



Manual de *Web Services* de **DIR3CAIB**

**Manual d'Integració dels Serveis Web disponibles a
Dir3Caib**



G VICEPRESIDÈNCIA
O I CONSELLERIA
I INNOVACIÓ,
B RECERCA I TURISME



G VICEPRESIDÈNCIA
O I CONSELLERIA
I INNOVACIÓ,
B RECERCA I TURISME



FUNDACIÓ BIT

Informació general del document.

Descripció.

Títol:	Manual de WebServices de Dir3Caib
Estat:	Esborrany/Aprovat
Versió:	1.0
Autor/s:	Antoni Nadal Bennasar, Marilén González
Creat:	10/12/14
Modificat	10/09/18
Fitxer:	Manual_de_WebServices_i_APIREST_de_DIR3CAIB.odt

Històric de modificacions.

Comentari:	Autor/s:	Data:
Actualitzat els camps fecha de String a Date.	Marilen González	17/03/15
Actualitzada API Rest	Marilen González	07/04/17
Afegit apartat Procés Sincronització DIR3CAIB-REGWEB3	Marilen González	09/05/18
Nuevo método WS Unidades: buscarUnidad()	Eduardo Arrivi	10/09/19

Font documental.

Index de Contingut

1.- Introducció.....	5
1.1.- Introducció General.....	5
1.2.- Rols dels Usuaris Aplicació.....	6
1.3.- API de Comunicació.....	6
1.3.1.- Generar codi a partir de WSDL.....	6
1.3.2.- Emprar llibreries compilades.....	7
1.3.3.- Utilitzar jar de l'API del repositori maven.....	7
1.4.- Instanciar API de Web Service.....	7
1.5.- Mètodes Comuns a Totes les APIs.....	8
1.5.1.- String getVersion() i int getVersionWs().....	8
1.6.- Gestió d'Errors.....	9
1.6.1.- Exception.....	9
2.- API de Catàleg.....	9
2.1.- Introducció.....	10
2.2.- Mètodes de l'API Catàleg.....	10
2.2.1.- obtenirCatComunidadAutonoma().....	10
2.2.2.- obtenirCatEntidadGeografica().....	11
2.2.3.- obtenirCatProvincia().....	11
2.2.4.- obtenirCatLocalidad().....	12
2.2.5.- obtenirCatNivelAdministracion().....	12
2.2.6.- obtenirCatEstadoEntidad().....	13
2.2.7.- obtenirCatPais().....	13
3.- API D'Unitats.....	14
3.1.- obtenirArbolUnidades.....	14
3.2.- obtenirUnidad.....	15
3.3.- buscarUnidad.....	16
4.- API d'Oficines.....	16
4.1.- obtenirOficina.....	16
4.2.- obtenirArbolOficinas.....	17
4.3.- obtenirOficinasSIRUnidad.....	17
4.4.- Exemples.....	18
5.- Model de dades.....	20
5.1.- Classe CatEstadoEntidad.....	20
5.2.- Classe CatNivelAdministracion.....	20
5.3.- Classe CatPais.....	21
5.4.- Classe CatComunidadAutonomaTF.....	21
5.5.- Classe CatProvinciaTF.....	21
5.6.- Classe CatLocalidadTF.....	22
5.7.- Classe CatEntidadGeograficaTF.....	22
5.8.- Classe UnidadTF.....	22
5.9.- Classe OficinaTF.....	23
5.10.- Classe RelacionSirOfiTF.....	24
5.11.- Classe RelacionOrganizativaOfiTF.....	24
5.12.- Classe HistoricoUO.....	24
5.13.- ContactoTF.....	24
6.- Descripció dels serveis REST.....	24
6.1.- Serveis.....	25

6.1.1.- unidadesPorDenominacion.....	25
6.1.2.- oficinasPorDenominacion.....	25
6.1.3.- arbolUnidades.....	26
6.1.4.- oficinasOrganismo.....	26
6.1.5.- busquedaOrganismos.....	26
6.1.6.- busquedaOficinas.....	28
6.1.7.- unidadDenominacion.....	29
6.1.8.- oficinaDenominacion.....	29
6.1.9.- organigrama.....	29
6.1.10.- obtenerUnidadesDenominacionComunidad.....	30
6.1.11.- ambitosTerritoriales.....	30
6.1.12.- comunidadesAutonomas.....	31
6.1.13.- entidadesGeograficas.....	31
6.1.14.- provincias.....	32
6.1.15.- provinciasCA.....	32
6.1.16.- localidades.....	33
6.1.17.- nivelesAdministracion.....	33
6.2.- Model de dades dels serveis REST.....	34
6.2.1.- ObjetoDirectorio.....	34
6.2.2.- Nodo.....	34
6.2.3.- CodigoValor.....	35
6.3.- Exemple.....	35
7.- Wsdl.....	36
7.1.- Catàleg.....	36
7.2.- Unitats.....	38
7.3.- Oficines.....	40
8.- Procés Sincronització Organigrama (DIR3CAIB-REGWEB).....	42
8.1.- Exemple 1.....	42
8.2.- Exemple 2 (múltiples fusions divisions).....	43
8.3.- Classes que implementen aquesta funcionalitat.....	44
8.3.1.- ObtenerUnidadesEjb.java.....	44
8.3.2.- SincronizadorDir3Bean.java.....	44

1.- Introducció

1.1.- Introducció General

Dir3Caib ofereix tres web services que ens permeten fer feina amb les tres principals àrees que són Catàleg, Unitats i Oficines.

- **Catàleg:** Dir3Caib proveeix d'un catàleg amb un conjunt de conceptes que ens permet classificar les Unitats i Oficines. El catàleg està format per:
 - Àmbits Territorial
 - Comunitats Autònomes
 - Entitats Geogràfiques
 - Estat d'Entitat
 - Illes
 - Jerarquies Oficina
 - Localitats
 - Motius d'Extinció
 - Nivells d'Administració
 - Països
 - Províncies
 - Tipus de Contactes
 - Tipus d'Entitats Públiques
 - Tipus d'Unitat Orgànica
 - Tipus de Via
- **Unitats:** Dir3Caib també permet obtenir arbres d'unitats o unitats de manera individual.
- **Oficines:** Les oficines funcionen pràcticament igual que les unitats, es pot obtenir un arbre d'oficines, una oficina en concret i un llistat de les oficines SIR d'una unitat especificada.

Les àrees de les que s'ha parlat anteriorment són:

<i>Nom</i>	<i>Descripció</i>	<i>Context</i>
Catàlegs	Obté els catàlegs de l'aplicació	/dir3caib/ws/Dir3CaibObtenerCatalogos
Unitats	Obté les unitats de l'aplicació i les seves relacions	/dir3caib/ws/Dir3CaibObtenerUnidades
Oficines	Obté les Oficines de l'aplicació i les seves relacions	/dir3caib/ws/Dir3CaibObtenerOficinas

Per exemple, en un servidor en el nostre ordinador, la ruta per accedir als serveis de Catàlegs seria la següent <http://localhost:8080/dir3caib/ws/Dir3CaibObtenerCatalogos> i per accedir al WSDL de Catàlegs seria la següent

<http://localhost:8080/dir3caib/ws/Dir3CaibObtenerCatalogos?wsdl>.

1.2.- Rols dels Usuaris Aplicació

Els usuaris aplicació són els usuaris autoritzats a connectar-se via web services amb el DIR3CAIB. El rol que han de tenir és el següent:

- **ROLE ADMIN (DIR_ADMIN):** És el rol administrador de l'aplicació i que permet fer les operacions dels diferents ws.

Quan es detalli cada mètode de web service s'indicarà quins rols es requereixen per poder cridar al mètode en qüestió.

Com és lògic, els web services no autenticats no requereixen d'usuari aplicació per la cridada dels seus mètodes cosa que fa sense sentit els rols descrits en aquest punt.

