

Consulta a Padrón Nivel 10

Manual para el desarrollador

Revisión correspondiente al 27 de Agosto de 2013

Administración Federal de Ingresos Públicos
Subdirección General de Sistemas y Telecomunicaciones
Buenos Aires
27 de Agosto de 2013

Historial de modificaciones

Ver	Fecha	Edición	Descripción
1.0	30-04-2006	DINTR	Versión inicial del documento
1.1	28-04-2009	DIT-CC	Modificación DTD (páginas 11 y 12)
1.2	16-06-2009	DIT-CC	Modificación mensajes error, en Flujo de excepción (páginas 12 y 13)
1.3	27-08-2013	DINTR	Modificaciones de forma y Punto 2.1.1. – Actores Pag.6

Contenido

1	INTRODUCCIÓN	4
1.1.	OBJETIVO	4
1.2.	ALCANCE	4
1.3.	DEFINICIONES, SIGLAS Y ABREVIATURAS	4
2	WS DE NEGOCIO	5
2.1.	CASO DE USO	5
2.1.1.	ACTORES	6
2.1.2.	PRE CONDICIONES	6
2.1.3.	POST CONDICIONES DE ÉXITO	6
2.1.4.	FLUJO PRINCIPAL DE EVENTOS	7
2.1.5.	ENVÍO DEL REQUEST HACIA WSPN10.....	7
2.1.6.	RESPONSE DEL WSPN10.....	9
2.1.7.	FLUJO DE EXCEPCIÓN	12
2.1.8.	SERVICIO DE TEST	13

1 Introducción

El servicio de Consulta de Padrón Nivel10 permite que un organismo externo acceda a los datos de un contribuyente registrado en el Padrón de AFIP.

La consulta se realiza mediante un webServices SOAP que básicamente recibe como parámetro una cuit y responde con los datos públicos del contribuyente identificado con esa cuit.

Este documento está dirigido a quienes tengan la misión de probar y utilizar este webServices.

Para tener acceso a este webServices el organismo usuario debe obtener un ticket de acceso.

El proceso de obtención del ticket de acceso esta fuera del alcance de este documento.

1.1. Objetivo

Este documento está dirigido a quienes tengan la misión de probar y utilizar este webServices y tiene por objeto definir una especificación técnica para el desarrollo de la consulta.

1.2. Alcance

Descripción del Caso de Uso.

1.3. Definiciones, Siglas y Abreviaturas

SOAP	Simple Object Acces Protocolo
WSDL	Web Services Definition Language
WSAA	Web Service de Autenticación y Autorización de AFIP
WSPN10	Web Service de Padrón Nivel10
CE	Cliente externo usuario de los webServices de AFIP
CUIT	Clave Unica de Identificación Tributaria. Campo numérico de 11 dígitos que identificada unívocamente a un contribuyente.
SSO	Ticket para poder acceder a los webServices de AFIP. Son generados por WSAA.

2 WS de Negocio

2.1. Caso de Uso

El caso de uso es muy simple dado que solo implica una operación atómica de consulta mediante una CUIT soportada por un webServices SOAP.

La interfaz del webservices esta descripta en el siguiente WSDL:

<https://domain/padron-puc-ws/services/select.ContribuyenteNivel10SelectServiceImpl?WSDL>

donde el domain puede ser:

- ambiente de Producción: ws.afip.gov.ar
- ambiente de Homologación: awshomo.afip.gov.ar

