



**TeleMadrid**

## **ANEXO - 03**

### **SERVICIO DE BÚSQUEDAS**

## ÍNDICE

<b>SERVICIO DE BÚSQUEDA.....</b>	<b>3</b>
<b>ARQUITECTURA DE LA SOLUCIÓN .....</b>	<b>3</b>
<b>SERVICIO DE BÚSQUEDA.....</b>	<b>4</b>
DEFINICIÓN FUNCIONAL.....	4
Tipos de búsqueda .....	5
Campos de búsqueda.....	5
<b>DISEÑO DEL SERVICIO .....</b>	<b>9</b>
SERVICIO XMLINDEXSEARCH.....	9
Definición de los esquemas de los ficheros XML entrada y salida.....	10
Registro de operaciones.....	26
<b>INDEXACIÓN EN EXALEAD.....</b>	<b>28</b>
DEFINICIÓN DEL ÍNDICE .....	28
Objeto Contenido .....	29
Objeto Minutaje .....	30
Objeto MediaVideo .....	30
DEFINICIÓN DE CONECTORES .....	31
DEFINICIÓN DE LA INDEXACIÓN .....	31
Mapeo del conector de Contenidos .....	32
Mapeo del conector de Sub-contenidos .....	33
Mapeo del conector de Minutajes .....	35
Mapeo campos del conector de MediaVideos .....	36
<b>ANEXO: INVOCACIÓN SÍNCRONA Y ASÍNCRONA DEL SERVICIO .....</b>	<b>37</b>
DEFINICIÓN DEL SERVICIO WEB INDEXSEARCH .....	37
INVOCACIÓN AL SERVICIO WEB INDEXSEARCH DE FORMA SÍNCRONA .....	37
Referencias al servicio .....	37
Tarea Principal .....	37
Ejemplo de Invocación síncrona .....	38
INVOCACIÓN AL SERVICIO WEB INDEXSEARCH DE FORMA ASÍNCRONA .....	38
Referencias al servicio .....	39
Tarea Principal .....	39
Función de Callback .....	39
Ejemplo de Invocación asíncrona .....	40

## SERVICIO DE BÚSQUEDA

### ARQUITECTURA DE LA SOLUCIÓN

El siguiente diagrama muestra la arquitectura de la solución

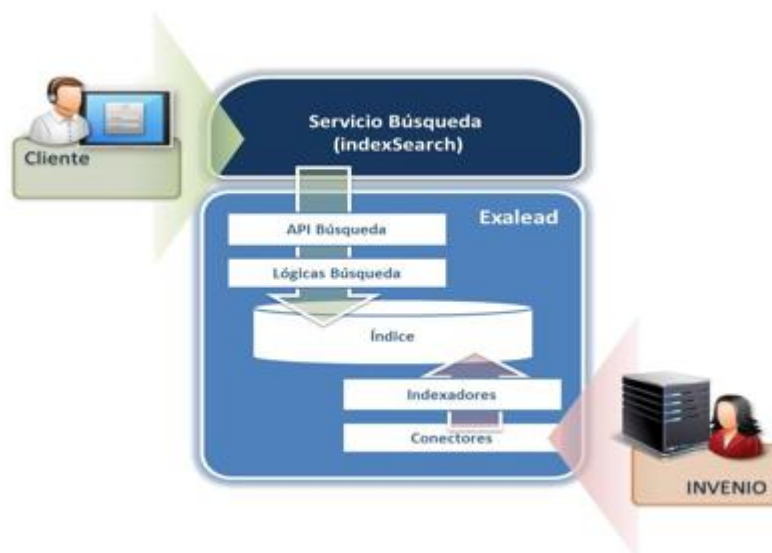


Ilustración 1: Arquitectura de la solución

Vemos con detalle cada uno de sus componentes:

- Aplicación Cliente:** Aplicación que hace uso del servicio web de búsqueda, esta aplicación invocará al servicio pasándole los parámetros y como resultado obtendrá un objeto con los resultados obtenidos sacados del índice de Exalead, dado que el servicio **xmlIndexSearch** es publicado como servicio web, la aplicación cliente no tiene por qué estar basada en una tecnología .NET y podríamos tener otros clientes desarrollados por ejemplo en tecnología Java
  - SONAPS o HIVE:** Ejemplo de aplicación cliente que ofrece un buscador de contenidos y sub-contenidos a los usuarios, a los que se les presentará un formulario con una serie de campos para que realicen la búsqueda y se les mostrará el listado de los contenidos y sub-contenidos que la cumplen.
- Servicio de búsqueda (xmlIndexSearch):** Servicio objeto de la solución, funciona como intermediario entre los clientes y el índice de Exalead, transformando los parámetros recibidos en la Consulta esperada por Exalead y los registros obtenidos del índice en los contenidos y sub-contenidos devueltos.  
El servicio estará desarrollado en .NET utilizando el Framework 3.5
- INVENIO:** Aplicación donde se registra toda la información sobre los contenidos y minutajes de la cadena, de toda la información almacenada en INVENIO para el caso que nos ocupa nos interesan los siguientes elementos:
  - Contenidos:** Cada uno de los elementos almacenados (Por ejemplo, el informativo de mediodía del 21/05/2015)
  - Sub-Contenidos o Minutajes:** Divisiones del contenido (Por ejemplo, una noticia del informativo donde se habla de las elecciones municipales)
  - MediaVideo:** Formatos en los que se almacena un contenido (Por ejemplo, si el video está en alta definición)
 La aplicación almacena su información en una base de datos relacional
- Exalead:** Motor de indexación y búsqueda, está formado por los siguientes componentes:

- **Conectores:** Elementos encargados de extraer la información a indexar, en este caso se tratará de conectores SQL que acceden a la base de datos de INVENIO
- **Indexadores:** Elementos que transforman la información obtenida por los conectores y la almacenan en el índice de la herramienta
- **Índice:** Repositorio donde se almacenan los registros
- **Lógicas de búsqueda:** Componente que define como van a ejecutarse las búsquedas dentro del índice.
- **API de búsqueda:** Componente que publica los servicios de búsqueda que van a ser utilizados desde *xmlindexSearch* para recuperar los elementos, este componente hace uso de las lógicas de búsqueda para saber exactamente como debe acceder al índice para obtener los resultados

## SERVICIO DE BÚSQUEDA

### DEFINICIÓN FUNCIONAL

El objetivo del servicio de búsqueda es el de permitir que aplicaciones cliente realicen consultas sobre los contenidos registrados en INVENIO a través del motor de indexación y búsqueda de Exalead.

De esta forma las aplicaciones pueden ofrecer a sus usuarios la posibilidad de buscar contenidos usando la potencia y flexibilidad de este tipo de buscadores sin tener que conocer el funcionamiento interno de este tipo de tecnologías de búsqueda. Para el cliente la búsqueda no deja de ser una búsqueda normal, en la que pasamos un filtro basado en una serie de parámetros por los que buscar y unas condiciones entre dichos parámetros y recupera como resultado los contenidos que cumplen con dicho filtro.

El uso un buscador basado en índices como es **Exalead** ofrece una serie de ventajas sobre las búsquedas habituales basadas en SQL:

- **Búsqueda en información desestructurada:** A diferencia de las bases de datos donde toda la información está acotada a tablas y campos, Exalead permite indexar por ejemplo documentos y realizar búsquedas sobre sus metadatos o contenidos, de la misma forma podemos indexar bases de datos, páginas web, etc.
- **Rapidez:** La búsqueda indexada es mucho más rápida que la búsqueda SQL, está basada en la generación de un índice que se construye precisamente pensando en optimizar las búsquedas, aunque el número de elementos sea muy alto.
- **Carga de trabajo:** La búsqueda indexada permite liberar de trabajo a los sistemas finales, una vez que la información es indexada, el sistema ya no debe acceder a la fuente de la información (Base de datos) y buscará únicamente sobre su índice, esto va a permitir que los sistemas finales se vean liberados de la carga de las consultas de los usuarios.
- **Búsqueda inteligente:** Ampliando los operadores de búsqueda a los que estamos acostumbrados en SQL, ahora podremos buscar palabras sin tener que escribirlas exactamente, obviar las tildes, los plurales, las familias de palabras, etc.

En definitiva, lo que buscamos con este servicio es que los usuarios puedan acceder a los contenidos registrados en **INVENIO** como si lo estuvieran haciendo a través de un buscador de tipo **Google**

Como el servicio es publicado como un servicio web su uso es simple e independiente de la tecnología utilizada en su desarrollo, una vez programado el servicio, se generará el fichero WSDL que permite que cualquier aplicación genere un cliente web para este servicio y lo invoque como si estuviese invocando a cualquier método de una clase.

Dado que se desea hacer un uso síncrono y asíncrono del servicio, publicaremos el servicio como síncrono y dejaremos que sea el cliente el que decide si hacer uso de él de forma síncrona o asíncrona, podemos ver con más detalle cómo se lleva a cabo este tipo de invocaciones en el [ANEXO: INVOCACIÓN SÍNCRONA Y ASÍNCRONA DEL SERVICIO](#)

Para hacer un seguimiento más sencillo de las ejecuciones del servicio, este generará dos tipos de registro o log, un **registro de operación** donde se registrarán las operaciones que va ejecutando el servicio incluyendo los errores y un **registro de auditoría** donde se guardarán las consultas que se van solicitando. Los registros de log se van a almacenar en un esquema de Base de Datos Microsoft SQL Server.

## TIPOS DE BÚSQUEDA

---

El servicio tendrá tres modalidades de búsqueda:

- **Full Search:** Permite realizar búsquedas de tipo Google, el usuario introduce el valor a buscar y el sistema busca este valor en una serie de campos del índice previamente definidos.
- **Quick Search:** Permite hacer una búsqueda sobre un conjunto limitado de los campos del índice, está orientado a la presentación de un formulario de búsqueda sencillo.
- **Advanced Search:** Permite hacer una búsqueda sobre un conjunto más amplio de los campos del índice, está orientado a la presentación de un formulario de búsqueda más complejo, con más campos y opciones.

En cualquiera de los casos la llamada al servicio siempre se ejecuta de la misma forma, como veremos más adelante en el diseño del servicio, el tipo de búsqueda es un parámetro del mismo.