1.3.- API de Comunicació

Per connectar-se amb el servidor es necessita un conjunt de classes java (o qualsevol altra llenguatge) que anomenarem API de comunicació. Aquesta API de comunicació està formada per elements Bean que contindran la informació a passar i uns gestors que són classes amb mètodes que són les accions disponibles per cada web service. El codi necessari per connectar-se al servidor via web services es pot aconseguir de tres formes:

- Generar codi a partir de wsdl o
- Emprar llibreries compilades
- Utilitzar jar de l'API del repositori maven

Es recomana la tercera opció ja que aquesta opció inclou classes d'utilitat de client web service com classes amb les constants del projecte Dir3Caib, mètodes d'utilitat de gestió d'excepcions i fitxers de traduccions, i a més no s'ha d'anar a cercar el jar per la web de github.

1.3.1.- Generar codi a partir de WSDL

Per exemple per obtenir l'API de comunicació del web service Dir3CaibObtenerCatalogos executariem la següent comanda:

```
wsconsume -k http://localhost:8080/dir3caib/ws/Dir3CaibObtenerCatalogos?wsdl -s src/main/java -n -p es.caib.dir3caib.ws.api.v1
```

El programa wsconsume és un script (.bat i .sh) que es troba en el directori bin del JBoss al que abans se li ha instal·lat el *patch* de CXF i el que fa és generar el codi java necessari per connectar-se al web service.

1.3.2.- Emprar llibreries compilades

Com ja s'ha comentat la utilització de les APIs ja compilades són la segona millor opció ja que inclou un conjunt de classes d'utilitat com són les constants utilitzades en les cridades a l'API.

Aquesta API la podem trobar a la següent ruta `ws/dir3caib_api/target/dir3caib-ws-api-1.0.X.jar` si compilam el codi de DIR3CAIB o en la mateixa ruta si ens descarregam el zip del binari de Dir3Caib que trobarem dins el repositori a github <https://github.com/GovernIB/dir3caib>

1.3.3.- Utilitzar jar de l'API del repositori maven

És el mateix jar descrit en el punt anterior però no l'hem d'anar a cercar a cap pàgina web sinó que està emmagatzemat dins d'un repositori maven que s'ha fet dins el projecte DIR3CAIB de github. Només hem d'afegir el repositori de DIR3CAIB dins el nostre pom.xml del projecte:

```
<repositories>
  <repository>
    <id>es.caib.dir3caib</id>
    <name>Dir3Caib Maven Repository</name>
    <url>http://GovernIB.github.io/dir3caib/maven/</url>
  </repository>
</repositories>
```

I a més afegir la dependència al jar corresponent:

```
<dependencies>
  <dependency>
    <artifactId>dir3caib-ws-api</artifactId>
    <groupId>es.caib.dir3caib</groupId>
    <version>[dir3caib_required_version]</version>
  </dependency>
</dependencies>
```

Anam per la versió 1.0.7

1.4.- Instanciar API de Web Service

El serveis descrits en el punt 1.1 Introducció General és poden instanciar com es mostra a continuació. Només requerim un projecte amb un jar que inclogui l'api del punt 1.3 API de Comunicació.

```
String endpoint="http://localhost:8080/dir3caib/ws/Dir3CaibObtenerCatalogos";
URL wsd1 = new URL(endpoint + "?wsdl");
Dir3CaibObtenerCatalogosWsService catalagosService =
    new Dir3CaibObtenerCatalogosWsService(wsd1);
Dir3CaibObtenerCatalogosWs catalago =
    catalagosService.getDir3CaibObtenerCatalogosWs();

// Adreça servidor
String usr_app = "user_app"; // username application
String pwd_app = "mypassword"; // password application
Map<String, Object> reqContext = ((BindingProvider)
    catalago).getRequestContext();
reqContext.put(BindingProvider.ENDPOINT_ADDRESS_PROPERTY, endpoint);
reqContext.put(BindingProvider.USERNAME_PROPERTY, usr_app);
reqContext.put(BindingProvider.PASSWORD_PROPERTY, pwd_app);

System.out.println("Versio " + catalago.getVersion());
```

Les dues línies en negreta introdueixen dins la petició l'usuari i contrasenya de l'usuari aplicació que volem utilitzar per fer la cridada.

1.5.- Mètodes Comuns a Totes les APIs

Cada Web Service conté uns mètodes d'utilitat comuns a totes les APIs i que són:

1.5.1.- String getVersion() i int getVersionWs()

Retorna la versió del Dir3Caib i la versió dels Web Service del Dir3Caib respectivament.

Vegem un exemple emprant l'API instanciada en el punt anterior:

```
System.out.println("Versió Dir3Caib : " + catalago.getVersion());
System.out.println("Versió Dir3Caib-WS: " + catalago.getVersionWs());
```

I la sortida de l'execució del programa obtindrem una cosa com:

```
Versió Dir3Caib : 1.0.0
Versió Dir3Caib-WS: 1.0
```



En l'entorn de la CAIB (Govern Balear) la versió de Dir3Caib sempre inclourà un sufix “-caib” de forma que l'execució del programa anterior en un servidor Dir3Caib de la CAIB mostrarà per pantalla el següent:

```
Versió Dir3Caib : 1.0.0-caib
Versió Dir3Caib-WS: 1.0
```


1.6.- Gestió d'Errors

Les excepcions controlades que llança Dir3caib estan preparades per ser traduïdes en el client.

Existeixen tres tipus d'excepcions:

1.6.1.- Exception

Són excepcions no controlades per Dir3Caib o gestionades pel propi sistema de web services (camps o paràmetres requerits, camps o paràmetres no null, autenticació, autorització, ...). Amb un simple try-catch es poden capturar.

A causa de la diversitat de tipus d'excepcions s'engloben en Exception.

Aquí és mostra el try-catch per l'exemple del punt 1.4Instanciar API de Web Service:

```
try {
    Dir3CaibObtenerCatalogosWs api = getObtenerCatalogosApi(true);
    List<CatComunidadAutonomaTF> list = api.obtenerCatComunidadAutonoma();
} catch (Exception e) {
    System.err.println("Exception: " + e.getMessage());
    e.printStackTrace(System.err);
}
```

Si no existeix el servidor o no està en marxa:

```
Exception: HTTP transport error: java.net.UnknownHostException: localhost22
com.sun.xml.internal.ws.client.ClientTransportException: HTTP transport error:
java.net.UnknownHostException: localhost22
```

Si l'usuari no té rols, l'usuari no és correcte o la contrasenya no és correcte llavors l'error és:

```
Exception: The server sent HTTP status code 401: No Autorizado
com.sun.xml.internal.ws.client.ClientTransportException: The server sent HTTP
status code 401: No Autorizado
```

Si l'usuari no té els rols requerits pel mètode o servei que crida llavors l'error és:

```
Exception: Caller unauthorized
javax.xml.ws.soap.SOAPFaultException: Caller unauthorized
```

Quan algun paràmetre es valida en la capa de web services (En aquest cas s'ha cridat al mètode obtenerCatComunidadAutonoma() del servei Dir3CaibObtenerCatalogosWs):

```
Exception: Method Parameter: obtenerCatComunidadAutonoma cannot be null. This
is BP 1.1 R2211 violation.
javax.xml.ws.WebServiceException: Method Parameter: obtenerCatComunidadAutonoma
cannot be null. This is BP 1.1 R2211 violation.
```

2.- API de Catàleg

Les següents notes s'aplicaran a totes les APIs descrites en aquest document.

NOTA 1: En les següents descripcions de mètodes, en el prototipus es descriurà la llista de permisos necessaris de l'usuari aplicació per poder cridar al mètode en qüestió.

NOTA 2: Els mètodes que llancin algun tipus d'excepció, significa que en certes condicions aquestes excepcions poden ser llançades, per la qual cosa el programador és l'encarregat de gestionar-les com toqui.

2.1.- Introducció

Aquesta API serveix per obtenir un subconjunt del catàleg de dir3Caib. La llista de mètodes disponibles són (no es mostren els mètodes comuns):

- `obtenerCatComunidadAutonoma()`
- `obtenerCatEntidadGeografica()`
- `obtenerCatProvincia()`
- `obtenerCatLocalidad()`
- `obtenerCatNivelAdministracion()`
- `obtenerCatEstadoEntidad()`
- `obtenerCatPais()`

Per tots aquests mètodes, l'usuari aplicació requereix tenir el role DIR_ADMIN.

