas
Tipo	ConsultaSQLType
Obligatorio	Sí
Descripción	Consulta para obtención de listado de información referente a los códigos RPA de expedientes.

Nombre	consultaAgrupada/sqlTotales
Tipo	ConsultaSQLType
Obligatorio	Sí
Descripción	Consulta para obtención de información sobre el conjunto de expedientes del ciudadano.

Nombre	consultaAgrupada/sqlNumeroExpedientesNoVisibles
Tipo	ConsultaSQLType
Obligatorio	Sí
Descripción	Consulta para obtención de número de expedientes que no se pueden visualizar con el QAA indicado.

Nombre	consultaDetalleOrganismo
Tipo	Bloque
Obligatorio	Sí
Descripción	Configuración de consultas necesarias para generar la respuesta del servicio 'consultaDetalleOrganismo'.

Nombre	consultaDetalleOrganismo/sqlExpedientes
Tipo	ConsultaSQLType
Obligatorio	Sí
Descripción	Consulta para obtención de lista de expedientes básicos.

Nombre	consultaDetalleOrganismo/sqlNumeroExpedientesNoVisibles
Tipo	ConsultaSQLType
Obligatorio	Sí
Descripción	Consulta para obtención de número de expedientes que no se pueden visualizar con el QAA indicado.

Nombre	consultaDetalleExpediente
Tipo	Bloque
Obligatorio	No
Descripción	Configuración de consultas necesarias para generar la respuesta del servicio 'consultaDetalleExpediente'.

Nombre	consultaDetalleExpediente/sqlExpediente
Tipo	ConsultaSQLType
Obligatorio	No
Descripción	Consulta para obtención de expediente solicitado.

Nombre	consultaDetalleExpediente/sqlInteresados
Tipo	ConsultaSQLType
Obligatorio	No
Descripción	Consulta para obtención de datos identificativos de los interesados asociados al expediente.

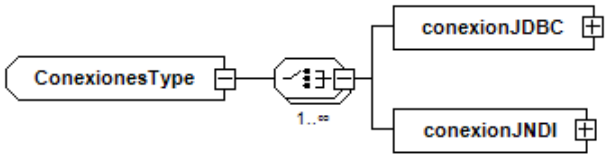
Nombre	consultaDetalleExpediente/sqlRepresentantes
Tipo	ConsultaSQLType
Obligatorio	No
Descripción	Consulta para obtención de datos identificativos de los representantes asociados al expediente.

Nombre	consultaDetalleExpediente/sqlDocumentosIndice
Tipo	ConsultaSQLType
Obligatorio	No
Descripción	Consulta para obtención de índice del expediente.

Nombre	consultaDetalleExpediente/sqlCarpetasIndice
Tipo	ConsultaSQLType
Obligatorio	No
Descripción	Consulta para obtención de carpetas del expediente.

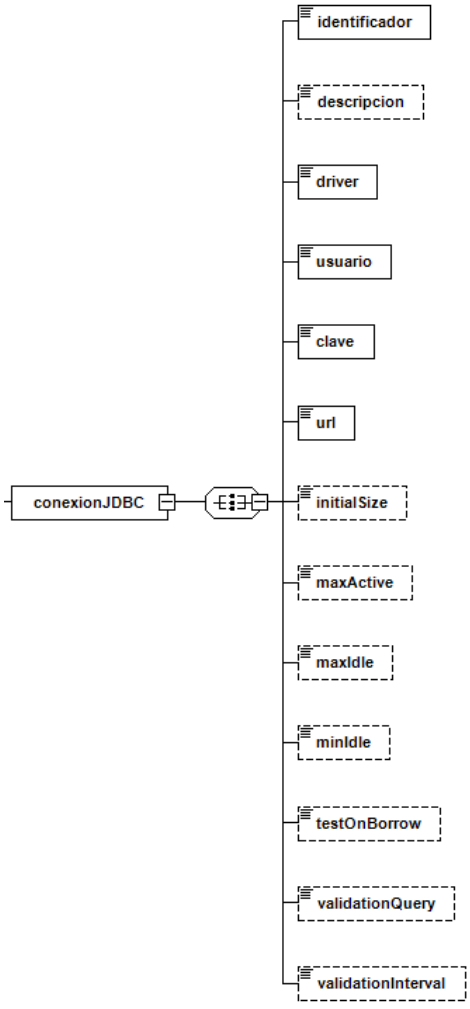
7.5 Conexiones a bases de datos

Las conexiones a base de datos se podrán definir de dos modos: mediante conexión JDBC o bien haciendo uso de una conexión definida mediante JNDI en el servidor de aplicaciones.



ConexionesType

Nombre	conexionJDBC
Tipo	Bloque
Obligatorio	Obligatorio 'conexionJDBC' o bien 'conexionJNDI'
Descripción	Configuración de conexión a base de datos mediante JDBC.



Nombre	conexionJDBC/identificador
Tipo	Cadena
Obligatorio	Sí
Descripción	Identificador de la conexión.

Nombre	conexionJDBC/descripcion
Tipo	Cadena
Obligatorio	No
Descripción	Descripción de la conexión.

Nombre	conexionJDBC/driver
Tipo	Cadena
Obligatorio	Sí
Descripción	Nombre cualificado la clase usada como driver de conexión a la base de datos.

Nombre	conexionJDBC/usuario
Tipo	Cadena
Obligatorio	Sí
Descripción	Usuario para la conexión a base de datos.