## CAMPOS DE BÚSQUEDA

---

La aplicación va a permitir buscar por los siguientes campos:

- **Id:** Identificador del contenido, la búsqueda por este campo es tipo “contiene”, es decir se devolverá aquellos registros que tengan parte del identificador. Por defecto en Exalead tiene que tener un mínimo de tres caracteres para hacer efectiva la búsqueda.
- **KeyDate:** Fecha de creación del contenido, nos permitirá consultar elementos antes de la fecha [<date o <=date], después de la fecha [>date o >=date], igual a la fecha [=date] y entre dos fechas [>dateA y <dateB]
- **Description:** Descripción larga del contenido o sub-contenido. La búsqueda es de tipo “contiene”. Es decir, buscaremos que la palabra o frase introducida se encuentren dentro del valor almacenado en el campo o Por ejemplo si la descripción almacenada es **Imágenes de la manifestación por los derechos de la mujer en la plaza dos de mayo** y el usuario busca **plaza ó plaza dos** el buscador devolverá el registro.
- **Notes:** Derechos de uso del contenido. La búsqueda es de tipo “contiene”
- **Title:** Título del contenido o sub-contenido. La búsqueda es de tipo “contiene”
- **SecondaryTitle:** Título del contenido en castellano. La búsqueda es de tipo “contiene”
- **Forms:** Ámbito de búsqueda del contenido o sub-contenido. La búsqueda es de tipo “contiene”
- **contentDescriptorsImagen:** Descripción del contenido o sub-contenido de la imagen. En este caso el índice almacena en un campo multivaluado todos los valores del Thesaurus asociados al registro. Al igual que en otros campos de texto, la búsqueda será de tipo “contiene”, aunque ahora el buscador devolverá el registro si el texto introducido por el usuario está contenido en uno de los valores del Thesaurus almacenados

Por ejemplo, un registro posee 3 valores en este campo

**Estación de esquí Navacerrada, Puerto de Navacerrada, Comunidad de Madrid**

Si el usuario buscara **esquí navacerrada**, este registro sería devuelto

- **contentDescriptorsMateria:** Descripción del contenido o sub-contenido de la materia. Su uso es igual al de contentDescriptorsImagen
- **locationDescriptorsImagen:** Localización que aparece en la imagen. Su uso es igual al de contentDescriptorsImagen
- **locationDescriptorsNoticia:** Localización que aparece en la noticia. Su uso es igual al de contentDescriptorsImagen

- **personalitiesDescriptorsImagen:** Persona que aparece en la imagen. Su uso es igual al de contentDescriptorsImagen
- **personalitiesDescriptorsMateria:** Persona que aparece en la imagen. Su uso es igual al de contentDescriptorsImagen
- **contentRelatedCompaniesDescriptorsDerechos:** Derechos del contenido. Su uso es igual al de contentDescriptorsImagen
- **Creador:** La búsqueda es de tipo “contiene”
- **Audio:** La búsqueda es de tipo “contiene”
- **NumeroCapitulo:** La búsqueda es de tipo “contiene”
- **Formavideoclips:** La búsqueda es de tipo “contiene”
- **IndexingStatus:** La búsqueda es de tipo “contiene”
- **Candidate:** La búsqueda es de tipo “contiene”
- **Property:** La búsqueda es de tipo “contiene”
- **Task:** La búsqueda es de tipo “contiene”
- **RegionsOfRights:** La búsqueda es de tipo “contiene”
- **Language:** La búsqueda es de tipo “contiene”
- **ContentDescriptorGeneromusical:** Su uso es igual al de contentDescriptorsImagen
- **ContentDescriptorEstilomusical:** Su uso es igual al de contentDescriptorsImagen
- **LocationDescriptorMateria;** Su uso es igual al de contentDescriptorsImagen
- **LocationDescriptorPaisMusico:** Su uso es igual al de contentDescriptorsImagen
- **LocationDescriptorPaisProduccion:** Su uso es igual al de contentDescriptorsImagen
- **EventDescriptorAnoProduccion;** Su uso es igual al de contentDescriptorsImagen
- **EventDescriptorTemporada:** Su uso es igual al de contentDescriptorsImagen
- **CompaniesDescriptorDistribuidora:** Su uso es igual al de contentDescriptorsImagen
- **CompaniesDescriptorProductora:** Su uso es igual al de contentDescriptorsImagen
- **Credits\_Comentarista:** Su uso es igual al de contentDescriptorsImagen
- **Credits\_CreadorMusica:** Su uso es igual al de contentDescriptorsImagen
- **Credits\_Director:** Su uso es igual al de contentDescriptorsImagen
- **Credits\_DirectorFotografia:** Su uso es igual al de contentDescriptorsImagen
- **Credits\_Documentalista:** Su uso es igual al de contentDescriptorsImagen
- **Credits\_DocumentalistaPrograma:** Su uso es igual al de contentDescriptorsImagen
- **Credits\_Editor:** Su uso es igual al de contentDescriptorsImagen
- **Credits\_Guionista:** Su uso es igual al de contentDescriptorsImagen
- **Credits\_Interprete:** Su uso es igual al de contentDescriptorsImagen
- **Credits\_Presentador:** Su uso es igual al de contentDescriptorsImagen
- **Credits\_Productor:** Su uso es igual al de contentDescriptorsImagen
- **Credits\_Realizador;** Su uso es igual al de contentDescriptorsImagen

La siguiente tabla muestra el mapeo entre los nombres de los campos de búsqueda, el campo por el que se buscaría en **INVENIO** en función de si la búsqueda es de contenido o de sub-contenido y el campo donde se almacena la información dentro del índice de Exalead

CAMPO DE BÚSQUEDA	CONTENIDO	SUBCONTENIDO	MODELO EXALEAD
ID	IDContent		contenido_id
		IDSubcontent	contenido_idsubcontenido
CreationDate	CreationDate	CreationDate	contenido_creationdate
KeyDate	KeyDate	KeyDate	contenido_keydate
AssetOrigin	AssetOrigin	AssetOrigin	contenido_assetorigin
LongDescription	LongDescription	Description	contenido_description
Notes	Notes	Notes	contenido_notes
Title	Title	Title	contenido_title
SecondaryTitle	SecondaryTitle	SecondaryTitle	contenido_secondarytitle

Tabla 1: Relación entre campos de búsqueda, campos de INVENIO y campos del índice Exalead

Como vemos en la tabla el índice de Exalead nos permite normalizar el modelo de datos de INVENIO, almacenando en un único elemento información de dos distintos orígenes (contenido y sub-contenido), podemos ver con detalle el contenido del índice en el punto [Definición de la indexación](#)

A continuación, adjuntamos las queries para la recuperación de los datos

### Contenidos

```
SELECT IDContent, CreationDate, KeyDate, AssetOrigin, LongDescription, Notes, Title, SecondaryTitle, FORMS,
Template,
Creadopor, Audio, NumeroCapitulo, Formavideoclips, IndexingStatus, Candidate, Property, Task,
RegionsOfRights,
Language, ContentDescriptorImagen, ContentDescriptorMateria, ContentDescriptorGeneromusical,
ContentDescriptorEstilomusical, LocationDescriptorImagen, LocationDescriptorNoticia,
LocationDescriptorMateria,
LocationDescriptorPaisMusico, LocationDescriptorPaisProduccion, PersonalitiesDescriptorImagen,
PersonalitiesDescriptorMateria, EventDescriptorAnoProduccion, EventDescriptorTemporada,
CompaniesDescriptorDerechos, CompaniesDescriptorDistribuidora, CompaniesDescriptorProductora,
Credits_Comentarista, Credits_CreadorMusica, Credits_Director, Credits_DirectorFotografia,
Credits_Documentalista, Credits_DocumentalistaPrograma, Credits_Editor, Credits_Guionista,
Credits_Interprete, Credits_Presentador, Credits_Productor, Credits_Realizador
FROM Extraccion_CONTENT
```

### Subcontenidos

```
SELECT IDContent, IDSubcontent, CreationDate, KeyDate, AssetOrigin, Description, Notes, Title, SecondaryTitle,
FORMS
as SUBCONTENT_FORMS, Template, ContentDescriptorImagen, ContentDescriptorMateria,
ContentDescriptorGeneromusical, ContentDescriptorEstilomusical, LocationDescriptorImagen,
LocationDescriptorNoticia, LocationDescriptorMateria, LocationDescriptorPaisMusico,
LocationDescriptorPaisProduccion, PersonalitiesDescriptorImagen, PersonalitiesDescriptorMateria,
EventDescriptorAnoProduccion, EventDescriptorTemporada, CompaniesDescriptorDerechos,
CompaniesDescriptorDistribuidora, CompaniesDescriptorProductora, Credits_Comentarista,
Credits_CreadorMusica,
Credits_Director, Credits_DirectorFotografia, Credits_Documentalista, Credits_DocumentalistaPrograma,
Credits_Editor, Credits_Guionista, Credits_Interprete, Credits_Presentador, Credits_Productor,
Credits_Realizador
FROM Extraccion_SUBCONTENT
```

Cada servicio de búsqueda permite buscar sobre unos determinados campos, la siguiente tabla muestra en función del servicio que campos de búsqueda se pueden recibir y sobre que campos del índice se estará buscando.

Campo de búsqueda	advancedSearch	quickSearch	fullSearch
Id	Id	Id	Id
KeyDate	KeyDate	KeyDate	KeyDate
Description	Description	Description	Description

Notes	Notes		Description
Title	Title	Title	Description
SecondaryTitle	SecondaryTitle	SecondaryTitle	Description
FORMS	FORMS		Description
Template	Template		Description
Creadopor	Creadopor		Description
Audio	Audio		Description
NumeroCapitulo	NumeroCapitulo		Description
Formavideoclips	Formavideoclips		Description
IndexingStatus	IndexingStatus		Description
Candidate	Candidate		Description
Property	Property		Description
Task	Task		Description
RegionsOfRights	RegionsOfRights		Description
Language	Language		Description
ContentDescriptorImagen	ContentDescriptorImagen		Description
ContentDescriptorMateria	ContentDescriptorMateria		Description
ContentDescriptorGeneromusical	ContentDescriptorGeneromusical		Description
ContentDescriptorEstilomusical	ContentDescriptorEstilomusical		Description
LocationDescriptorImagen	LocationDescriptorImagen		Description
LocationDescriptorNoticia	LocationDescriptorNoticia		Description



Campo de búsqueda	advancedSearch	quickSearch	fullSearch
LocationDescriptorMateria	LocationDescriptorMateria		Description
LocationDescriptorPaisMusico	LocationDescriptorPaisMusico		Description
LocationDescriptorPaisProduccion	LocationDescriptorPaisProduccion		Description
PersonalitiesDescriptorImagen	PersonalitiesDescriptorImagen		Description
PersonalitiesDescriptorMateria	PersonalitiesDescriptorMateria		Description
EventDescriptorAnoProduccion	EventDescriptorAnoProduccion		Description
EventDescriptorTemporada	EventDescriptorTemporada		Description
CompaniesDescriptorDerechos	CompaniesDescriptorDerechos		Description
CompaniesDescriptorDistribuidora	CompaniesDescriptorDistribuidora		Description
CompaniesDescriptorProductora	CompaniesDescriptorProductora		Description
Credits_Comentarista	Credits_Comentarista		Description
Credits_CreadorMusica	Credits_CreadorMusica		Description
Credits_Director	Credits_Director		Description
Credits_DirectorFotografia	Credits_DirectorFotografia		Description
Credits_Documentalista	Credits_Documentalista		Description
Credits_DocumentalistaPrograma	Credits_DocumentalistaPrograma		Description
Credits_Editor	Credits_Editor		Description
Credits_Guionista	Credits_Guionista		Description
Credits_Interprete	Credits_Interprete		Description
Credits_Presentador	Credits_Presentador		Description
Credits_Productor	Credits_Productor		Description
Credits_Realizador	Credits_Realizador		Description
Fecha_Actualizacion	Fecha_Actualizacion		Description

Tabla 2: Uso de los campos de búsqueda por tipo de búsqueda

El resumen de esta tabla es el siguiente:

- En las búsquedas de tipo **advancedSearch** se pueden recibir todos los campos como parámetros de búsqueda y cada uno de ellos será buscado en su correspondiente campo del índice
- En las búsquedas de tipo **quickSearch** solo algunos campos son recibidos como parámetros (Id, KeyDate, Description, Title, SecondaryTitle) y cada uno de ellos será buscado en su correspondiente campo del índice
- En las búsquedas de tipo **fullSearch** los parámetros Id y KeyDate funcionan como en los casos anteriores, pero el parámetro Description en vez de buscarse únicamente en el campo Description del índice es buscado en más campos (Notes, Title, etc.)