2.2.- Mètodes de l'API Catàleg

2.2.1.- `obtenerCatComunidadAutonoma()`

Prototipus:
<pre>@RolesAllowed({ DIR_ADMIN }) public List<CatComunidadAutonomaTF> obtenerCatComunidadAutonoma() throws Exception;</pre>
Descripció:
Aquest mètode serveix per obtenir totes les comunitats autònomes
Paràmetres:
Resposta:
<ul style="list-style-type: none">Llista de CatComunidadAutonomaTF , la descripció del bean CatComunidadAutonomaTF es pot veure a l'apartat 5.4Classe CatComunidadAutonomaTF

2.2.2.- obtenerCatEntidadGeografica()

Prototipus:
<pre>@RolesAllowed({ DIR_ADMIN }) public List<CatEntidadGeograficaTF> obtenerCatEntidadGeografica() throws Exception;</pre>
Descripció:
Aquest mètode serveix per obtenir totes les Entitats geogràfiques
Paràmetres:
Resposta:
<ul style="list-style-type: none">Llista de CatEntidadGeograficaTF , la descripció del bean CatEntidadGeograficaTF es pot veure a l'apartat 5.7Classe CatEntidadGeograficaTF

2.2.3.- obtenerCatProvincia()

Prototipus:
<pre>@RolesAllowed({ DIR_ADMIN }) public List<CatProvínciaTF> obtenirCatProvíncia() throws Exception;</pre>
Descripció:
Aquest mètode serveix per obtenir totes les províncies.
Paràmetres:
Resposta:
<ul style="list-style-type: none">Llista de CatProvínciaTF , la descripció del bean CatProvínciaTF es pot veure a l'apartat 5.5Classe CatProvínciaTF

2.2.4.- obtenirCatLocalidad()

Prototipus:
<pre>@RolesAllowed({DIR_ADMIN }) public List<CatLocalidadTF> obtenirCatLocalidad() throws Exception;</pre>
Descripció:
Aquest mètode serveix per obtenir totes les localitats
Paràmetres:
Resposta:
<ul style="list-style-type: none">Llista de CatLocalidadTF , la descripció del bean CatLocalidadTF es pot veure a l'apartat 5.6Classe CatLocalidadTF

2.2.5.- obtenirCatNivelAdministracion()

Prototipus:
<pre>@RolesAllowed({DIR_ADMIN }) public List<CatNivelAdministracion> obtenerCatNivelAdministracion() throws Exception;</pre>
Descripció:
Aquest mètode serveix per obtenir tots els nivells d'administració
Paràmetres:
Resposta:
<ul style="list-style-type: none">Llista de CatNivelAdministracion , la descripció del bean CatNivelAdministracion es pot veure a l'apartat 5.2Classe CatNivelAdministracion

2.2.6.- obtenerCatEstadoEntidad()

Prototipus:
<pre>@RolesAllowed({ DIR_ADMIN }) public List<CatEstadoEntidad> obtenerCatEstadoEntidad() throws Exception;</pre>
Descripció:
Aquest mètode serveix per obtenir tots els estats que pot tenir una entitat.
Paràmetres:
Resposta:
<ul style="list-style-type: none">Llista de CatEstadoEntidad , la descripció del bean CatEstadoEntidad es pot veure a l'apartat 5.1Classe CatEstadoEntidad

2.2.7.- obtenerCatPais()

Prototipus:
<pre>@RolesAllowed({DIR_ADMIN }) public List<CatPais> obtenerCatPais() throws Exception;</pre>
Descripció:
Aquest mètode serveix per obtenir tots els països.
Paràmetres:
Resposta:
<ul style="list-style-type: none">• Llista de CatPais , la descripció del bean CatPais es pot veure a l'apartat 5.3Classe CatPais

3.- API D'Unitats

Aquesta API permet obtenir un arbre d'unitats a partir d'una arrel i obtenir informació d'unitats. El llistat de mètodes que conté aquesta API són:

- obtenerArbolUnidades
- obtenerUnidad
- buscarUnidad

3.1.- obtenerArbolUnidades

Prototipus:

```
@RolesAllowed({ DIR_ADMIN })  
public List<UnidadTF> obtenerArbolUnidades(String codigo, Date  
fechaActualizacion, Date fechaSincronizacion) throws Exception;
```

Descripció:

Mètode que retorna l'arbre d'unitats de la unitat indicada pel codi, partint de la data de la darrera actualització que indiquen i la data de la sincronització indicada també. Si no s'indica data d'actualització es considera que és una sincronització i et retorna tot l'arbre d'unitats vigents. Si s'indica data d'actualització, retorna les unitats que han estat actualitzades dins el període de temps indicat tenint en compte també la data de la darrera importació des de dir3 Madrid.

S'empra per a la sincronització i actualització amb regweb

El valor que retorna és l'arbre en forma de llista.

Paràmetres:

- **String codigo:** codi de la unitat arrel
- **Date fechaActualizacion:** data de la darrera actualització
- **Date fechaSincronizacion:** data en la que es va fer la primera sincronització amb dir3caib.

Resposta:

- Llista d'UnidadTF , la descripció del bean UnidadTF es pot veure a l'apartat 5.8Classe UnidadTF

3.2.- obtenerUnidad

Prototipus:

```
@RolesAllowed({ DIR_ADMIN })  
public UnidadTF obtenerUnidad(String codigo, Date fechaActualizacion, Date  
fechaSincronizacion) throws Exception;
```

Descripció:

Mètode que retorna una UnidadTF **vigent** a partir del codi indicat en funció de la data d'actualització i la de sincronització (indica quan es va fer la primera sincronització). Si la data d'actualització és posterior a la d'importació amb dir3 de Madrid es suposa que ha canviat i s'envia.

Paràmetres:

- **String codigo:** codi de la unitat arrel
- **Date fechaActualizacion:** data de la darrera actualització amb dir3caib.
- **Date fechaSincronizacion:** data en la que es va fer la primera sincronització amb dir3caib.

Resposta:

- UnidadTF (la unitat en qüestió). La descripció del bean UnidadTF es pot veure a l'apartat 5.8Classe UnidadTF

3.3.- buscarUnidad

Prototipus:
<pre>@RolesAllowed({ DIR_ADMIN }) public UnidadTF buscarUnidad(String codigo) throws Exception;</pre>
Descripció:
Mètode que retorna una UnidadTF a partir del codi indicat, sense tenir en compte el seu estat.
Paràmetres:
<ul style="list-style-type: none">• String codigo: codi de la unitat
Resposta:
<ul style="list-style-type: none">• UnidadTF (la unitat en qüestió). La descripció del bean UnidadTF es pot veure a l'apartat 5.8Classe UnidadTF

4.- API d'Oficines

La següent API permet obtenir les oficines relacionades amb unitats depenent de certs criteris. El llistat de mètodes que conté aquesta API són:

- obtenerOficina
- obtenerArbolOficinas
- obtenerOficinasSIRUnidad

4.1.- obtenerOficina

Prototipus:
<pre>@RolesAllowed({ DIR_ADMIN }) public OficinaTF obtenerOficina(String codigo, Date fechaActualizacion, Date fechaSincronizacion) throws Exception;</pre>
Descripció:
Obté les dades d'una oficina en funció del codi i la data d'actualització i la data de la primera sincronització. Si la data d'actualització és posterior a la d'importació amb dir3 de Madrid se suposa que ha canviat i s'envia.
Paràmetres:

- **String codigo:** Codi de l'oficina.
- **Date fechaActualizacion:** data en la que es fa l'actualització.
- **Date fechaSincronizacion:** data en la que es va fer la primera sincronització.