WSDL
<pre> <?xml version="1.0" encoding="UTF-8" ?> - <wSDL:definitions targetNamespace="https://domain/padron-puc- ws/services/select.ContribuyenteNivel10SelectServiceImpl" xmlns:apacheSOAP="http://xml.apache.org/xml-soap" xmlns:impl= "https://domain/padron-puc-ws/services/select.ContribuyenteNivel10SelectServiceImpl" xmlns:intf="https://domain/padron-puc- ws/services/select.ContribuyenteNivel10SelectServiceImpl" xmlns:SOAPENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:wSDL="http://schemas.xmlsoap.org/wSDL/" xmlns:wSDLSOAP="http://schemas.xmlsoap.org/wSDL/soap/" xmlns:XSD="http://www.w3.org/2001/XMLSchema"> + <!-- WSDL created by Apache Axis version: 1.2RC2 - <wSDL:message name="getRequest"> <wSDL:part name="pContribuyentePK" type="SOAPENC:string" /> <wSDL:part name="pToken" type="SOAPENC:string" /> <wSDL:part name="pSign" type="SOAPENC:string" /> </wSDL:message> - <wSDL:message name="getResponse"> <wSDL:part name="getReturn" type="SOAPENC:string" /> </wSDL:message> - <wSDL:portType name="ContribuyenteNivel10SelectServiceImpl"> - <wSDL:operation name="get" parameterOrder="pContribuyentePK pToken pSign"> <wSDL:input message="impl:getRequest" name="getRequest" /> <wSDL:output message="impl:getResponse" name="getResponse" /> </wSDL:operation> </wSDL:portType> - <wSDL:binding name="select.ContribuyenteNivel10SelectServiceImplSoapBinding" </pre>

```

    type="impl:ContribuyenteNivel10SelectServiceImpl">
    <wsdlsoap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http" />
- <wsdl:operation name="get">
    <wsdlsoap:operation soapAction="" />
- <wsdl:input name="getRequest">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        namespace="http://select.services.server.ws.impl.puc.padron.afip.gov" use="encoded" />
    </wsdl:input>
- <wsdl:output name="getResponse">
    <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        namespace="https://domain/padron-puc-
        ws/services/select.ContribuyenteNivel10SelectServiceImpl" use="encoded" />
    </wsdl:output>
    </wsdl:operation>
    </wsdl:binding>
- <wsdl:service name="ContribuyenteNivel10SelectServiceImplService">
-   <wsdl:port binding="impl:select.ContribuyenteNivel10SelectServiceImplSoapBinding"
        name="select.ContribuyenteNivel10SelectServiceImpl">
        <wsdlsoap:address location="https://domain/padron-puc-
            ws/services/select.ContribuyenteNivel10SelectServiceImpl" />
    </wsdl:port>
    </wsdl:service>
  </wsdl:definitions>

```

2.1.1. Actores

- WSAA
- CE

2.1.2. Pre Condiciones

- El CE debe haber obtenido desde WSAA un SSO para poder usar el WSPN10. El SSO esta compuesto por dos string base64: SSO.token y SSO.signature.
- El SSO debe estar vigente.
- El CE debe conocer la CUIT del contribuyente que se desea consultar.

2.1.3. Post Condiciones de Éxito

- El CE obtiene los datos del contribuyente identificado por la CUIT.

2.1.4. Flujo Principal de Eventos

1. El CE envía un request SOAP al WSPN10 pasando como parámetro un contribuyentePK.xml conteniendo la CUIT del contribuyente que se desea consultar, un SSO.token y un SSO.signature, y se queda esperando respuesta. Ver Envío del Request hacia WSPN10
2. WSPN10 responde con las datos de un contribuyente. Ver Response del WSPN10.

2.1.5. Envío del Request hacia WSPN10

Con el valor de la CUIT del contribuyente que se desea consultar el CE construye un contribuyentePK.xml

Ejemplo de contribuyentePK.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<contribuyentePK>
<id>20000000001</id>
</contribuyentePK>
```

contribuyentePK.dtd

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT contribuyentePK (id)>
<!ELEMENT id (#PCDATA)>
```

El CE envía un request SOAP a la operación get del servicio ContribuyenteNivel10SelectServiceImplService.

La operación get recibe como parámetro un mensaje getRequest compuesto por 3 string pContribuyentePK, pToken y pSing.

Nombre	Descripción						
PContribuyentePK	<p>contribuyentePK.xml conteniendo la cuit del contribuyente que se quiere consultar</p> <p>El xml debe viajar escapeado de acuerdo a lo indicado en http://www.w3.org/TR/REC-xml/.</p> <p>Los caracteres que se escapan son:</p> <table> <tr> <td>Carácter</td><td>Valor escapeado</td></tr> <tr> <td>ampersand (&)</td><td>&amp;</td></tr> <tr> <td>left angle bracket (<)</td><td></td></tr> </table>	Carácter	Valor escapeado	ampersand (&)	&	left angle bracket (<)	
Carácter	Valor escapeado						
ampersand (&)	&						
left angle bracket (<)							