En las búsquedas **fullSearch** los campos contra los que vamos a mapear el **Description** podrán modificarse. Los nombres de los campos a mapear se van a definir en fichero de configuración del Servicio Web. Ver documento "Instalación y Configuración Servicio Web".

## DISEÑO DEL SERVICIO

### SERVICIO XMLINDEXSEARCH

El servicio se denomina **xmlIndexSearch** y consta de los siguientes componentes:

- **Parámetros de entrada:**
  - **xmlRequest:** String con el contenido del fichero XML de entrada. El esquema que debe cumplir el fichero XML de entrada está definido en [request\\_schema\\_v5.0.xsd](#)

- **Parámetro de salida:** Como resultado de la consulta obtendremos un String con el contenido del fichero XML de salida. El esquema que debe cumplir el fichero XML de salida está definido en [response\\_schema\\_v5.0.xsd](#).

Cuando un cliente usa al servicio *xmlIndexSearch*, el servicio analizará el fichero XML pasado como parámetro de entrada, verificando que cumple con el esquema definido, ([request\\_schema\\_v5.0.xsd](#)) y extrayendo la información necesaria para generar la consulta (Query) en el lenguaje propio de Exalead. Con esta Query el servicio invocará al API de búsqueda de la herramienta para que busque en el índice donde se almacenaron los contenidos y sub-contenidos, una vez ejecutada la búsqueda el servicio transformará el resultado devuelto por Exalead en un fichero XML que cumple el esquema definido ([response\\_schema\\_v5.0.xsd](#)) y lo devolverá al cliente como un String.

## DEFINICIÓN DE LOS ESQUEMAS DE LOS FICHEROS XML ENTRADA Y SALIDA

---

A continuación, definiremos los esquemas de los ficheros XML de entrada y salida del servicio *xmlIndexSearch*

### REQUEST\_SCHEMA\_V5.0.XSD

---

```
<? xml version="1.0" encoding="UTF-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://DCMPlatform/v2.0" xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="BaseSearchEntity">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="DCMBaseSearchCondition">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="LoginInfo" maxOccurs="1" minOccurs="1">
              <xs:complexType>
                <xs:sequence>
                  <xs:element type="xs:string" name="Id_Peticion_SONAPS" minOccurs="1"/>
                  <xs:element type="xs:string" name="Sistema_Origen" minOccurs="1"/>
                  <xs:element type="xs:string" name="Nombre_Flujo" minOccurs="1"/>
                  <xs:element type="xs:string" name="UserToken" minOccurs="0"/>
                  <xs:element type="xs:string" name="SystemID" minOccurs="0"/>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
            <xs:element name="BasicCondition" maxOccurs="1" minOccurs="1">
              <xs:complexType>
                <xs:sequence>
                  <xs:element type="xs:string" name="AdvancedSearch" minOccurs="0"/>
                  <xs:element name="SearchType">
                    <xs:simpleType>
                      <xs:restriction base="xs:string">
                        <xs:pattern value="Basic|Advanced|FullText"/>
                      </xs:restriction>
                    </xs:simpleType>
                  </xs:element>
                  <xs:element name="Scopes" maxOccurs="1" minOccurs="1">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="ScopeSearch" maxOccurs="2" minOccurs="1">
                          <xs:simpleType>
                            <xs:restriction base="xs:string">
                              <xs:pattern value="News-Content|News-SubContent|Programs-Content|Programs-
```

```
SubContent|Sports-Content|Sports-SubContent|VideoClips-Content|VideoClips-SubContent|Movies-Content|Movies-SubContent|Series-Content|Series-SubContent|RadioP-Content|RadioP-SubContent|RadioB-Content|RadioB-SubContent"/>
```

```
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Keyword" maxOccurs="unbounded" minOccurs="1">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="SimpleCondition" minOccurs="1">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="Keyword" minOccurs="1">
              <xs:simpleType>
                <xs:restriction base="xs:string">
                  <xs:enumeration value="Id"/>
                  <xs:enumeration value="KeyDate"/>
                  <xs:enumeration value="LongDescription"/>
                  <xs:enumeration value="Description"/>
                  <xs:enumeration value="Notes"/>
                  <xs:enumeration value="Title"/>
                  <xs:enumeration value="SecondaryTitle"/>
                  <xs:enumeration value="Forms"/>
                  <xs:enumeration value="Content_Descriptors_Imagen"/>
                  <xs:enumeration value="Content_Descriptors_Materia"/>
                  <xs:enumeration value="Location_Descriptors_Imagen"/>
                  <xs:enumeration value="Location_Descriptors_Noticia"/>
                  <xs:enumeration value="ContentRelatedCompanies_Descriptors_Derechos"/>
                  <xs:enumeration value="Personalities_Descriptors_Imagen"/>
                  <xs:enumeration value="Personalities_Descriptors_Materia"/>
                  <xs:enumeration value="Custom4"/>
                  <xs:enumeration value="Custom5"/>
                  <xs:enumeration value="Custom6"/>
                  <xs:enumeration value="Creadopor"/>
                  <xs:enumeration value="Audio"/>
                  <xs:enumeration value="NumeroCapitulo"/>
                  <xs:enumeration value="Formavideoclips"/>
                  <xs:enumeration value="IndexingStatus"/>
                  <xs:enumeration value="Candidate"/>
                  <xs:enumeration value="Property"/>
                  <xs:enumeration value="Task"/>
                  <xs:enumeration value="RegionsOfRights"/>
                  <xs:enumeration value="Language"/>
                  <xs:enumeration value="ContentDescriptorGeneromusical"/>
                  <xs:enumeration value="ContentDescriptorEstilomusical"/>
                  <xs:enumeration value="LocationDescriptorMateria"/>
                  <xs:enumeration value="LocationDescriptorPaisMusico"/>
                  <xs:enumeration value="LocationDescriptorPaisProduccion"/>
                  <xs:enumeration value="EventDescriptorAnoProduccion"/>
                  <xs:enumeration value="EventDescriptorTemporada"/>
                  <xs:enumeration value="CompaniesDescriptorDistribuidora"/>
                  <xs:enumeration value="CompaniesDescriptorProductora"/>
```

```
<xs:enumeration value="Credits_Comentarista"/>
<xs:enumeration value="Credits_CreadorMusica"/>
<xs:enumeration value="Credits_Director"/>
<xs:enumeration value="Credits_DirectorFotografia"/>
<xs:enumeration value="Credits_Documentalista"/>
<xs:enumeration value="Credits_DocumentalistaPrograma"/>
<xs:enumeration value="Credits_Editor"/>
<xs:enumeration value="Credits_Guionista"/>
<xs:enumeration value="Credits_Interprete"/>
<xs:enumeration value="Credits_Presentador"/>
<xs:enumeration value="Credits_Productor"/>
<xs:enumeration value="Credits_Realizador"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element type="xs:string" name="Value" minOccurs="1"/>
<xs:element name="Operator" minOccurs="1">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="=|&gt;|&gt;=|&lt;|&lt;=|LIKE"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="Relation" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="NOT|AND|OR"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="CreateTime" maxOccurs="1" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Relation" minOccurs="0"/>
      <xs:element type
="xs:dateTime" name="Begin" minOccurs="1"/>
      <xs:element type="xs:dateTime" name="End" minOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ModifyTime" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Relation" minOccurs="0"/>
      <xs:element type="xs:dateTime" name="Begin" minOccurs="0"/>
      <xs:element type="xs:dateTime" name="End" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="BasicOrder" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="SimpleOrder" maxOccurs="unbounded" minOccurs="0">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="Keyword" minOccurs="0">
              <xs:simpleType>
                <xs:restriction base="xs:string">
                  <xs:pattern value="Id|KeyDate|Title|SecondaryTitle"/>
                </xs:restriction>
              </xs:simpleType>
            </xs:element>
            <xs:element name="Operator" minOccurs="0">
              <xs:simpleType>
                <xs:restriction base="xs:string">
                  <xs:pattern value="ORDER BY"/>
                </xs:restriction>
              </xs:simpleType>
            </xs:element>
            <xs:element name="Type" minOccurs="0" default="DESC">
              <xs:simpleType>
                <xs:restriction base="xs:string">
                  <xs:pattern value="ASC|DESC"/>
                </xs:restriction>
              </xs:simpleType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="PageParameter" maxOccurs="1" minOccurs="1">
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="PageIndex">
        <xs:annotation>
          <xs:documentation>Supuesto: Estas etiquetas no aplica</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element type="xs:string" name="ItemCountPerPage"/>
      <xs:element type="xs:string" name="StartIndex"/>
      <xs:element type="xs:short" name="Numero_Resultados_Devueltas" default="300" minOccurs="1"/>
      <xs:element type="xs:int" name="Numero_Resultado" default="0" minOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Parametros_Detalle" maxOccurs="1" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Log" default="False">
```

```
<xs:simpleType>
  <xs:restriction base="xs:string">
    <xs:pattern value="True|False"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>
```

Los tags de primer nivel incluidos en el esquema son los siguientes:

- **<LoginInfo>**. Se define información del cliente que solicita la operación. Obligatorio.
- **<BasicCondition>**. Se define información del tipo y parámetros de la consulta a realizar. Obligatorio.
- **<BasicOrder>**. Se definen los criterios de ordenación a aplicar en los resultados de la consulta. Obligatorio.
- **<PageParameter>**. Se definen los criterios de paginación a aplicar en los resultados de la consulta. Obligatorio.
- **<Parametros\_Detalle>**. Se define si se debe devolver la query utilizada en la consulta o no, a efectos de depuración. No obligatorio.