Resposta:

- **OficinaTF:** Les dades de l'oficina en qüestió. El bean que la descriu es pot consultar a l'apartat 5.9Classe OficinaTF

4.2.- obtenerArbolOficinas

Prototipus:

```
@RolesAllowed({ DIR_ADMIN })
public List<OficinaTF> obtenerArbolOficinas(String codigo, Date
fechaActualizacion, Date fechaSincronizacion) throws Exception;
```

Descripció:

Obté l'arbre d'oficines que depenen de la unitat indicada pel codi. Només s'envien aquelles que han estat actualitzades. Si no s'indica data d'actualització es considera que és una sincronització i et retorna tot l'arbre d'oficines vigents relacionades amb la unitat.

Si s'indica data d'actualització, retorna les oficines que han estat actualitzades dins el període de temps indicat considerant també la data de la darrera importació des de dir3 Madrid.

S'empra per a la sincronització i actualització amb regweb

El valor que retorna és l'arbre en forma de llista.

Paràmetres:

- **String codigo:** codi de la unitat
- **Date fechaActualizacion:** data de la darrera actualització amb dir3caib.
- **Date fechaSincronizacion:** data en la que es va fer la primera sincronització amb dir3caib.

Resposta:

- Llista d'OficinaTF , la descripció del bean OficinaTF es pot veure a l'apartat 5.8Classe UnidadTF

4.3.- obtenerOficinasSIRUnidad

Prototipus:

```
@RolesAllowed({ DIR_ADMIN })
public List<OficinaTF> obtenerOficinasSIRUnidad(String codigoUnidad) throws
Exception;
```

Descripció:

Obté el llistat d'oficines integrades en SIR de la unitat indicada pel codi.

Paràmetres:

- **String codigoUnidad:** codi de la unitat

Resposta:

- Llista d'OficinaTF, la descripció del bean OficinaTF es pot veure a l'apartat 5.9Classe OficinaTF

4.4.- Exemples

El següent exemple mostra la petició de les comunitats autònomes de dir3caib.

```
package es.caib.dir3caib.ws.api.test;

import java.io.FileInputStream;
import java.net.URL;
import java.util.List;
import java.util.Map;
import java.util.Properties;

import javax.xml.ws.BindingProvider;

import org.junit.Test;

import es.caib.dir3caib.ws.api.catalogo.CatComunidadAutonomaTF;
import es.caib.dir3caib.ws.api.catalogo.Dir3CaibObtenerCatalogosWs;
import es.caib.dir3caib.ws.api.catalogo.Dir3CaibObtenerCatalogosWsService;

/**
 *
 * @author anadal
 */
public class TestWs {

    public static final String OBTENERCATALOGOS = "Dir3CaibObtenerCatalogos";

    private static Properties testProperties = new Properties();

    static {
        // Propietats del Servidor
        try {
            testProperties.load(new FileInputStream("test.properties"));
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    public static String getEndPoint(String api) {
        return testProperties.getProperty("test_host") + api;
    }

    public static String getTestAppUserName() {
        return testProperties.getProperty("test_usr");
    }

    public static String getTestAppPassword() {
```

```
        return testProperties.getProperty("test_pwd");
    }

    @Test
    public void test() {
        main(new String[] {});
    }

    public static Dir3CaibObtenerCatalogosWs getObtenerCatalogosApi(boolean
setUsrPwd)throws Exception {
        final String endpoint = getEndPoint(OBTENERCATALOGOS);

        System.out.println("endPoint: " + endpoint);

        URL wsdlLocation = new URL(endpoint + "?wsdl");
        System.out.println("WSDL: " + wsdlLocation);

        Dir3CaibObtenerCatalogosWsService service= new
        Dir3CaibObtenerCatalogosWsService(wsdlLocation);

        Dir3CaibObtenerCatalogosWs api = service.getDir3CaibObtenerCatalogosWs();

        if (setUsrPwd) {
            configAddressUserPassword(getTestAppUserName(), getTestAppPassword(),
            endpoint, api);
        }

        return api;
    }

    public static void configAddressUserPassword(String usr, String pwd, String
endpoint, Object api) {

        Map<String, Object> reqContext = ((BindingProvider) api).getRequestContext();
        reqContext.put(BindingProvider.ENDPOINT_ADDRESS_PROPERTY, endpoint);
        reqContext.put(BindingProvider.USERNAME_PROPERTY, usr);
        reqContext.put(BindingProvider.PASSWORD_PROPERTY, pwd);
    }

    public static void main(String[] args) {

        try {

            Dir3CaibObtenerCatalogosWs api = getObtenerCatalogosApi(true);

            System.out.println("Versio: " + api.getVersion());

            List<CatComunidadAutonomaTF> list = api.obtenerCatComunidadAutonoma();

            for (CatComunidadAutonomaTF catComunidadAutonomaTF : list) {
                System.out.println(catComunidadAutonomaTF.getCodigoComunidad() + "\t\t"
                + catComunidadAutonomaTF.getDescripcionComunidad());
            }

        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
}  
  
}
```

5.- Model de dades

A continuació es descriuen les diferents classes que comporten el model de dades que es transfereix via Ws i que és un subconjunt de dades del model de dir3caib complet.

5.1.- Classe CatEstadoEntidad

Classe que representa els possibles estats en els que es pot trobar una Entitat(Unidad, Oficina, etc)

Nom	Tipus	Descripció
codigoEstadoEntidad	String	Codi de l'estat de l'entitat (V,A,E,T)
descripcionEstadoEntidad	String	Descripció de l'estat de l'entitat V= Vigent A=Anul·lat E=Extingit T=Transitori

5.2.- Classe CatNivelAdministracion

Classe que representa els diferents nivells d'administració que hi ha a dir3Caib. Els valors són:

- Administración Autonómica
- Administración General del Estado
- Administración Institucional
- Administración Local
- Historicos AGE
- Historicos CCAA
- Historicos EELL
- Universidades

Nom	Tipus	Descripció
-----	-------	------------

codigoNivelAdministracion	String	Codi del nivell d'administració
descripcionNivelAdministracion	String	Descripció del nivell d'administració

5.3.- Classe CatPais

Classe que representa un País.

Nom	Tipus	Descripció
codigoPais	Long	Numèric que indica el codi del País
descripcionPais	String	Descripció del país.
alfa3Pais	String	Codi alfa3 del País.
alfa2Pais	String	Codi alfa2 del País.

5.4.- Classe CatComunidadAutonomaTF

Classe que representa una comunitat autònoma.

Nom	Tipus	Descripció
codigoComunidad	Long	Codi de la comunitat.
descripcionComunidad	String	Descripció de la comunitat.
codigoPais	Long	Codi del país al que pertany la comunitat.

5.5.- Classe CatProvinciaTF

Classe que representa una província.

Nom	Tipus	Descripció
codigoProvincia	Long	Codi de la província
codigoComunidadAutonoma	Long	Codi de la comunitat autònoma a la que pertany.
descripcionProvincia	String	Descripció de la província.

5.6.- Classe CatLocalidadTF

Classe que representa una localitat.

Camp	Tipus	Descripció
codigoLocalidad	Long	Codi de la localitat.
codigoProvincia	Long	Codi de la província a la que pertany.
descripcionLocalidad	String	Descripció de la localitat.
codigoEntidadGeografica	String	Codi de l'entitat geogràfica a la que pertany.

5.7.- Classe CatEntidadGeograficaTF

Classe que representa una entitat geogràfica.

Camp	Tipus	Descripció
codigoEntidadGeografica	String	Codi entitat geogràfica
descripcionEntidadGeografica	String	Descripció de l'entitat geogràfica.

5.8.- Classe UnidadTF

Classe que representa el conjunt de dades d'una unitat que són transferides via WS.