	<p>&lt;</p> <p>right angle bracket (>)</p> <p>&gt;</p> <p>single-quote character (')</p> <p>&apos;</p> <p>double-quote character (")</p> <p>&quot;</p> <p>Por lo general los software estándares utilizados como clientes http o soap se encargan de escapar el contenido de los parámetros string:</p>
Ptoken	<p>SSO.token</p> <p>Es un string base64</p> <p>Si no comienza con "-----" (5 guiones) entonces deben agregarse los delimitadores: "-----BEGIN SSOTOKENBASE64-----\n" y "-----END SSOTOKENBASE64-----"</p>
PSign	<p>SSO.signature</p> <p>Es un string base64</p> <p>Si no comienza con "-----" (5 guiones) entonces deben agregarse los delimitadores: -----BEGIN SSOSIGNBASE64-----\n" y "\n -----END SSOSIGNBASE64-----"</p>

getRequest según el wsdl

```

<wsdl:message name="getRequest">
  <wsdl:part name="pContribuyentePK" type="soapenc:string"/>
  <wsdl:part name="pToken" type="soapenc:string" />
  <wsdl:part name="pSign" type="soapenc:string" />
</wsdl:message>

```


Ejemplo de request soap hacia la operación get

```

<soapenv:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:sel="http://select.services.server.ws.impl.puc.padron.afip.gov">

  <soapenv:Header/>

  <soapenv:Body>

    <sel:get soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">

      <pContribuyentePK xsi:type="soapenc:string"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/">

        <contribuyentePK>

          <id>20000000001</id>

        </contribuyentePK>

      </pContribuyentePK>

      <pToken xsi:type="soapenc:string" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/">-----
BEGIN SSOTOKENBASE64-----
PD94bWwgdWVyc2lvbjo0MS4wliBlbmNvZGluZz0iVVRGLTgiIHNOYW5kYWxvbmU9InllcyllPgo8c3NvIHZlcnNpb249IjIu
uMCI+CiAgICA8aWVwQgdW5pcXVIX2lkPSI0ODk1Nzg5MzMzMiHNYyZ0iQ049d3NhYWwhbW8sIE89QUZJUCwgQz1BUi
wgU0VSSUFTIVNQkVSPUNVSQgMzM2OTM0NTAyMzkilGdlbi90aW1IPSIXMTc1MTkxMjQ5liBleHBfdGltZT0iMT
E3NTIzNDQ0OSIgZHN0PSJDTj1wYWRyb24tcHVjLXdzLWNvbnN1bHRhLW5pdmVsMyxPPUFGSVAsQz1BUitvPgo
glCAgPG9wZXJhdGlviB2YWx1ZT0iZ3JhbnRIZCldHlwZT0ibG9naW4iPgogICAgICAgIDxsb2dpbiB1aWQ9IiINFUkl
BTE5VTUJFUj1DVUIUIUMzMzMzMzMzMzM5LCBDTj1vc2lyaXN0ZXN0LCBPVT1kaW50ciwgTz1kaXJlY2Npb24ga
W5mb3JtYXRpY2EgdHJpYnV0YXJpYS4sIFNUPWJ1ZW5vcyBhaXJlcywgQz1hcilgc2VydmljZT0icGFkcm9uLXB1Yy
13cy1jb25zdWx0YS1uaXZlbiDMilHJlZ21ldGhvZD0iMjliilGVudGI0eT0iMzM2OTM0NTAyMzkilGF1dGhtZXRob2Q9Im
Ntcyl+CiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAg
V5PSIzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzMzM
cGVyYXRpb24+Cjwvc3NvPgoK -----END SSOTOKENBASE64-----</pToken>

      <pSign xsi:type="soapenc:string" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/">-----
BEGIN SSOSIGNBASE64-----
Uul70iNP7WwQWkY7ieyKuoaVvODkPbdr4Jf5zqeBPHeXW6aibPkpwFEJn2gjDoDRktitfFICoJiqSZr2AO06HuBh7jb
+GdqpX6CRq7CNEIqVuiCECXE5wYIH+BT53pZ4bcRf65Och8V5JzQ8g8wHwQUdF4vdEm+LqWVMxHR0A= -----
END SSOSIGNBASE64-----</pSign>

    </sel:get>

  </soapenv:Body>

</soapenv:Envelope>

```

2.1.6. Response del WSPN10

La operación get responde un mensaje getResponse compuesto por un string getReturn conteniendo una session CDATA (no parseable) conteniendo un contribuyente.xml (ver contribuyente.dtd) con los datos del contribuyente consultado.