## <LOGININFO>

Los tags incluidos en este nivel son los siguientes:

- **<Id\_Peticion\_SONAPS>**. Identificador numérico generado por la aplicación SONAPS que identifica de forma unívoca la petición. Obligatorio. Ejemplo: "000000001".
- **<Sistema\_Origen>**. Identificador de la aplicación que realiza la petición. Obligatorio. Ejemplo "SONAPS".
- **<Nombre\_Flujo>**. Identificador de la operación de la aplicación que genera la petición. Obligatorio. Ejemplo "CONSULTAS\_SONAPS".
- **<userToken>**. Este valor no se va a tener en cuenta. No Obligatorio.
- **<SystemID>**. Este valor no se va a tener en cuenta. No Obligatorio.

## <BASICCONDITION>

Los tags incluidos en este nivel son los siguientes:

- **<AdvancedSearch>**. Este valor no se va a tener en cuenta. No Obligatorio.
- **<SearchType>**. Indica el tipo de búsqueda a realizar. Los valores posibles son: Basic, Advanced, FullText. Obligatorio.
- **<Scopes><ScopeSearch>**. Indica el ámbito de búsqueda dentro de la Base de Datos de Invenio. Obligatorio (1 ó 2 ocurrencias). Los valores posibles son: News-Content, News-SubContent, Programs-Content, ProgramsSubContent, Sports-Content, Sports-SubContent, VideoClips-Content, VideoClips-SubContent, Movies-Content, Movies-SubContent, Series-Content, Series-SubContent. De este valor se obtienen dos condiciones de búsqueda en Exalead:
  - Texto antes del carácter "-". Búsqueda sobre campo *contenido\_template* del índice
  - Texto después del carácter "-". Indica si la búsqueda es por Contenido o SubContenido. Búsqueda sobre campo *source* del índice, (se indica el conector que indexó el registro TM-Contenidos ó TMSubContenidos).

En el caso de que haya 2 tags definidos, se generan las dos condiciones de búsqueda en Exalead por separado y se unen con el operador lógico OR.

- <Keyword><SimpleCondition>. Indican las condiciones de búsquedas simples. Obligatorio (1 a N ocurrencias)
- <Keyword>. Nombre del campo de la Base de Datos de Invenio al que se le va aplicar la condición de búsqueda. Están definidos en la [Tabla 1: Relación entre campos de búsqueda, campos de INVENIO y campos del índice Exalead](#). Obligatorio
- <value>. Valor que se aplica en la condición de búsqueda. La búsqueda por un valor de menos tres caracteres no dará ningún resultado. Obligatorio.
- <operator>. Operador que se aplicará en la condición de búsqueda. Los valores posibles son: =, >, >=, <, <=, LIKE. Obligatorio. De momento no está implementada la búsqueda exacta en el producto Exalead. Para los campos de búsqueda diferentes a *keydate*, sólo está implementada la búsqueda tipo "contiene", por lo que sólo se acepta el operador LIKE.
- <relation>. Indica cómo se relacionan los diferentes contenidos simples de una query. Los valores posibles son: AND, OR, NOT. No obligatorio. En caso de no llevar valor se tomará el valor AND por defecto.

La relación entre las condiciones no permite el paréntesis, con lo que las relaciones se aplican siguiendo el peso de la relación **NOT > AND > OR**, es decir que si por ejemplo tenemos 4 condiciones (**A NOT, B OR, C AND, D OR**) la condición final será. (**NOT A AND C OR B OR D**) o lo que es lo mismo (**((NOT A) AND C) OR (B) OR (D)**)

- <CreateTime>. Condición de búsqueda a aplicar sobre el campo *contenido\_creationdate* del índice. Obligatorio.
  - <Relation>. Indica cómo se relaciona esta condición de búsqueda con el resto de condiciones de búsquedas definidas. Los valores posibles son: AND, OR, NOT. No obligatorio. En caso de no llevar valor se tomará el valor AND por defecto.
  - <Begin>. Indica la fecha desde en la condición de la relación. Obligatorio
  - <End>. Indica la fecha hasta en la condición de la relación. Obligatorio
- <ModifyTime>. Este valor no se va a tener en cuenta. No obligatorio
  - <Relation>.
  - <Begin>.
  - <End>.

## <BASICORDER>

Los tags incluidos en este nivel son los siguientes:

- <SimpleOrder>. Indica la condición de ordenación a aplicar en el resultado de la búsqueda. No obligatorio.
  - <keyword>. Nombre del campo de la Base de Datos de Invenio sobre el que se aplicará el criterio de ordenación. Los valores posibles son: **Id, KeyDate, Title, SecondaryTitle**. No obligatorio. Si no se recibe valor se supone **KeyDate**.
  - <Operator>. Indica el tipo de operación a realizar. Los valores posibles son: ORDER BY. No obligatorio. Si no se recibe se supone **ORDER BY** ○ <Type>. Indica el tipo de ordenación a realizar. Los valores posibles son: ASC, DESC. No obligatorio. Si no se recibe se supone **DESC**

## <PAGEPARAMETER>

Los tags incluidos en este nivel son los siguientes:

- <PageIndex>. Este valor no se va a tener en cuenta. No obligatorio.
- <ItemCountPerPage>. Este valor no se va a tener en cuenta. No obligatorio.
- <StartIndex>. Este valor no se va a tener en cuenta. No obligatorio.
- <Numero\_Resultado\_Devueltas>. Indica el número máximo de resultados a devolver. Si se recibe vacío, se supone un valor de 300. Debe ser un valor positivo mayor que 0. En el fichero de configuración del

Servicio Web se define un valor máximo para este número. Si el valor indicado en el tag es superior a este valor máximo, se toma como valor el definido en el fichero de configuración. Obligatorio. Ver documento "Instalación y Configuración Servicio Web".

- <Numero\_Resultado>. Indica el número de fila del resultado a partir del cual hay que devolver el resultado. El valor 1 indica desde la primera fila. Debe ser un valor positivo mayor que 0. Obligatorio.

Para explicar su funcionamiento, supongamos que el cliente ha definido una paginación de 50 elementos en su sistema. La primera vez que hace la consulta, el cliente no sabe cuántos elementos cumplen la condición introducida por el usuario con lo que en la consulta este parámetro tendrá el valor (<Numero\_Resultado\_Devueltas>50, <Numero\_Resultado>1)

Supongamos que en la respuesta el servicio indica al usuario que **existen 220 registros** que cumplen con la búsqueda y le ha devuelto los 50 primeros, en ese momento el cliente ya sabe que debe mostrar el resultado de forma paginada y si página de 50 en 50 elementos que tiene que muestra 5 páginas.

- Pagina 1: Del registro 1 al 50
- Pagina 2: Del registro 51 al 100
- Pagina 3: Del registro 101 al 150
- Pagina 4: Del registro 151 al 200
- Pagina 5: Del registro 201 al 220

Si el usuario pulsa en la tercera **página 3** el cliente necesitará recuperar los elementos del 101 al 150 y en su petición por tanto deberá pasar los parámetros (<Numero\_Resultado\_Devueltas>50, Numero\_Resultado>=101)

## <PARAMETROSDETALLE>

Los tags incluidos en este nivel son los siguientes:

- <Log>. Indica si se debe devolver la query ejecutada en la consulta o no. Los valores posibles son: True, False. No Obligatorio. Si no se recibe se supone False.

## RESPONSE\_SCHEMA\_V5.0.XSD

---

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://DCMPlatform/v2.0" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="BaseSearchEntityResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="DCMSearch_result">
          <xs:complexType>
            <xs:sequence>
              <xs:element type="xs:int" name="Id_Peticion_SONAPS" minOccurs="1"/>
              <xs:element type="xs:string" name="Sistema_Origen" minOccurs="1"/>
              <xs:element type="xs:string" name="Nombre_Flujo" minOccurs="1"/>
              <xs:element type="xs:string" name="SystemID" minOccurs="1"/>
              <xs:element type="xs:int" name="Total_Resultados" minOccurs="1"/>
              <xs:element type="xs:int" name="Numero_Resultados_Devueltos" minOccurs="1"/>
              <xs:element type="xs:int" name="Numero_Resultado" minOccurs="1"/>
              <xs:element type="xs:int" name="Codigo_Error" default="0" />
              <xs:element type="xs:string" name="Descripcion_Error"/>
              <xs:element name="command">
                <xs:complexType>
                  <xs:sequence>
```



```
        <xs:element name="Log" default="False">
<xs:simpleType>
  <xs:restriction base="xs:string">
    <xs:pattern value="True|False"/>
  </xs:restriction>
</xs:simpleType>
  </xs:element>
  <xs:element type="xs:string" name="query_string" minOccurs="1"/>      <xs:element
name="Scopes">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ScopeSearch" maxOccurs="2" minOccurs="1">      <xs:simpleType>
<xs:restriction base="xs:string">
  <xs:pattern value="News-Content|News-SubContent|Programs-Content|Programs-
SubContent|Sports-Content|Sports-SubContent|VideoClips-Content|VideoClips-SubContent|Movies-
Content|Movies-SubContent|Series-Content|Series-SubContent|RadioP-Content|RadioP-SubContent|RadioB-
Content|RadioB-SubContent"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
  <xs:element name="result_item" maxOccurs="unbounded" minOccurs="0">
<xs:complexType>
  <xs:sequence>
    <xs:element type="xs:int" name="Numero_Resultado" minOccurs="1"/>
    <xs:sequence>
      <xs:element type="xs:string" name="AssetOrigin"/>
      <xs:element type="xs:string" name="CreationDate"/>
      <xs:element type="xs:string" name="MaterialId"/>
      <xs:element type="xs:string" name="KeyDate"/>
      <xs:element type="xs:string" name="LongDescription"/>
      <xs:element type="xs:string" name="Notes"/>
      <xs:element type="xs:string" name="SecondaryTitle"/>
      <xs:element type="xs:string" name="Template"/>      <xs:element
type="xs:string" name="Title"/>
      <xs:element name="Content_Descriptors_Imagen">
        <xs:complexType>
          <xs:sequence>
            <xs:element type="xs:string" name="Imagen" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="Content_Descriptors_Materia">
        <xs:complexType>
          <xs:sequence>
            <xs:element type="xs:string" name="Materia" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="Location_Descriptors_Imagen">
```