Camp	Tipus	Cardinalitat	Descripció
codigo	String	1	Codi de la unitat.
denominacion	String	1	Denominació de la unitat.
codigoEstadoEntidad	String	1	Codi del estat de la unitat.
nivelJerarquico	Long	0..1	Nivell jeràrquic de la unitat.
codUnidadSuperior	String	1	Codi de la seva unitat superior
codUnidadRaiz	String	1	Codi de la seva unitat arrel.
competencias	String	0..1	Descripció de les competències de la unitat.
fechaAltaOficial	Date	1	Data en la que es va donar d'alta per primera vegada la unitat.
fechaBajaOficial	Date	0..1	Data en la que es va donar de baixa.

fechaExtincion	Date	0..1	Data en la que es va extingir.
fechaAnulacion	Date	0..1	Data en la que es va anul·lar
nivelAdministracion	Long	1	Nivell d'administració al que pertany
codAmbProvincia	Long	0..1	Codi de la província a la que pertany.
codAmbComunidad	Long	0..1	Codi de la comunitat autònoma a la que pertany.
historicosUO	Set<String>	0..n	Conjunt de codis que substitueixen a la unitat indicada. Veure detall de funcionament a 8Procés Sincronització Organigrama (DIR3CAIB-REGWEB)
contactos	List<ContactoTF>	0..n	Llistat de contactes que te l'unitat

5.9.- Classe OficinaTF

Classe que representa el conjunt de dades d'una oficina que són transferides via WS.

Camp	Tipus	Cardinalitat	Descripció
codigo	String	1	Codi de l'oficina.
denominacion	String	1	Denominació de l'oficina.
estado	String	1	Estat de l'oficina.
nivelAdministracion	Long	1	Nivell d'administració de l'oficina.
tipoOficina	Long	0..1	Tipus d'oficina (auxiliar o general)
codUoResponsable	String	0..1	Codi de la seva unitat responsable (relació Funcional).
codOfiResponsable	String	0..1	Codi de la seva oficina responsable.
sirOfi	List<RelacionSirOfiTF>	1..n	Llistat de les relacions amb les unitats a les que dona servei via SIR.
organizativasOfi	List<RelacionOrganizativaOfiTF>	0..n	Llistat de les relacions organitzatives de les unitats a les que dona servei.
contactos	List<ContactoTF>	0..n	Llistat de contactes que te l'oficina

5.10.- Classe RelacionSirOfiTF

Classe que representa la relació SIR entre una Oficina i una Unitat. Ens permet saber quina és l'oficina de la unitat que es troba adherida a la plataforma d'intercanvi registral del SIR

Camp	Tipus	Descripció
oficina	String	Codi de l'oficina
unidad	String	Codi de la unitat
estado	String	Estat de la relació.

5.11.- Classe RelacionOrganizativaOfiTF

Classe que representa la dependència funcional entre una Oficina i una Unitat.

Camp	Tipus	Descripció
oficina	String	Codi de l'oficina
unidad	String	Codi de la unitat
estado	String	Estat de la relació.

5.12.- Classe HistoricoUO

Camp	Tipus	Descripció
codanterior	String	Codi de la unitat que és substituïda
codultima	String	Codi de la unitat que substitueix a la anterior

5.13.- ContactoTF

Camp	Tipus	Descripció
tipoContacto	String	Tipo de contacte.(Tel,email,Fax...)
valorContacto	String	Valor del contacte.

6.- Descripció dels serveis REST

A continuació es descriuen els serveis REST que proporciona Dir3Caib.

6.1.- Serveis

6.1.1.- unidadesPorDenominacion

Prototipus:
<pre>public @ResponseBody ResponseEntity<List<ObjetoDirectorio>> unidadesPorDenominacion(@RequestParam String denominacion) throws Exception;</pre>
Descripció:
Aquest servei cerca organismes/unitats que contenguin la denominació indicada.
Paràmetres:
<ul style="list-style-type: none">String denominacion: denominació de l'organisme que cercam
Resposta:
<ul style="list-style-type: none">Llista <ObjetoDirectorio>, la descripció del bean ObjetoDirectorio es pot veure a l'apartat 6.2.1ObjetoDirectorio
RequestMapping:
[urlServidor]/rest/unidad/unidadesDenominacion

6.1.2.- oficinasPorDenominacion

Prototipus:
<pre>public @ResponseBody ResponseEntity<List<ObjetoDirectorio>> oficinasPorDenominacion(@RequestParam String denominacion) throws Exception;</pre>
Descripció:
Aquest servei cerca oficines que contenguin la denominació indicada.
Paràmetres:
<ul style="list-style-type: none">String denominacion: denominació de l'oficina que cercam
Resposta:
<ul style="list-style-type: none">Llista <ObjetoDirectorio>, la descripció del bean ObjetoDirectorio es pot veure a l'apartat 6.2.1ObjetoDirectorio
RequestMapping:
[urlServidor]/rest/oficina/oficinasDenominacion

6.1.3.- arbolUnidades

Prototipus:
<pre>public @ResponseBody ResponseEntity<List<ObjetoDirectorio>> arbolUnidades(@RequestParam String codigo) throws Exception;</pre>
Descripció:
Aquest servei obté l'arbre d'organismes/unitats del codi indicat
Paràmetres:
String codigo: codi de l'organisme arrel del que volem obtenir el seu arbre
Resposta:
<ul style="list-style-type: none">Llista <ObjetoDirectorio>, la descripció del bean ObjetoDirectorio es pot veure a l'apartat 6.2.1ObjetoDirectorio
RequestMapping:
[urlServidor]/rest/unidad/arbolUnidades

6.1.4.- oficinasOrganismo

Prototipus:
<pre>public @ResponseBody ResponseEntity<List<ObjetoDirectorio>> oficinasOrganismo(@RequestParam String codigo) throws Exception;</pre>
Descripció:
Aquest servei obté les oficines que depenen de l'organisme del codi indicat
Paràmetres:
String codigo: codi de l'organisme del que volem obtenir les seves oficines
Resposta:
<ul style="list-style-type: none">Llista <ObjetoDirectorio>, la descripció del bean ObjetoDirectorio es pot veure a l'apartat 6.2.1ObjetoDirectorio
RequestMapping:
[urlServidor]/rest/oficina/oficinasOrganismo

6.1.5.- busquedaOrganismos

Prototipus:

```
Public @ResponseBody
ResponseEntity<List<Nodo>> busquedaOrganismos(@RequestParam String codigo,
@RequestParam String denominacion, @RequestParam Long codNivelAdministracion,
@RequestParam Long codComunidadAutonoma, @RequestParam boolean conOficinas,
@RequestParam boolean unidadRaiz, @RequestParam Long provincia, @RequestParam
String localidad, @RequestParam boolean vigentes) throws Exception;
```

Descripció:

Aquest servei permet fer una cerca d'organismes dins dir3 en funció dels paràmetres indicats. S'empra per cercar organismes externs destinataris quan es fa un registre d'entrada/sortida

Paràmetres:

- String codigo: codi de l'organisme que cerquem.
- String denominacion: denominació de l'organisme que cerquem
- Long codNivelAdministracion: codi del nivell d'administració dels organismes que cerquem
- Long codComunidadAutonoma: codi de la comunitat autònoma per la qual cerquem.
- Boolean conOficinas : Indicam si volem que només es tornin els organismes que tinguin oficines.
- Boolean unidadRaiz: indicam que només volem que es retornin els organismes arrels.
- Long provincia: Indicam la província dels organismes que cerquem
- String localidad: Indicam la localitat dels organismes que cerquem.
- Boolean vigentes: Indicam que només volem aquells organismes que són vigents

Resposta:

- Llista <Nodo>, la descripció del bean Nodo es pot veure a l'apartat 6.2.2Nodo

RequestMapping:

[urlServidor]/rest/busqueda/organismos

Exemple:

```
http://localhost:8080/dir3caib/rest/busqueda/organismos?
codigo=&denominacion=&codNivelAdministracion=3&codComunidadAutonoma=4&co
nOficinas=false&unidadRaiz=true&provincia=7&localidad=407-
01&vigentes=true
```

En aquest enllaç es pot trobar els valors estàndars per el «directorio común» que s'han d'indicar als paràmetres.

<http://administracionelectronica.gob.es/ctt/dir3/descargas> a l'apartat de «Catálogos».