Nombre	conexionJDBC/clave
Tipo	Cadena
Obligatorio	Sí
Descripción	Clave para la conexión a base de datos.

Nombre	conexionJDBC/url
Tipo	Cadena
Obligatorio	Sí
Descripción	Url de conexión a base de datos.

Nombre	conexionJDBC/initialSize
Tipo	Entero
Obligatorio	No
Descripción	Número inicial de conexiones que se crean al inicializar el pool.

Nombre	conexionJDBC/maxActive
Tipo	Entero
Obligatorio	No
Descripción	Número máximo de conexiones activas que el pool puede mantener simultáneamente.

Nombre	conexionJDBC/maxIdle
Tipo	Entero
Obligatorio	No
Descripción	Número máximo de conexiones inactivas que permanecerán abiertas.

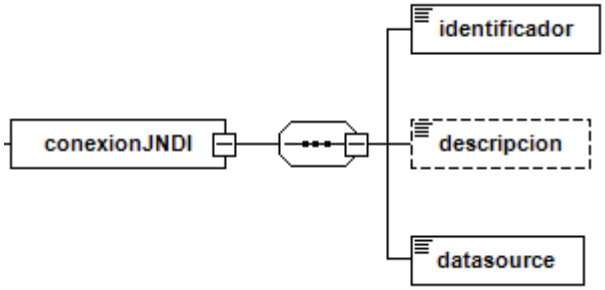
Nombre	conexionJDBC/minIdle
Tipo	Entero
Obligatorio	No
Descripción	Número mínimo de conexiones que permanecerán activas en el pool.

Nombre	conexionJDBC/testOnBorrow
Tipo	Booleano
Obligatorio	No
Descripción	Indica si se comprobarán las conexiones antes de ser sacadas del pool. Posibles valores: "true" o "false".

Nombre	conexionJDBC/validationQuery
Tipo	Cadena
Obligatorio	No
Descripción	Consulta de validación a emplear.

Nombre	conexionJDBC/validationInterval
Tipo	Long
Obligatorio	No
Descripción	Evita validaciones excesivas. Número de milisegundos mínimos que deben pasar desde la anterior validación.

Nombre	conexionJNDI
Tipo	Bloque
Obligatorio	Obligatorio 'conexionJDBC' o bien 'conexionJNDI'
Descripción	Configuración de conexión a base de datos mediante JDBC.



Nombre	conexionJDBC/identificador
Tipo	Cadena
Obligatorio	Sí
Descripción	Identificador de la conexión.

Nombre	conexionJDBC/descripcion
Tipo	Cadena
Obligatorio	No
Descripción	Descripción de la conexión.

Nombre	conexionJDBC/datasource
Tipo	Cadena
Obligatorio	Sí
Descripción	Datasource del que se obtendrán las conexiones.

8 API DEL TOOLKIT

Los métodos que expone el core del toolkit para la ejecución de servicios se encuentran en la clase es.juntadeandalucia.cc.toolkit.sql.CargadorDatos. Todos los métodos se encuentran sobrecargados, de modo que soportan como entrada un mensaje XML de petición o un Map con la relación de parámetros de entrada del servicio, y retornan como repuesta un XML conforme al servicio ejecutado.

- **ejecutarServicioMI**: Genera la respuesta para una petición a un servicio de tipo 'Mi información'.
- **ejecutarServicioMEConsultaAgrupada**: Genera la respuesta para una petición al servicio de consulta agrupada de 'Mis expedientes'.
- **ejecutarServicioMEDetalleOrganismo**: Genera la respuesta para una petición al servicio de consutla de detalle de organismo de 'Mis expedientes'.
- **ejecutarServicioMEDetalleExpediente**: Genera la respuesta para una petición al servicio de consulta de detalle de expediente de 'Mis expedientes'.

Más información sobre estos métodos y clases disponible en el javadoc incluido en la distribución del toolkit.

9 DEPENDENCIAS

Para un correcto funcionamiento del componente toolkit-core se requieren las librerías:

- joda-time-1.6.2.jar
- slf4j-api-1.6.1.jar
- log4j-1.2.15.jar
- commons-codec-1.2.jar

Si el servidor de aplicaciones no es Tomcat 7 o superior será necesario incorporar adicionalmente las siguientes dependencias:

- tomcat-jdbc-7.0.82.jar
- tomcat-juli-7.0.82.jar

En el ejemplo de despliegue de servicios web con Axis 2 descrito en el presente documento, no fue necesaria la inclusión de ninguna de estas dependencias ya que todas ellas vienen incluidas entre los binarios de Axis 2 1.7.7 y el módulo Rampart 1.7.1.

10 NOTAS DE LA VERSIÓN

Notas de esta versión del Toolkit:

- Las consultas relativas a índices de documentos deberán devolver la información en orden descendente acorde a la estructura de árbol. Es decir, partiendo de los nodos raíz y continuando jerárquicamente por los siguientes niveles del árbol.

11 CAMBIOS DE LA VERSIÓN

Versión 1.1.1

Se habilita la configuración de conexión de base de datos a nivel de consulta.

Se soporta el uso de rutas relativas en la configuración del esquema de validación en servicios 'Mi información'.

Corrección de errores.

Versión 1.1

Soporte de integración para sección 'Mi información' mediante estándar SCSP.

Revisión de parámetros de entrada para servicios 'Mi información' ad hoc.

Versión 1.0

Primera versión. Soporte de integración para secciones 'Mi información' ad hoc y 'Mis expedientes' de Carpeta Ciudadana.