```
<xs:complexType>
  <xs:sequence>
    <xs:element type="xs:string" name="Location_Imagen" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Location_Descriptors_Noticia">
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Location_Noticia" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Personalities_Descriptors_Imagen">
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Personality_Imagen" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Personalities_Descriptors_Materia">
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Personalities_Materia" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ContentRelatedCompanies_Descriptors_Derechos">
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="ContentRelatedCompany" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Forms">
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Ambito" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element type="xs:string" name="Rating"/>
<xs:element type="xs:string" name="TCIN"/>
<xs:element type="xs:string" name="Duration"/>
<xs:element type="xs:string" name="Custom4"/>
<xs:element type="xs:string" name="Custom5"/>
<xs:element type="xs:string" name="Custom6"/>
<!-- Elementos nuevos 10/06/2020 -->
<xs:element type="xs:string" name="Creadopor" minOccurs="0" />
<xs:element type="xs:string" name="Audio" minOccurs="0" />
<xs:element type="xs:string" name="NumeroCapitulo" minOccurs="0" />
<xs:element type="xs:string" name="Formavideoclips" minOccurs="0" />
<xs:element type="xs:string" name="IndexingStatus" minOccurs="0" />
<xs:element type="xs:unsignedByte" name="Candidate" minOccurs="0" />
<xs:element type="xs:string" name="Property" minOccurs="0" />
```

```
<xs:element type="xs:string" name="Task" minOccurs="0" />
<xs:element type="xs:string" name="RegionsOfRights" minOccurs="0" />
<xs:element type="xs:string" name="Language" minOccurs="0" />
<xs:element name="ContentDescriptorGeneromusical" minOccurs="0" >
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Content_Generomusical" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ContentDescriptorEstilomusical" minOccurs="0" >
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Content_Estilomusical" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="LocationDescriptorMateria" minOccurs="0" >
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Location_Materia" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="LocationDescriptorPaisMusico" minOccurs="0" >
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Location_PaisMusico" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="LocationDescriptorPaisProduccion" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Location_PaisProduccion" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="EventDescriptorAnoProduccion" minOccurs="0" >
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Event_AnoProduccion" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="EventDescriptorTemporada" minOccurs="0" >
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Event_Temporada" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="CompaniesDescriptorDistribuidora" minOccurs="0" >
<xs:complexType>
  <xs:sequence>
```

```
        <xs:element type="xs:string" name="Companies_Distribuidora" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="CompaniesDescriptorProductora" minOccurs="0" >
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="Companies_Productora" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Credits_Comentarista" minOccurs="0" >
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="Credit_Comentarista" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Credits_CreadorMusica" minOccurs="0" >
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="Credit_CreadorMusica" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Credits_Director" minOccurs="0" >
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="Credit_Director" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Credits_DirectorFotografia" minOccurs="0" >
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="Credit_DirectorFotografia" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Credits_Documentalista" minOccurs="0" >
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="Credit_Documentalista" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Credits_DocumentalistaPrograma" minOccurs="0" >
    <xs:complexType>
      <xs:sequence>
        <xs:element type="xs:string" name="Credit_DocumentalistaPrograma"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Credits_Editor" minOccurs="0" >
```

```
<xs:complexType>
  <xs:sequence>
    <xs:element type="xs:string" name="Credit_Editor" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Credits_Guionista" minOccurs="0" >
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Credit_Guionista" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Credits_Interprete" minOccurs="0" >
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Credit_Interprete" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Credits_Presentador" minOccurs="0" >
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Credit_Presentador" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Credits_Productor" minOccurs="0" >
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Credit_Productor" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Credits_Realizador" minOccurs="0" >
  <xs:complexType>
    <xs:sequence>
      <xs:element type="xs:string" name="Credit_Realizador" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<!-- Fin Elementos nuevos 10/06/2020 -->
<xs:element name="Subcontents" maxOccurs="1" minOccurs="0">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="SubContent" maxOccurs="unbounded" minOccurs="0">
        <xs:complexType>
          <xs:sequence>
            <xs:element type="xs:string" name="Id" minOccurs="1"/>
<xs:element type="xs:string" name="Description" minOccurs="0"/>
            <xs:element type="xs:string" name="TCIN" minOccurs="0"/>
            <xs:element type="xs:string" name="Duration" minOccurs="0"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```



- <Descripcion\_Error>. Descripción del error producido en la operación de consulta. La descripción del error mostrada se lee de la tabla CODIGOS\_ERROR a partir del tag indicado en <Codigo\_Error>. Ver [Esquema de Base de Datos](#).
- <command>. Contiene información resumida de los parámetros de búsqueda solicitados.
- <result\_item>. Contiene información de detalle de los resultados de la búsqueda solicitada. Si la búsqueda no devuelve ningún resultado, el tag no debe aparecer.

## <COMMAND>

Los tags incluidos en este nivel son los siguientes:

- <Log>. Indica si se debe devolver la query ejecutada en la consulta o no. Coincide con el valor pasado en el tag <ParametrosDetalle><Log> del XML de entrada.
- <query\_string>. Cadena que contiene la consulta que se ha ejecutado contra Exalead. Sólo tiene valor si el tag <ParametrosDetalle><Log> del XML de entrada vale True.
- <Scopes><ScopeSearch>. Coincide con los valores pasados en el tag <BasicCondition><Scopes><ScopeSearch> del XML de entrada.

## <RESULT\_ITEM>

Los tags incluidos en este nivel son los mismos tanto si la búsqueda es por contenido como si la búsqueda es por subcontenidos. La única diferencia es el valor de algunos de los tags devueltos que provienen de contenido o de subcontenido en función de los que se esté buscando.

Los tags incluidos en este nivel son los siguientes:

- <Numero\_Resultado>. Número de resultado dentro de la lista de resultados devueltos por la consulta de Exalead. Obligatorio.
- <entity\_info>. Almacena el contenido de cada uno de los elementos de la lista de resultados devueltos por la consulta en Exalead. Todas las etiquetas son obligatorias. Si algún valor es nulo se muestra vacío.
  - <AssetOrigin>. Campo **contenido\_assetorigin** del índice.
  - <MaterialId>. Campo **contenido\_id** del índice.
  - <CreationDate>. Campo **contenido\_creationdate** del índice.
  - <KeyDate>. Campo **contenido\_keydate** del índice.
  - <LongDescription>. Campo **contenido\_description** del índice.
  - <Notes>. Campo **contenido\_notes** del índice.
  - <SecondaryTitle>. Campo **contenido\_secondarytitle** del índice.
  - <Template>. Campo **contenido\_template** del índice.
  - <Title>. Campo **contenido\_title** del índice.
  - <Content\_Descriptors\_Imagen>.
    - <Imagen>. Campo **contenido\_contentdescriptorsimagen** del índice. Ocurrencias 1 a N.
    - <Content\_Descriptors\_Materia>.
    - <Materia>. Campo **contenido\_contentdescriptorsmateria** del índice. Ocurrencias 1 a N.
    - <Location\_Descriptors\_Imagen>.
    - <Location\_Imagen>. Campo **contenido\_locationdescriptorsimagen** del índice. Ocurrencias 1 a N.
  - <Location\_Descriptors\_Noticia>.
    - <Location\_Noticia>. Campo **contenido\_locationdescriptorsnoticia** del índice. Ocurrencias 1 a N.
  - <Personalities\_Descriptors\_Imagen>.
    - <Personality\_magen>. Campo **contenido\_personalitiesdescriptorsimagen** del índice. Ocurrencias 1 a N.
  - <Personalities\_Descriptors\_Materia>.
    - <Personalities\_Materia>. Campo **contenido\_personalitiesdescriptorsmateria** del índice. Ocurrencias 1 a N.
  - <ContentRelatedCompanies\_Descriptors\_Derechos>.

- <ContentRelatedCompany>. Campo **contenido\_contentrelatedcompaniesscriptorsderechos** del índice. Ocurrencias 1 a N.
- <Forms>.
  - <Ambito>. Campo **contenido\_forms** del índice. Ocurrencias 1 a N.
- <Rating>. Campo **minutaje\_rating** del índice. Si la búsqueda es por Contenido, este campo del índice almacena el valor de Rating del Contenido. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de Rating del Contenido asociado al Subcontenido tratado. Ver documento "Instalación y Configuración Exalead Telemadrid".
- <TCIN>. Campo **minutaje\_tcin** del índice. Si la búsqueda es por Contenido, este campo del índice almacena el valor de TCIN del Contenido. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de TCIN del Contenido asociado al Subcontenido tratado. Ver documento "Instalación y Configuración Exalead Telemadrid".
- <Duration>. Campo **minutaje\_duration** del índice. Si la búsqueda es por Contenido, este campo del índice almacena el valor de Duration del Contenido. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de Duration del Contenido asociado al Subcontenido tratado. Ver documento "Instalación y Configuración Exalead Telemadrid".
- <Custom4>.
- <Custom5>.
- <Custom6>.
- <Creadopor>. Campo **contenido\_creadopor** del índice. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de CreadoPor del Contenido asociado al Subcontenido tratado
- <Audio>. Campo **contenido\_audio** del índice. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de Audio del Contenido asociado al Subcontenido tratado
- <NumeroCapitulo>. Campo **contenido\_numerocapitulo** del índice. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de NumeroCapitulo del Contenido asociado al Subcontenido tratado
- <Formavideoclips>. Campo **contenido\_formavideoclips** del índice. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de Formavideoclips del Contenido asociado al Subcontenido tratado o <IndexingStatus>. Campo **contenido\_indexingstatus** del índice. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de IndexingStatus del Contenido asociado al Subcontenido tratado
- <Candidate>. Campo **contenido\_candidate** del índice. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de Candidate del Contenido asociado al Subcontenido tratado
- <Property>. Campo **contenido\_property** del índice. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de Property del Contenido asociado al Subcontenido tratado
- <Task>. Campo **contenido\_task** del índice. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de Task del Contenido asociado al Subcontenido tratado
- <RegionsOfRights>. Campo **contenido\_regionsofrights** del índice. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de RegionsOfRights del Contenido asociado al Subcontenido tratado
- <Language>. Campo **contenido\_language** del índice. Si la búsqueda es por SubContenido, este campo del índice almacena el valor de Language del Contenido asociado al Subcontenido tratado o <ContentDescriptorGeneromusical>.
  - <Content\_Generomusical>. Campo **contenido\_contentdescriptorgeneromusical** del índice. Ocurrencias 1 a N.
- <ContentDescriptorEstilomusical>.
  - <Content\_Estilomusical>. Campo **contenido\_contentdescriptorestilomusical** del índice. Ocurrencias 1 a N.
- <LocationDescriptorMateria>.
  - <Location\_Materia>. Campo **contenido\_locationdescriptormateria** del índice. Ocurrencias 1 a N.
- <LocationDescriptorPaisMusico>.
  - <Location\_PaisMusico>. Campo **contenido\_locationdescriptorpaismusico** del índice.