Ejemplo de response soap desde la operación get

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

  <soapenv:Body>

    <ns1:getResponse soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:ns1="http://select.services.server.ws.impl.puc.padron.afip.gov">

      <getReturn xsi:type="soapenc:string"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"><![CDATA[

```

```
<?xml version="1.0" encoding="UTF-8"?>
<contribuyente>
  <contribuyentePK>
    <id>20000000001</id>
  </contribuyentePK>
  <domicilios>
    <domicilio>
      <domicilioPK>
        <idPersona>20000000001</idPersona>
        <idTipoDomicilio>2</idTipoDomicilio>
        <orden>1</orden>
      </domicilioPK>
      <idEstadoDomicilio>6</idEstadoDomicilio>
      <idTipoNomenclador>2</idTipoNomenclador>
      <idNomenclador>5071</idNomenclador>
      <calle>LOS FEDERALES</calle>
      <numero>1245</numero>
      <codigoPostal>4600</codigoPostal>
      <localidad>ALTO COMEDERO</localidad>
      <idProvincia>6</idProvincia>
      <direccion>LOS FEDERALES 1245</direccion>
      <fechaActualizacion>2006-08-17 16:12:03</fechaActualizacion>
    </domicilio>
    <domicilio>
      <domicilioPK>
        <idPersona>20000000001</idPersona>
        <idTipoDomicilio>1</idTipoDomicilio>
        <orden>1</orden>
      </domicilioPK>
      <idEstadoDomicilio>5</idEstadoDomicilio>
      <idTipoNomenclador>2</idTipoNomenclador>
      <idNomenclador>1</idNomenclador>
      <calle>aqui</calle>
      <numero>2</numero>
      <oficinaDeptoLocal>1</oficinaDeptoLocal>
      <sector>2</sector>
      <torre>3</torre>
      <manzana>4</manzana>
      <piso>5</piso>
      <codigoPostal>4700</codigoPostal>
      <localidad>alla</localidad>
      <idProvincia>6</idProvincia>
      <datoAdicionalDomicilio>junini</datoAdicionalDomicilio>
      <idTipoDatoAdicDomicilio>1</idTipoDatoAdicDomicilio>
      <direccion>junin 2</direccion>
    </domicilio>
  </domicilios>
</contribuyente>
```

```

    <nota>esta es la nota nueva </nota>
    <fechaActualizacion>2006-08-17 16:12:03</fechaActualizacion>
  </domicilio>
</domicilios>
<persona>
  <personaPK>
    <id>20000000001</id>
  </personaPK>
  <tipoPersona>F</tipoPersona>
  <tipold>C</tipold>
  <descripcionCorta>HIGA JUAN</descripcionCorta>
</persona>
</contribuyente>
]]></getReturn>
</ns1:getResponse>
</soapenv:Body>
</soapenv:Envelope>

```

2.1.6.1 contribuyente.dtd

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Este archivo constituye la version 1.1 de la Definicion de Tipo de Documento -->
<!-- del objeto contribuyente (nivel-10) de PUC (Padron Unico de Contribuyentes) -->
<!-- Generado en Abril del 2007 por el sector Padron de la Direccion Informatica Tributaria. -->

<!-- contribuyente (nivel-10) - conjunto de datos incluidos en PUC referidos a una misma clave con filtros de restricción nivel-10-->
<ELEMENT contribuyente (contribuyentePK, domicilios?, persona?)>
<ELEMENT contribuyentePK (id)>
<ELEMENT domicilios (domicilio+)>
<ELEMENT domicilio (domicilioPK, idEstadoDomicilio?, idTipoNomenclador?, idNomenclador?, calle?, numero?, oficinaDeptoLocal?, sector?, torre?, manzana?, piso?, codigoPostal?, localidad?, idProvincia?, datoAdicionalDomicilio?, idTipoDatoAdicDomicilio?, direccion?, nota?, fechaActualizacion?)>
<ELEMENT persona (personaPK, tipoPersona?, tipold?, idDependencia?, razonSocial?, nombre?, apellido?, descripcionCorta?)>

<ELEMENT domicilioPK (idPersona, idTipoDomicilio, orden)>
<ELEMENT personaPK (id)>

<ELEMENT idEstadoDomicilio (#PCDATA)>
<ELEMENT idTipoNomenclador (#PCDATA)>
<ELEMENT idNomenclador (#PCDATA)>
<ELEMENT calle (#PCDATA)>
<ELEMENT codigoPostal (#PCDATA)>
<ELEMENT datoAdicionalDomicilio (#PCDATA)>