La resta de valors que no es trobin es poden trobar al «manual de Atributos- pg 62» al mateix enllaç a l'apartat «Manuales de Usuario».

Nota: als codis se'ls ha de llevar el 0 de davant.

Nota2: El codi de la localidad està format per codigoLocalidad (sense 0 inicial) – Cod.Entidad

Exemple Localidad:

Localidad: Palma (407-01)

6.1.6.- busquedaOficinas

Prototipus:
<pre>public @ResponseBody ResponseEntity<List<Nodo>> busquedaOficinas(@RequestParam String codigo, @RequestParam String denominacion, @RequestParam Long codNivelAdministracion, @RequestParam Long codComunidadAutonoma, @RequestParam Long provincia, @RequestParam String localidad, @RequestParam boolean oficinasSir, @RequestParam boolean vigentes) throws Exception;</pre>
Descripció:
Aquest servei permet fer una cerca d'oficines origen dins dir3 en funció dels paràmetres indicats. S'empra per cercar oficines d'origen externes quan es fa un registre d'entrada/sortida
Paràmetres:
<ul style="list-style-type: none"> • String codigo: codi de l'oficina que cercam. • String denominacion: denominació de l'oficina que cercam • Long codNivelAdministracion: codi del nivell d'administració de les oficines que cercam • Long codComunidadAutonoma: codi de la comunitat autònoma per la qual cercam. • Long provincia: indicam la provincia de les oficines que cercam. • String localidad: indicam la localitat de les oficines que cercam. • Boolean oficinasSir: indicam que només volem aquelles oficines que estan integrades en SIR. • Boolean vigentes: indicam que només volem aquelles oficines que són vigents.
Resposta:
<ul style="list-style-type: none"> • Llista<Nodo>, la descripció del bean Nodo es pot veure a l'apartat 6.2.2Nodo
RequestMapping:
[urlServidor]/rest/busqueda/oficinas
Exemple:
<pre>http://localhost:8080/dir3caib/rest/busqueda/oficinas? codigo=&denominacion=&codNivelAdministracion=3&codComunidadAutonoma=4&pr ovincia=7&localidad=407-01&oficinasSir=true&vigentes=true</pre>

En aquest enllaç es pot trobar els valors estandars per el «directorio común» que s'han d'indicar als paràmetres.

<http://administracionelectronica.gob.es/ctt/dir3/descargas> a l'apartat de «Catálogos».

La resta de valors que no es trobin es poden trobar al «manual de Atributos- pg 62» al mateix enllaç a l'apartat «Manuales de Usuario».

Nota: als codis se'ls ha de llevar el 0 de davant.

Nota2: El codi de la localidad està format per codigoLocalidad (sense 0 inicial) – Cod.Entidad

Exemple Localidad:

Localidad: Palma (407-01)

6.1.7.- unidadDenominacion

Prototipus:
<pre>public @ResponseBody ResponseEntity<String> unidadDenominacion(@RequestParam String codigo) throws Exception;</pre>
Descripció:
Aquest servei permet obtenir la denominació d'una unitat a partir del seu codi dir3.
Paràmetres:
<ul style="list-style-type: none">String codigo: codi de la unitat que cercam.
Resposta:
<ul style="list-style-type: none">String: la denominació de la unitat.

6.1.8.- oficinaDenominacion

Prototipus:
<pre>public @ResponseBody ResponseEntity<String> oficinaDenominacion(@RequestParam String codigo) throws Exception;</pre>
Descripció:
Aquest servei permet obtenir la denominació d'una oficina a partir del seu codi dir3.
Paràmetres:
<ul style="list-style-type: none">String codigo: codi de l'oficina que cercam.
Resposta:
<ul style="list-style-type: none">String: la denominació de l'oficina.

6.1.9.- organigrama

Prototipus:
<pre>Public @ResponseBody ResponseEntity<Nodo> organigrama(@RequestParam String codigo) throws Exception;</pre>
Descripció:
Aquest servei obté l'arbre d'organismes/unitats del codi indicat pero la estructura que monta es un conjunt d'informació reduïda. La classe Nodo que es retorna conté un subconjunt d'informació de l'organisme.
Paràmetres:
String codigo: codi de l'organisme arrel del que volem obtenir el seu arbre
Resposta:
<ul style="list-style-type: none">• Llista <Nodo>, la descripció del bean Nodo es pot veure a l'apartat 6.2.2Nodo
RequestMapping:
[urlServidor]/rest/organigrama

6.1.10.- obtenerUnidadesDenominacionComunidad

Prototipus:
<pre>Public @ResponseBody ResponseEntity<List<Nodo>> obtenerUnidadesDenominacionComunidad(@RequestParam String denominacion, @RequestParam Long codComunidad) throws Exception;</pre>
Descripció:
Aquest servei permet cercar unitats per denominació i comunitat autònoma
Paràmetres:
<ul style="list-style-type: none">String denominacion: denominació per la que ens interessa cercarString codComunidad: codi de la comunitat autònoma a la que volem cercar.
Resposta:
<ul style="list-style-type: none">• Llista <Nodo>, la descripció del bean Nodo es pot veure a l'apartat 6.2.2Nodo
RequestMapping:
[urlServidor]/rest/busqueda/unidades/denominacion/comunidad

6.1.11.- ambitosTerritoriales

Prototipus:
<pre>public @ResponseBody ResponseEntity<List<CodigoValor>> ambitosTerritoriales(@RequestParam Long id) throws Exception;</pre>

Descripció:
Aquest servei permet obtenir els àmbits territorials del nivell d'administració indicat
Paràmetres:
Long id: identificador del nivell d'administració
Resposta:
<ul style="list-style-type: none">Llista <CodigoValor>, la descripció del bean CodigoValor es pot veure a l'apartat 6.2.3CodigoValor
RequestMapping:
[urlServidor]/rest/catalogo/ambitosTerritoriales

6.1.12.- comunidadesAutonomas

Prototipus:
<pre>public @ResponseBody ResponseEntity<List<CodigoValor>> comunidadesAutonomas () throws Exception;</pre>
Descripció:
Aquest servei permet obtenir totes les comunitats autònomes
Paràmetres:
Resposta:
<ul style="list-style-type: none">Llista <CodigoValor>, la descripció del bean CodigoValor es pot veure a l'apartat 6.2.3CodigoValor
RequestMapping:
[urlServidor]/rest/catalogo/comunidadesAutonomas

6.1.13.- entidadesGeograficas

Prototipus:
<pre>public @ResponseBody ResponseEntity<List<CodigoValor>> entidadesGeograficas () throws Exception;</pre>
Descripció:
Aquest servei permet obtenir totes les entitats geogràfiques
Paràmetres:

Resposta:
<ul style="list-style-type: none">Llista <CodigoValor>, la descripció del bean CodigoValor es pot veure a l'apartat 6.2.3CodigoValor
RequestMapping:
[urlServidor]/rest/catalogo/entidadesGeograficas

6.1.14.- provincias

Prototipus:
<pre>public @ResponseBody ResponseEntity<List<CodigoValor>> provincias() throws Exception;</pre>
Descripció:
Aquest servei permet obtenir totes les provincies
Paràmetres:
Resposta:
<ul style="list-style-type: none">Llista <CodigoValor>, la descripció del bean CodigoValor es pot veure a l'apartat 6.2.3CodigoValor
RequestMapping:
[urlServidor]/rest/catalogo/provincias

6.1.15.- provinciasCA

Prototipus:
<pre>public @ResponseBody ResponseEntity<List<CodigoValor>> provinciasCA(@RequestParam Long id) throws Exception;</pre>
Descripció:
Aquest servei permet obtenir totes les provincies de la comunitat autònoma indicada per paràmetre.
Paràmetres:
Long id: codi de la comunitat autònoma
Resposta:
<ul style="list-style-type: none">Llista <CodigoValor>, la descripció del bean CodigoValor es pot veure a l'apartat 6.2.3CodigoValor

RequestMapping:

[urlServidor]/rest/catalogo/provincias/comunidadAutonoma

6.1.16.- localidades

Prototipus:

```
public @ResponseBody ResponseEntity<List<CodigoValor>>  
localidades(@RequestParam Long codigoProvincia, String codigoEntidadGeografica)  
throws Exception;
```

Descripció:

Aquest servei permet obtenir totes les localitats d'una província i d'una entitat geogràfica especificades.