- Ocurrencias 1 a N.
- <LocationDescriptorPaisProduccion>.
  - <Location\_PaisProduccion>. Campo **contenido\_locationdescriptorpaisproduccion** del índice.
  - Ocurrencias 1 a N.
- <EventDescriptorAnoProduccion>.
  - <Event\_AnoProduccion>. Campo **contenido\_eventdescriptoranoproduccion** del índice.
  - Ocurrencias 1 a N.
- <EventDescriptorTemporada>.
  - <Event\_Temporada>. Campo **contenido\_eventdescriptortemporada** del índice.
  - Ocurrencias 1 a N.
- <CompaniesDescriptorDistribuidora>.
  - <Companies\_Distribuidora>. Campo **contenido\_companiesdescriptoristribuidora** del índice.
  - Ocurrencias 1 a N.
- <CompaniesDescriptorProductora>.
  - <Companies\_Productora>. Campo **contenido\_companiesdescriptorproductora** del índice.
  - Ocurrencias 1 a N.
- <Credits\_Comentarista>.
  - <Credit\_Comentarista>. Campo **contenido\_credits\_comentarista** del índice. Ocurrencias 1 a N.
- <Credits\_CreadorMusica>.
  - <Credit\_CreadorMusica>. Campo **contenido\_credits\_creadormusica** del índice.
  - Ocurrencias 1 a N.
- <Credits\_Director>.
  - <Credit\_Director>. Campo **contenido\_credits\_director** del índice. Ocurrencias 1 a N.
- <Credits\_DirectorFotografia>.
  - <Credit\_DirectorFotografia>. Campo **contenido\_credits\_directorfotografia** del índice.
  - Ocurrencias 1 a N.
- <Credits\_Documentalista>.
  - <Credit\_Documentalista>. Campo **contenido\_credits\_documentalista** del índice.
  - Ocurrencias 1 a N.
- <Credits\_DocumentalistaPrograma>.
  - <Credit\_DocumentalistaPrograma>. Campo **contenido\_credits\_documentalistaprograma** del índice.
  - Ocurrencias 1 a N.
- <Credits\_Editor>.
  - <Credit\_Editor>. Campo **contenido\_credits\_editor** del índice. Ocurrencias 1 a N.
- <Credits\_Guionista>.
  - <Credit\_Guionista>. Campo **contenido\_credits\_guionista** del índice. Ocurrencias 1 a N.
- <Credits\_Interprete>.
  - <Credit\_Interprete>. Campo **contenido\_credits\_interprete** del índice. Ocurrencias 1 a N.
- <Credits\_Presentador>.
  - <Credit\_Presentador>. Campo **contenido\_credits\_presentador** del índice. Ocurrencias 1 a N.
- <Credits\_Productor>.
  - <Credit\_Productor>. Campo **contenido\_credits\_productor** del índice.
  - Ocurrencias 1 a N.
- <Credits\_Realizador>.
  - <Credit\_Realizador>. Campo **contenido\_credits\_realizador** del índice. Ocurrencias 1 a N.
- <SubContents><SubContent>. Si la búsqueda es por Contenido, devuelve todos los minutajes encontrados en Exalead del contenido tratado, (cuyo **minutaje\_idcontenido** coincida con el

**contenido\_id** tratado). Si la búsqueda es por SubContenido, devuelve sólo el minutaje correspondiente al subcontenido tratado, (cuyo **minutaje\_id** coincida con **contenido\_idsubcontent**). Ocurrencias de 0 a N.

- <Id>. Campo **minutaje\_id** del índice.
- <Description>. Campo **minutaje\_description** del índice.
- <TCIN>. Campo **minutaje\_tcin** del índice.
- <Duration>. Campo **minutaje\_duration** del índice.
- <Video\_Files><Video\_File>. Si la búsqueda es por Contenido, devuelve todos los VideoFiles encontrados en Exalead del contenido tratado, (cuyo **mediavideo\_idcontenido** coincida con el **contenido\_id** tratado). Si la búsqueda es por SubContenido, devuelve todos los VideoFiles encontrados en Exalead del Contenido asociado al Subcontenido tratado, (cuyo **mediavideo\_idcontenido** coincida con el **contenido\_id** tratado). Ocurrencias de 0 a N.
  - <ID>. Campo **mediavideo\_id** del índice.
  - <Tipo>. Campo **mediavideo\_tipo** del índice.
  - <AspectRatio>. Campo **mediavideo\_aspectratio** del índice.
  - <Path>. Campo **mediavideo\_path** del índice.
  - <TCIN>. Campo **mediavideo\_tcin** del índice.
  - <Duration>. Campo **mediavideo\_duration** del índice.
  - <UseType>. Campo **mediavideo\_useType** del índice.

## REGISTRO DE OPERACIONES

La actividad del servicio será registrada en un esquema de Base de Datos. Existen dos tipos de registros:

- **Registro de operación:** Donde se registran las operaciones y errores que produce la ejecución del servicio
- **Registro de auditoria:** Donde se registra de forma separada la información sobre las consultas que se le están solicitando al servicio

Además, los ficheros XML de entrada y salida se van a guardar en una carpeta local configurable desde la aplicación

## REGISTRO DE OPERACIÓN

Registra las operaciones que el servicio va ejecutando, incluyendo los errores que se hayan podido producir. Por cada operación se guarda la siguiente información:

- Fecha de la operación YYYY-MM-DD HH:MM: SSS (Milisegundos)
- Tipo de registro (INFO, WARNING, ERROR)
  - INFO: Indica que la línea registra la ejecución de una operación
  - WARNING: Indica que la línea registra un error leve de la operación que no detiene la solicitud
  - ERROR: Indica que la línea registra un error de la operación que puede detener la ejecución del servicio y provocar una excepción
- Identificador de la petición, (<Id\_Peticion\_SONAPS >)
- Identificador del peticionario (<Sistema\_Origen>)
- Identificador de la operación de la aplicación que genera la petición, (<Nombre\_Flujo>).
  - Código del error, si la línea es de tipo ERROR se almacena el código asociado al error producido
  - Descripción: Texto que describe la operación.
- Descripción extendida: En caso de error, describirá lo más detalladamente posible el error que se ha producido.

Mediante fichero de configuración del servicio se podrá configurar el nivel mínimo de tipo de operación a registrar:

INFO, WARNING, ERROR

Existirá una tabla en el esquema donde estén recogidos todos los códigos de error y su descripción asociada.

## REGISTRO DE AUDITORIA

Registra las operaciones desde un punto de vista de negocio, en nuestro caso registrará cada una de las operaciones de consulta recibidas, guardando la siguiente información de cada una de ellas

- Fecha de la operación YYYY-MM-DD HH:MM: SSS (Milisegundos)
- Identificador de la petición, (<Id\_Peticion\_SONAPS>)
- Identificador del peticionario (<Sistema\_Origen>)
- Identificador de la operación de la aplicación que genera la petición, (<Nombre\_Flujo>).
- Tipo de petición (**Basic**, **Advanced**, **FullText**)
- Número de resultados totales de la consulta (<Total\_Resultados>)
- Índice del primer resultado devuelto (<Numero\_Resultados\_Devueltas>)
- Número de resultados devueltos (<Numero\_Resultado>)
- Tiempo de ejecución total de la consulta, desde que el servicio recibe la petición hasta que devuelve la respuesta (Milisegundos)
- Tiempo de ejecución de la consulta en Exalead (Milisegundos)

## ESQUEMA DE BASE DE DATOS

El esquema de la Base de Datos va a tener dos tablas, una para almacenar cada tipo de registro:

- **REGISTRO\_OPERACION:** Operaciones y errores en la ejecución del servicio
  - COD\_OPE INT (PK) (NN) (I): Código interno de la operación. Secuencial ○ FECHA DATETIME (NN (I)): Fecha en la que se registra la operación
  - TIPO TINYINT (NN): Tipo de registro. Será uno de los siguientes valores numéricos
    - 0 – ERROR
    - 1 – WARNING
    - 2 – INFO
  - ID\_PETICION\_SONAPS VARCHAR (255) (NN): Identificador de la operación ○ SISTEMA\_ORIGEN VARCHAR (255) (NN): Identificador del peticionario
  - NOMBRE\_FLUJO VARCHAR (255) (NN): Identificador de la operación de la aplicación que genera la petición
  - COD\_ERROR SMALLINT (FK) (NN): Si el registro es de tipo ERROR, se almacena el código de error producido. Si no es de tipo ERROR devuelve 0.
  - DESCRIPCIÓN VARCHAR (255) (NN): Texto que describe la operación o error producido
  - DESCRIPCION\_EXTENDIDA VARCHAR (1024): En caso de error, muestra lo más detalladamente posible el error producido.
- **REGISTRO\_AUDITORIA:** Información de las consultas llevadas a cabo en el servicio
  - COD\_AUD INT (PK) (NN) (I): Código interno de la auditoría. Secuencial ○ FECHA DATETIME (NN) (I): Fecha en la que se registra la auditoría
  - TIPO TINYINT (NN): Tipo de búsqueda. Será uno de los siguientes valores numéricos
    - 1 – Basic
    - 2 – Advanced
    - 3 – FullText
  - ID\_PETICION\_SONAPS VARCHAR (255) (NN): Identificador de la operación
  - SISTEMA\_ORIGEN VARCHAR (255) (NN): Identificador del peticionario
  - NOMBRE\_FLUJO VARCHAR (255) (NN): Identificador de la operación de la aplicación que genera la petición
  - TOTAL\_RESULTADOS INT (NN): Número de resultados totales de la consulta
  - NUMERO\_RESULTADOS\_DEVUELTOS SMALLINT (NN): Índice del primer resultado devuelto
  - NUMERO\_RESULTADO SMALLINT (NN): Número de resultados devueltos
  - TIEMPO\_EJECUCION\_TOTAL INT (NN): Tiempo de ejecución total de la consulta, desde que el servicio recibe la petición hasta que devuelve la respuesta, en milisegundos
  - TIEMPO\_EJECUCION\_EXALEAD INT (NN): Tiempo de ejecución de la consulta en el producto Exalead, en milisegundos
- **CODIGOS\_ERROR:** Relación de posibles casos de error a incluir en el registro de operaciones
  - COD\_ERROR SMALLINT (PK) (NN) (I): Código de error

- DESC\_ERROR VARCHAR (255) (NN) (I): Descripción del error

## REGISTRO FICHEROS XML

Los ficheros XML de entrada y salida de cada operación de búsqueda llevada a cabo por el servicio, se van a guardar en dos carpetas del File System, una para los ficheros XML de entrada y otra para los ficheros XML de salida. Las dos carpetas serán configurables mediante el fichero de configuración del servicio.