```

```

<!ELEMENT direccion (#PCDATA)>
<!ELEMENT fechaActualizacion (#PCDATA)>
<!ELEMENT idPersona (#PCDATA)>
<!ELEMENT idProvincia (#PCDATA)>
<!ELEMENT idTipoDatoAdicDomicilio (#PCDATA)>
<!ELEMENT idTipoDomicilio (#PCDATA)>
<!ELEMENT localidad (#PCDATA)>
<!ELEMENT manzana (#PCDATA)>
<!ELEMENT nota (#PCDATA)>
<!ELEMENT numero (#PCDATA)>
<!ELEMENT oficinaDeptoLocal (#PCDATA)>
<!ELEMENT orden (#PCDATA)>
<!ELEMENT piso (#PCDATA)>
<!ELEMENT sector (#PCDATA)>
<!ELEMENT torre (#PCDATA)>

<!ELEMENT id (#PCDATA)>
<!ELEMENT tipoPersona (#PCDATA)>
<!ELEMENT tipoid (#PCDATA)>
<!ELEMENT idDependencia (#PCDATA)>
<!ELEMENT razonSocial (#PCDATA)>
<!ELEMENT nombre (#PCDATA)>
<!ELEMENT apellido (#PCDATA)>
<!ELEMENT descripcionCorta (#PCDATA)>

```

Consideraciones respecto de contribuyente.xml:

- Solo contiene elementos. No contiene atributos.
- Los campos tipo fecha se informan con el formato "YYYY-MM-DD"
- Los campos tipo fecha-hora se informan con el formato "YYYY-MM-DD HH:MI:SS" con HH desde 00 a 23.

2.1.7. Flujo de Excepción

Ante un error o excepción el webServices retorna un SOAP fault (http://www.w3schools.com/SOAP/soap_fault.asp).

Un SOAP fault esta compuesto entre otros elementos por un Faultcode que es un código que identifica la excepción y un faultstring que es un explicación de la excepción.

Caso de error	Respuesta	
Contribuyente-pk.xml inválido	Faultcode	soapenv:Server.userException
	Faultstring	gov.afip.padron.core.api.exceptions.PadronSystemException: Formato invalido en contribuyentePK
CUIT inexistente	Se devuelve el string vacío en el tag getReturn	
	Faultcode	No genera
	Faultstring	No genera

Caso de error	Respuesta	
CUIT invalida: Dígito verificador inválido	Se devuelve el string vacío en el tag getReturn	
	Faultcode	No genera
	Faultstring	No genera
CUIT invalida: Longitud distinta de 11	Se devuelve el string vacío en el tag getReturn	
	Faultcode	No genera
	Faultstring	No genera
CUIT inválida: Primeros dos dígitos inválidos	Se devuelve el string vacío en el tag getReturn	
	Faultcode	No genera
	Faultstring	No genera
CUIT inválida: No numérica	Faultcode	soapenv:Server.userException
	Faultstring	gov.afip.padron.core.api.exceptions.PadronSystemException: Formato invalido en contribuyentePK
SSO inválido vencido	Faultcode	soapenv:Server.userException
	Faultstring	gov.afip.padron.core.util.security.InvalidRequestException: El token ha expirado, tiempo de generacion [TIEMPO_DE_GENERACION], tiempo actual [TIEMPO_ACTUAL], tiempo de expiracion [TIEMPO_DE_EXPIRACION]
SSO inválido sin delimitadores	Faultcode	soapenv:Server.userException
	Faultstring	gov.afip.padron.core.util.security.InvalidRequestException: no autorizado, falta token y firma
SSO inválido con firma inválida	Faultcode	soapenv:Server.userException
	Faultstring	gov.afip.padron.core.util.security.InvalidRequestException: Par token/sign invalido
SSO inválido no corresponde a WSPN10	Faultcode	soapenv:Server.userException
	Faultstring	gov.afip.padron.core.util.security.InvalidRequestException: El token no sirve para acceder a este sistema
SSO inválido no informado (token y/o sign)	Faultcode	soapenv:Server.userException
	Faultstring	gov.afip.padron.core.util.security.InvalidRequestException: no autorizado, falta token y firma
SSO inválido con caracteres inválidos	Faultcode	soapenv:Server.userException
	Faultstring	gov.afip.padron.core.util.security.MalformedTokenException: El token recibido no tiene la forma esperada.
Error interno del servicio por el cual el mensaje no pudo ser procesado conexión con BD	Faultcode	soapenv:Server.userException
	Faultstring	gov.afip.padron.core.api.exceptions.PadronSystemException: Cannot open connection

2.1.8. Servicio de Test

Debido a que este servicio es de consulta y no modifica información en la base de datos para testear la disponibilidad del servicio el CE puede utilizar su propia CUIT.