Paràmetres:

- Long codigoProvincia: codi de la província
- String codigoEntidadGeografica: codi de l'entitat geogràfica.

Resposta:

- Llista <CodigoValor>, la descripció del bean CodigoValor es pot veure a l'apartat 6.2.3CodigoValor

RequestMapping:

[urlServidor]/rest/catalogo/localidades/provincia/entidadGeografica

6.1.17.- nivelesAdministracion

Prototipus:

```
public @ResponseBody ResponseEntity<List<CodigoValor>> nivelesAdministracion()  
throws Exception;
```

Descripció:

Aquest servei permet obtenir els nivells d'administració

Paràmetres:**Resposta:**

- Llista <CodigoValor>, la descripció del bean CodigoValor es pot veure a l'apartat 6.2.3CodigoValor

RequestMapping:

```
[urlServidor]/rest/catalogo/nivelesAdministracion
```

6.2.- Model de dades dels serveis REST

6.2.1.- ObjetoDirectorio

Objecte que representa la tupla (codigo-denominació) d'un objecte unitat o oficina

Camp	Tipus	Cardinalitat	Descripció
codigo	String	1	Codi de l'objecte
denominacion	String	1	Denominació de l'objecte

6.2.2.- Nodo

Nodo pot representar una unitat o una oficina. Conté un subconjunt d'informació d'aquests objectes i en funció de l'objecte que representa s'omplen uns camps o altres.

Camp	Tipus	Cardinalitat	Descripció
codigo	String	1	Codi de l'objecte que representa.
denominacion	String	1	Denominació de l'objecte que representa.
descripcionEstado	String	1	Descripció de l'estat de l'objecte.
raiz	String	0..1	Representa l'element arrel de l'objecte..
superior	String	0..1	Representa l'element superior de l'objecte.
localidad	String	0..1	Representa la localitat de l'objecte.
idPadre	String	0..1	Identificador del pare de l'objecte
hijos	List<Nodo>	0..N	Llista de fills de l'objecte.
oficinasDependientes	List<Nodo>	0..N	Llista d'oficines que depenen de l'objecte
oficinasDependientesAuxiliares	List<Nodo>	0..N	Llista d'oficines auxiliars depenents de l'objecte.
oficinasAuxiliares	List<Nodo>	0..N	Llista d'oficines auxiliars de l'objecte

oficinasFuncionales	List<Nodo>	0..N	Llista d'oficines funcionals de l'objecte quan representa una unitat.
---------------------	------------	------	---

6.2.3.- CodigoValor

CodigoValor es una tupla que representa (id-descripció) dels objectes del catàleg.

Camp	Tipus	Cardinalitat	Descripció
id	Object	1	Codi de l'objecte que representa.
descripcion	String	1	Descripció de l'objecte que representa.

6.3.- Exemple

A continuació es mostra un exemple d'una cridada REST al mètode *busquedaOrganismos* mitjançant ajax:

```
$.ajax({
  async: false,
  crossDomain: true,
  url: "http://localhost:8080/dir3caib/rest/busqueda/organismos",
  type: 'GET',
  dataType: 'json',
  data: { codigo: codigoOrganismo,
    denominacion: denominacion,
    codNivelAdministracion: codNivelAdministracion,
    codComunidadAutonoma: codComunidadAutonoma,
    conOficinas: conOficinas,
    unidadRaiz: unidadRaiz,
    provincia: provincia,
    localidad: localidad,
    oficinasSir: oficinasSir,
    vigentes: vigentes},

  success: function(result) {
    var len = result.length;
    for ( var i = 0; i < len; i++) {
      var denominacion = result[i].denominacion;
      var codigo = result[i].codigo;
      alert(codigo + "-" + denominacion);
    }
  }
});
```

7.- Wsdl

7.1.- Catàleg

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

<wsdl:definitions name="Dir3CaibObtenerCatalogosWsService"
targetNamespace="http://www.caib.es/dir3caib"
xmlns:ns1="http://catalogo.ws.dir3caib.caib.es/"
xmlns:ns2="http://schemas.xmlsoap.org/wsdl/soap/http"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://www.caib.es/dir3caib"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">

  <wsdl:import
location="http://ibit157:8080/dir3caib/ws/Dir3CaibObtenerCatalogos?
wsdl=Dir3CaibObtenerCatalogosWs.wsdl"
namespace="http://catalogo.ws.dir3caib.caib.es/">

    </wsdl:import>

  <wsdl:binding name="Dir3CaibObtenerCatalogosWsServiceSoapBinding"
type="ns1:Dir3CaibObtenerCatalogosWs">

    <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>

    <wsdl:operation name="obtenerCatComunidadAutonoma">

      <soap:operation soapAction="" style="document"/>

      <wsdl:input name="obtenerCatComunidadAutonoma">

        <soap:body use="literal"/>

      </wsdl:input>

      <wsdl:output name="obtenerCatComunidadAutonomaResponse">

        <soap:body use="literal"/>

      </wsdl:output>

    </wsdl:operation>

    <wsdl:operation name="obtenerCatEntidadGeografica">

      <soap:operation soapAction="" style="document"/>

      <wsdl:input name="obtenerCatEntidadGeografica">

        <soap:body use="literal"/>

      </wsdl:input>

      <wsdl:output name="obtenerCatEntidadGeograficaResponse">

        <soap:body use="literal"/>

      </wsdl:output>

    </wsdl:operation>

    <wsdl:operation name="obtenerCatProvincia">

      <soap:operation soapAction="" style="document"/>

      <wsdl:input name="obtenerCatProvincia">
```

```
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output name="obtenerCatProvinciaResponse">
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="obtenerCatLocalidad">
    <soap:operation soapAction="" style="document"/>
    <wsdl:input name="obtenerCatLocalidad">
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output name="obtenerCatLocalidadResponse">
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="obtenerCatNivelAdministracion">
    <soap:operation soapAction="" style="document"/>
    <wsdl:input name="obtenerCatNivelAdministracion">
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output name="obtenerCatNivelAdministracionResponse">
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="obtenerCatEstadoEntidad">
    <soap:operation soapAction="" style="document"/>
    <wsdl:input name="obtenerCatEstadoEntidad">
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output name="obtenerCatEstadoEntidadResponse">
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="obtenerCatPais">
    <soap:operation soapAction="" style="document"/>
    <wsdl:input name="obtenerCatPais">
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output name="obtenerCatPaisResponse">
```

```
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="getVersionWs">
    <soap:operation soapAction="" style="document"/>
    <wsdl:input name="getVersionWs">
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output name="getVersionWsResponse">
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="getVersion">
    <soap:operation soapAction="" style="document"/>
    <wsdl:input name="getVersion">
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output name="getVersionResponse">
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="Dir3CaibObtenerCatalogosWsService">
    <wsdl:port binding="tns:Dir3CaibObtenerCatalogosWsServiceSoapBinding"
name="Dir3CaibObtenerCatalogosWs">
        <soap:address
location="http://10.215.216.178:8080/dir3caib/ws/Dir3CaibObtenerCatalogos"/>
    </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

7.2.- Unitats

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions name="Dir3CaibObtenerUnidadesWsService"
targetNamespace="http://www.caib.es/dir3caib"
xmlns:ns1="http://unidad.ws.dir3caib.caib.es/"
xmlns:ns2="http://schemas.xmlsoap.org/wsdl/soap/http"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://www.caib.es/dir3caib"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
```

```
<wsdl:import
location="http://ibit157:8080/dir3caib/ws/Dir3CaibObtenerUnidades?
wsdl=Dir3CaibObtenerUnidadesWs.wsdl"
namespace="http://unidad.ws.dir3caib.caib.es/">