La nomenclatura de ambos ficheros será la siguiente:

-<COD\_AUD>\_<Sistema\_Origen>-YYYYMMDD-HHMM-<Id\_Peticion\_SONAPS>.xml

Siendo

<COD\_AUD> el valor del campo del mismo nombre en la tabla REGISTRO\_AUDITORIA, y

<Sistema\_Origen>/<Id\_Peticion\_SONAPS> el valor de los tags del mismo nombre en los ficheros XML de entrada y salida. Si no existiera alguno de estos valores, los ficheros XML de entrada y salida se grabarían, en cualquier caso. Por ejemplo, si falta el valor de las tres etiquetas se grabarían como YYYYMMDD-HHMM.xml

Esta operación es opcional y podrá activarse/desactivarse mediante el fichero de configuración del Servicio Web. Ver documento "Instalación y Configuración Servicio Web".

## INDEXACIÓN EN EXALEAD

El proceso de indexación es el encargado de recuperar la información de **INVENIO** e irla guardando en el índice que hayamos definido en Exalead.

Este proceso se divide en los siguientes apartados:

## DEFINICIÓN DEL ÍNDICE

Antes de cargar ninguna información debemos definir cómo la vamos a almacenar dentro del índice, esta definición debe estar enfocada a cubrir tanto los criterios de búsqueda como los resultados que deseamos obtener.

En nuestro caso optaremos por un índice dividido en tres tipos de objetos

- Un objeto **Contenido** donde almacenaremos la información general de contenidos y sub-contenidos
- Un objeto **Minutaje** donde almacenaremos el minutaje de los sub-contenidos que van asociados a cada contenido
- Un objeto **MediaVideo** donde almacenaremos la información de los formatos de video en los que se encuentra cada contenido

Esta división nos permite:

- Poder buscar por los campos de contenido y sub-contenido, al estar todos los campos de búsqueda estarán dentro del objeto **Contenido**
- Poder devolver en la respuesta el contenido relacionado con sus minutajes y formatos de video, al estar almacenados en tres objetos que al final funcionan como tablas relacionadas y en las que dado un **Contenido** podemos obtener sus **Minutajes** y **MediaVideos**

En los siguientes puntos se define cada uno de estos objetos con los campos que contiene, para cada objeto se mostrará una tabla con las siguientes columnas:

- Nombre del campo en el índice
- Tipo de datos almacenado
- Si podemos o no realizar búsquedas por el contenido de ese campo

## OBJETO CONTENIDO

Contiene la información general de contenidos y sub-contenidos, en este objeto encontramos todos los campos por los que vamos a poder buscar

Campo del índice	Tipo	Búsqueda
contenido_id	Alfanumérico	Si
contenido_idsubcontenido	Date_Time	Si
contenido_creationdate	Date_Time	Si
contenido_keydate	Date_Time	No
contenido_assetorigin	Alfanumérico	Si
contenido_description	Alfanumérico	Si
contenido_notes	Alfanumérico	Si
contenido_title	Alfanumérico	Si
contenido_secondarytitle	Alfanumérico	Si
contenido_forms	Alfanumérico (múltiple)	No
contenido_template	Alfanumérico	Si
contenido_creadopor	Alfanumérico	Si
contenido_audio	Alfanumérico	Si
contenido_numerocapitulo	Alfanumérico	Si
contenido_formavideoclips	Alfanumérico	Si
contenido_indexingstatus	Alfanumérico	Si
contenido_candidate	Alfanumérico	Si
contenido_property	Alfanumérico	Si
contenido_task	Alfanumérico	Si
contenido_regionsofrights	Alfanumérico	Si
contenido_language	Alfanumérico	Si
contenido_contentdescriptorimagen	Alfanumérico (múltiple)	Si
contenido_contentdescriptormateria	Alfanumérico (múltiple)	Si
contenido_contentdescriptorgeneromusical	Alfanumérico (múltiple)	Si
contenido_contentdescriptorestilomusical	Alfanumérico (múltiple)	Si
contenido_locationdescriptorimagen	Alfanumérico (múltiple)	Si

Campo del índice	Tipo	Búsqueda
contenido_locationdescriptornoticia	Alfanumérico (múltiple)	Si
contenido_locationdescriptormateria	Alfanumérico (múltiple)	Si
contenido_locationdescriptorpaismusico	Alfanumérico (múltiple)	Si
contenido_locationdescriptorpaisproduccion	Alfanumérico (múltiple)	Si
contenido_personalitiesdescriptorimagen	Alfanumérico (múltiple)	Si
contenido_personalitiesdescriptormateria	Alfanumérico (múltiple)	Si
contenido_eventdescriptoranoproduccion	Alfanumérico (múltiple)	Si
contenido_eventdescriptortemporada	Alfanumérico (múltiple)	Si
contenido_contentrelatedcompaniesdescriptorsderechos	Alfanumérico (múltiple)	Si
contenido_companiesdescriptordistribuidora	Alfanumérico (múltiple)	Si
contenido_companiesdescriptorproductora	Alfanumérico (múltiple)	Si
contenido_credits_comentarista	Alfanumérico (múltiple)	Si
contenido_credits_creadormusica	Alfanumérico (múltiple)	Si
contenido_credits_director	Alfanumérico (múltiple)	Si
contenido_credits_directorfotografia	Alfanumérico (múltiple)	Si
contenido_credits_documentalista	Alfanumérico (múltiple)	Si
contenido_credits_documentalistaprograma	Alfanumérico (múltiple)	Si
contenido_credits_editor	Alfanumérico (múltiple)	Si
contenido_credits_guionista	Alfanumérico (múltiple)	Si
contenido_credits_interprete	Alfanumérico (múltiple)	Si
contenido_credits_presentador	Alfanumérico (múltiple)	Si
contenido_credits_productor	Alfanumérico (múltiple)	Si
contenido_credits_realizador	Alfanumérico (múltiple)	Si
contenido_fecha_actualizacion	Alfanumérico (múltiple)	Si

## OBJETO MINUTAJE

Contiene la información sobre los contenidos y sub-contenidos, pero únicamente aquella que hace referencia al propio minutaje

Campo del índice	Tipo	Búsqueda
minutaje_idcontenido - (Identificador del contenido al que pertenece)	Alfanumérico	No
minutaje_id - (Identificador del sub-contenido)	Alfanumérico	No
Minutaje_description	Alfanumérico	No
minutaje_tcin	Alfanumérico	No
minutaje_duration	Alfanumérico	No
minutaje_rating	Alfanumérico	No
minutaje_nombrefichero	Alfanumérico	No

## OBJETO MEDIAVIDEO

Contiene la información sobre los formatos de video en los que podemos encontrar un contenido

Campo del índice	Tipo	Búsqueda
mediaVideo_idContenido - (Identificador del contenido al que pertenece)	Alfanumérico	No
mediaVideo_id - (Identificador del media video)	Alfanumérico	No
mediaVideo_tipo	Alfanumérico	No
mediaVideo_aspectRatio	Alfanumérico	No
mediaVideo_path	Alfanumérico	No
mediaVideo_Tcin	Alfanumérico	No
mediaVideo_duration	Alfanumérico	No
mediavideo_abbreviation	Alfanumérico	No
mediavideo_useType	Alfanumérico	No

## DEFINICIÓN DE CONECTORES

Los conectores son los componentes encargados de extraer la información de las fuentes de datos, en este caso de la base de datos de INVENIO.

Dada la estructura del índice, lo más lógico sería generar 4 conectores, todos ellos de base de datos:

- **Conector de contenidos:** Extraerá la información de la tabla de **contenidos** que va al objeto **contenido** del índice y tiene las siguientes propiedades
  - **Tipo de conector:** Base de datos
  - **Consulta de carga total:** Contra los contenidos de INVENIO, será definida junto con el equipo de Telemadrid, en caso de ser viable o **Consulta de carga incremental:** Se genera un documento de como configurar una carga incremental.
  - **Programación:** En principio diaria (nocturna), si es posible definir algún tipo de consulta incremental podríamos bajar la frecuencia de ejecución del conector
- **Conector de sub-contenidos:** Extraerá la información de la tabla de **sub-contenidos** que va al objeto **contenido** del índice y tiene las siguientes propiedades
  - **Tipo de conector:** Base de datos
  - **Consulta de carga total:** Contra los contenidos de INVENIO, será definida junto con el equipo de Telemadrid, en caso de ser viable o **Consulta de carga incremental:** Se genera un documento de como configurar una carga incremental.
  - **Programación:** En principio diaria (nocturna), si es posible definir algún tipo de consulta incremental podríamos bajar la frecuencia de ejecución del conector
- **Conector de minutajes:** Extraerá la información de la tabla de **contenidos** y **sub-contenidos** que va al objeto **minutaje** del índice y tiene las siguientes propiedades
  - **Tipo de conector:** Base de datos
  - **Consulta de carga total:** Contra los contenidos de INVENIO, será definida junto con el equipo de Telemadrid, en caso de ser viable o **Consulta de carga incremental:** Se genera un documento de como configurar una carga incremental.
  - **Programación:** En principio diaria (nocturna), si es posible definir algún tipo de consulta incremental podríamos bajar la frecuencia de ejecución del conector
- **Conector de MediaVideos:** Extraerá la información de la tabla de **media-videos** que va al objeto **MediaVideo** del índice y tiene las siguientes propiedades
  - **Tipo de conector:** Base de datos
  - **Consulta de carga total:** Contra los contenidos de INVENIO, será definida junto con el equipo de Telemadrid, en caso de ser viable o **Consulta de carga incremental:** Se genera un documento de como configurar una carga incremental.
  - **Programación:** En principio diaria (nocturna), si es posible definir algún tipo de consulta incremental podríamos bajar la frecuencia de ejecución del conector

## DEFINICIÓN DE LA INDEXACIÓN

Este proceso es el encargado de mapear la información obtenida por los conectores en los campos del índice incluyendo todas las posibles transformaciones que se pueden hacer sobre cada uno de sus campos.