</wsdl:import>

<wsdl:binding name="Dir3CaibObtenerUnidadesWsServiceSoapBinding"
type="ns1:Dir3CaibObtenerUnidadesWs">

<soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>

<wsdl:operation name="getVersionWs">

<soap:operation soapAction="" style="document"/>

<wsdl:input name="getVersionWs">

<soap:body use="literal"/>

</wsdl:input>

<wsdl:output name="getVersionWsResponse">

<soap:body use="literal"/>

</wsdl:output>

</wsdl:operation>

<wsdl:operation name="getVersion">

<soap:operation soapAction="" style="document"/>

<wsdl:input name="getVersion">

<soap:body use="literal"/>

</wsdl:input>

<wsdl:output name="getVersionResponse">

<soap:body use="literal"/>

</wsdl:output>

</wsdl:operation>

<wsdl:operation name="obtenerUnidad">

<soap:operation soapAction="" style="document"/>

<wsdl:input name="obtenerUnidad">

<soap:body use="literal"/>

</wsdl:input>

<wsdl:output name="obtenerUnidadResponse">

<soap:body use="literal"/>

</wsdl:output>

</wsdl:operation>

<wsdl:operation name="obtenerArbolUnidades">

<soap:operation soapAction="" style="document"/>

<wsdl:input name="obtenerArbolUnidades">

<soap:body use="literal"/>
```

```
</wsdl:input>
<wsdl:output name="obtenerArbolUnidadesResponse">
  <soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="Dir3CaibObtenerUnidadesWsService">
  <wsdl:port binding="tns:Dir3CaibObtenerUnidadesWsServiceSoapBinding"
name="Dir3CaibObtenerUnidadesWs">
    <soap:address
location="http://10.215.216.178:8080/dir3caib/ws/Dir3CaibObtenerUnidades"/>
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

7.3.- Oficines

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions name="Dir3CaibObtenerOficinasWsService"
targetNamespace="http://www.caib.es/dir3caib"
xmlns:ns1="http://oficina.ws.dir3caib.caib.es/"
xmlns:ns2="http://schemas.xmlsoap.org/wsdl/soap/http"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://www.caib.es/dir3caib"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:xsd="http://www.w3.org/2001/
XMLSchema">
  <wsdl:import location="http://ibit157:8080/dir3caib/ws/Dir3CaibObtenerOficinas?
wsdl=Dir3CaibObtenerOficinasWs.wsdl"
namespace="http://oficina.ws.dir3caib.caib.es/">
  </wsdl:import>
  <wsdl:binding name="Dir3CaibObtenerOficinasWsServiceSoapBinding"
type="ns1:Dir3CaibObtenerOficinasWs"><soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="getVersionWs">
      <soap:operation soapAction="" style="document"/>
      <wsdl:input name="getVersionWs">
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output name="getVersionWsResponse">
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="obtenerOficinasSIRUnidad">
```



```
<soap:operation soapAction="" style="document"/>
<wsdl:input name="obtenerOficinasSIRUnidad">
  <soap:body use="literal"/>
</wsdl:input>
<wsdl:output name="obtenerOficinasSIRUnidadResponse">
  <soap:body use="literal"/>
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="getVersion">
  <soap:operation soapAction="" style="document"/>
  <wsdl:input name="getVersion">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="getVersionResponse">
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="obtenerArbolOficinas">
  <soap:operation soapAction="" style="document"/>
  <wsdl:input name="obtenerArbolOficinas">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="obtenerArbolOficinasResponse">
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="obtenerOficina">
  <soap:operation soapAction="" style="document"/>
  <wsdl:input name="obtenerOficina">
    <soap:body use="literal"/>
  </wsdl:input>
  <wsdl:output name="obtenerOficinaResponse">
    <soap:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
</wsdl:binding>

<wsdl:service name="Dir3CaibObtenerOficinasWsService"><wsdl:port
binding="tns:Dir3CaibObtenerOficinasWsServiceSoapBinding"
name="Dir3CaibObtenerOficinasWs">
```

```
<soap:address
location="http://10.215.216.178:8080/dir3caib/ws/Dir3CaibObtenerOficinas"/>
</wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

8.- Procés Sincronització Organigrama (DIR3CAIB-REGWEB)

Per obtenir els canvis que s'han produït dins l'organigrama en Dir3caib s'empra el mètode 3.1obtenerArbolUnidades que retorna la llista d'Unitats actualitzades si s'han especificat unes dates. Si no s'especifiquen dates, retorna tot l'organigrama vigent.

Aquest llistat consta de la llista d'UnidadTF (Classe UnidadTF) que una de les propietats que té és el conjunt d'històrics de la UnidadTF.

A continuació posarem exemples de com funcionen els històrics de DIRCAIB:

8.1.- Exemple 1

Suposem que tenim el següent organigrama

- A
 - B
 - B1
 - B2
 - C
 - D

Els canvis que es produeixen són els següents:

C és substituït per C1

C.getHistoricoUO() = C1

D es divideix en D1 i D2

D.getHistoricoUO() = D1, D2

B1 i B2 es fusionen en B3.

B1.getHistoricoUO() = B3

B2.getHistoricoUO() = B3

Per tant els històrics d'una unitat sempre ens diuen qui el substitueix/substitueixen. (tal vegada s'hauria d'haver dit organosSustituyentes en lloc d'Històrics.)

L'organigrama resultant seria aquest:

Organigrama inicial	Organigrama resultant
<ul style="list-style-type: none">• A<ul style="list-style-type: none">◦ B<ul style="list-style-type: none">▪ B1▪ B2◦ C◦ D	<ul style="list-style-type: none">• A<ul style="list-style-type: none">◦ B<ul style="list-style-type: none">▪ B3◦ C1◦ D1◦ D2

8.2.- Exemple 2 (múltiples fusions divisions)

D1 es divideix en D1.1, D1.2, D1.3 i abans de que actualitzeu D1.2 i D1.3 es fusionen en D1.4

L'Organigrama passa per una fase intermitja

Organigrama Inicial	Organigrama intermig	Organigrama Final
<ul style="list-style-type: none">• A<ul style="list-style-type: none">◦ B<ul style="list-style-type: none">▪ B3◦ C1◦ D1◦ D2	<ul style="list-style-type: none">• A<ul style="list-style-type: none">◦ B<ul style="list-style-type: none">▪ B3◦ C1◦ D1.1◦ D1.2◦ D1.3◦ D2	<ul style="list-style-type: none">• A<ul style="list-style-type: none">◦ B<ul style="list-style-type: none">▪ B3◦ C1◦ D1.1◦ D1.4◦ D2

En aquest cas haureu d'obtenir els històrics de primer nivell de D1 i després per cada un dels retornats heu de continuar demanant històrics fins que teneu els darrers, o vos retorni buit(null).

Si no s'itera fins al final, es quedaria amb l'organigrama intermig.

D1. getHistoricosUO() = D1.1, D1.2, D1.3

D1.1.getHistoricosUO= null

D1.2.getHistoricosUO()= D1.4

D1.3.getHistoricosUO() = D1.4

Aquest funcionament està confirmat de dia 01/06/2018. Proves fetes de na Marilen a l'entorn de PRE de Madrid.

8.3.- Classes que implementen aquesta funcionalitat

8.3.1.- ObtenerUnidadesEjb.java

<https://github.com/GovernIB/dir3caib/blob/master/persistence/src/main/java/es/caib/dir3caib/persistence/ejb/ObtenerUnidadesEjb.java>

```
public List<UnidadTF> obtenerArbolUnidadesTF(String codigo, Date  
fechaActualizacion, Date fechaSincronizacion) throws Exception {  
}
```

En el cas de Regweb es poden consultar les classes:

8.3.2.- SincronizadorDir3Bean.java

Aquestes classe es la que processa la llista d'UnitatsTF que rep de Dir3caib.

<https://github.com/GovernIB/registre/blob/master/persistence/src/main/java/es/caib/regweb3/persistence/ejb/SincronizadorDir3Bean.java> (tota la classe en general).