A continuación, vemos como se mapean los datos obtenidos por cada conector en los campos del índice, para cada conector mostraremos una tabla con las siguientes columnas:

- Campo origen de INVENIO
- Campo destino del índice

## MAPEO DEL CONECTOR DE CONTENIDOS

Obtiene la información de las tablas y la añade al objeto Contenido del índice

CONTENIDO	MODELO EXALEAD
IDContent	contenido_id
CreationDate	creationdate
KeyDate	contenido_keydate
AssetOrigin	contenido_assetorigin
LongDescription	contenido_description
Notes	contenido_notes
Title	contenido_title
SecondaryTitle	contenido_secondarytitle
FORMS	contenido_forms
Template	contenido_template
Creadopor	contenido_creadopor
Audio	contenido_audio
NumeroCapitulo	contenido_numerocapitulo
Formavideoclips	contenido_formavideoclips
IndexingStatus	contenido_indexingstatus
Candidate	contenido_candidate
Property	contenido_property
Task	contenido_task
RegionsOfRights	contenido_regionsofrights
Language	contenido_language
ContentDescriptorImagen	contenido_contentdescriptorimagen
ContentDescriptorMateria	contenido_contentdescriptormateria
ContentDescriptorGeneromusical	contenido_contentdescriptorgeneromusical
ContentDescriptorEstilomusical	contenido_contentdescriptorestilomusical
LocationDescriptorImagen	contenido_locationdescriptorimagen
LocationDescriptorNoticia	contenido_locationdescriptornoticia
LocationDescriptorMateria	contenido_locationdescriptormateria
LocationDescriptorPaisMusico	contenido_locationdescriptorpaismusico
LocationDescriptorPaisProduccion	contenido_locationdescriptorpaisproduccion
PersonalitiesDescriptorImagen	contenido_personalitiesdescriptorimagen
PersonalitiesDescriptorMateria	contenido_personalitiesdescriptormateria
EventDescriptorAnoProduccion	contenido_eventdescriptoranoproduccion
EventDescriptorTemporada	contenido_eventdescriptortemporada



CONTENIDO	MODELO EXALEAD
CompaniesDescriptorDerechos	contenido_contentrelatedcompaniesdescriptorsderechos
CompaniesDescriptorDistribuidora	contenido_companiesdescriptoristribuidora
CompaniesDescriptorProductora	contenido_companiesdescriptorproductora
Credits_Comentarista	contenido_credits_comentarista
Credits_CreadorMusica	contenido_credits_creadormusica
Credits_Director	contenido_credits_director
Credits_DirectorFotografia	contenido_credits_directorfotografia
Credits_Documentalista	contenido_credits_documentalista
Credits_DocumentalistaPrograma	contenido_credits_documentalistaprograma
Credits_Editor	contenido_credits_editor
Credits_Guionista	contenido_credits_guionista
Credits_Interprete	contenido_credits_interprete
Credits_Presentador	contenido_credits_presentador
Credits_Productor	contenido_credits_productor
Credits_Realizador	contenido_credits_realizador
Fecha_Actualizacion	contenido_fecha_actualizacion

En un principio los campos Tcin y Duration se iban a registrar en el índice, pero se va a tratar a posteriori sacando los datos del índice del minutaje.

La query para recuperar los valores de la tabla de contenidos es la siguiente:

```
SELECT IDContent, CreationDate, KeyDate, AssetOrigin, LongDescription, Notes, Title, SecondaryTitle, FORMS,
Template,
Creadopor, Audio, NumeroCapitulo, Formavideoclips, IndexingStatus, Candidate, Property, Task,
RegionsOfRights,
Language, ContentDescriptorImagen, ContentDescriptorMateria, ContentDescriptorGeneromusical,
ContentDescriptorEstilomusical, LocationDescriptorImagen, LocationDescriptorNoticia,
LocationDescriptorMateria,
LocationDescriptorPaisMusico, LocationDescriptorPaisProduccion, PersonalitiesDescriptorImagen,
PersonalitiesDescriptorMateria, EventDescriptorAnoProduccion, EventDescriptorTemporada,
CompaniesDescriptorDerechos, CompaniesDescriptorDistribuidora, CompaniesDescriptorProductora,
Credits_Comentarista, Credits_CreadorMusica, Credits_Director, Credits_DirectorFotografia,
Credits_Documentalista, Credits_DocumentalistaPrograma, Credits_Editor, Credits_Guionista,
Credits_Interprete, Credits_Presentador, Credits_Productor, Credits_Realizador
```

FROM Extraccion\_CONTENT

## MAPEO DEL CONECTOR DE SUB-CONTENIDOS

Obtiene la información de las tablas ICVideoSubContent, ICVideoContent, ICVideoMediaVersion, ICVideoSegment y la añade al objeto Contenido del índice

SUBCONTENIDO	MODELO EXALEAD
IDSubcontent	contenido_idsubcontenido
CreationDate	creationdate

KeyDate	contenido_keydate
AssetOrigin	contenido_assetorigin
Description	contenido_description
Notes	contenido_notes
Title	contenido_title
SecondaryTitle	contenido_secondarytitle
FORMS	contenido_forms
Template	contenido_template
ContentDescriptorImagen	contenido_contentdescriptorimagen
ContentDescriptorMateria	contenido_contentdescriptormateria
ContentDescriptorGeneromusical	contenido_contentdescriptorgeneromusical
ContentDescriptorEstilomusical	contenido_contentdescriptorestilomusical
LocationDescriptorImagen	contenido_locationdescriptorimagen
LocationDescriptorNoticia	contenido_locationdescriptornoticia
LocationDescriptorMateria	contenido_locationdescriptormateria
LocationDescriptorPaisMusico	contenido_locationdescriptorpaismusico
LocationDescriptorPaisProduccion	contenido_locationdescriptorpaisproduccion
PersonalitiesDescriptorImagen	contenido_personalitiesdescriptorimagen
PersonalitiesDescriptorMateria	contenido_personalitiesdescriptormateria
EventDescriptorAnoProduccion	contenido_eventdescriptoranoproduccion
EventDescriptorTemporada	contenido_eventdescriptortemporada
CompaniesDescriptorDerechos	contenido_contentrelatedcompaniesdescriptorsderechos
CompaniesDescriptorDistribuidora	contenido_companiesdescriptordistribuidora
CompaniesDescriptorProductora	contenido_companiesdescriptorproductora
Credits_Comentarista	contenido_credits_comentarista
Credits_CreadorMusica	contenido_credits_creadormusica
Credits_Director	contenido_credits_director
Credits_DirectorFotografia	contenido_credits_directorfotografia
Credits_Documentalista	contenido_credits_documentalista
Credits_DocumentalistaPrograma	contenido_credits_documentalistaprograma
Credits_Editor	contenido_credits_editor
Credits_Guionista	contenido_credits_guionista
Credits_Interprete	contenido_credits_interprete
Credits_Presentador	contenido_credits_presentador
Credits_Productor	contenido_credits_productor
Credits_Realizador	contenido_credits_realizador

Fecha_Actualizacion	contenido_fecha_actualizacion
---------------------	-------------------------------

En un principio los campos Tcin y Duration se iban a registrar en el índice, pero se va a tratar a posteriori sacando los datos del índice del minutaje.

La query para recuperar los valores de la tabla de subcontenidos es la siguiente:

```
SELECT IDContent, IDSubcontent, CreationDate, KeyDate, AssetOrigin, Description, Notes, Title, SecondaryTitle,
FORMS
as SUBCONTENT_FORMS, Template, ContentDescriptorImagen, ContentDescriptorMateria,
ContentDescriptorGeneromusical, ContentDescriptorEstilomusical, LocationDescriptorImagen,
LocationDescriptorNoticia, LocationDescriptorMateria, LocationDescriptorPaisMusico,
LocationDescriptorPaisProduccion, PersonalitiesDescriptorImagen, PersonalitiesDescriptorMateria,
EventDescriptorAnoProduccion, EventDescriptorTemporada, CompaniesDescriptorDerechos,
CompaniesDescriptorDistribuidora, CompaniesDescriptorProductora, Credits_Comentarista,
Credits_CreadorMusica,
Credits_Director, Credits_DirectorFotografia, Credits_Documentalista, Credits_DocumentalistaPrograma,
Credits_Editor, Credits_Guionista, Credits_Interprete, Credits_Presentador, Credits_Producor,
Credits_Realizador
FROM Extraccion_SUBCONTENT
```

## MAPEO DEL CONECTOR DE MINUTAJES

Obtiene la información de las tablas ICVideoSubContent, ICVideoContent y ICVideoSegment y la añade al objeto Minutaje del índice

Campo de INVENIO	Campo del índice
ICVideoContent.ID	Minutaje_idContenido
ICVideoSubContent.ID	Minutaje_id
ICVideoSubContent.Description	Minutaje_description
ICVideoSegment.Tcin	Minutaje_Tcin
ICVideoSegment.Duration	Minutaje_duration

Obtiene la información de las tablas ICVideoFile y ICVideoContent y la añade al objeto MediaVideo del índice

Campo de INVENIO	Campo del índice
ICVideoContent.ID	MediaVideo.idContenido
ICVideoFile.ID	MediaVideo.id
min(isnull(FIL.FilePath,'ALTA'))	MediaVideo.tipo
min(wliAspectRatio.value)	MediaVideo_aspectRatio
min('\'+(case FIL.ServerName when 'app-tma' then 'app-tma-s\' +FIL.FilePath when 'nas- archivo.nastm.local\Baja' then FIL .ServerName+'\' +FIL.FilePath else FIL.FilePath end)) as FilePath	MediaVideo_path
min(IsNull(fil.startTC,0))	MediaVideo_Tcin
min(IsNull(fil.Duration,0))	MediaVideo_duration
min(isnull(wliUseType.value,'')) as UseType	MediaVideo_useType

## MAPEO CAMPOS DEL CONECTOR DE MEDIAVIDEOS

---

La query para recuperar los valores mediavideo es la siguiente:

```
Select  min(VC.Id) AS IdContent,  FIL.id as Fichero,
min(isnull(FIL.FilePath,'ALTA')) as TipoMediaVideo,
min(wliAspectRatio.value) as AspectRatio,
min(wliAspectRatio.Abbreviation) as Abbreviation,
min('\'+(case FIL.ServerName
            when 'app-tma'
            then 'app-tma-s\' +FIL.FilePath
when 'nas-archivo.nastm.local\Baja'  then FIL .ServerName+'\' +FIL.FilePath
            else FIL.FilePath
end)) as FilePath,
min(IsNull(fil.startTC,0)) as TCIN,
min(IsNull(fil.Duration,0)) as
Duracion,
min(isnull(wliUseType.value,'')) as
UseType from
    hxps.ICVideoFile
as FIL left join
    hxps.ICVideoSegment as SEGM      on
SEGM.RightSide=FIL.OID left join
    hxps.ICVideoRendition as RE      on
RE.OID=SEGM.LeftSide left join
    hxps.ICVideoRenditionsAssocItem as REAI
on RE.OID=REAI.RightSide left JOIN
    hxps.ICVideoMediaVersion as ICMV
on ICMV.OID=REAI.LeftSide left JOIN
    hxps.ICVideoContentMedVersAssocitem
as MVAI      on
MVAI.RightSide=ICMV.OID left join
    hxps.ICVideoContent as VC      on
VC.OID=MVAI.LeftSide
--AspectRatio left join
    hxps.QDIWordListItem AS
wliAspectRatio      ON
wliAspectRatio.OID =
FIL.AspectRatio
--RenditionTypes
left join  hxps.QDIWordListItem AS
wliRenditionTypes
            ON wliRenditionTypes.OID = RE.Type
--UseType left join
    hxps.QDIWordListItem AS
wliUseType      ON wliUseType.OID
= FIL.UseType where

VC.IsDigitized=1
group by
    FIL.id
```

## ANEXO: INVOCACIÓN SÍNCRONA Y ASÍNCRONA DEL SERVICIO

El Servicio Web de búsqueda indexSearch se ha diseñado como un Servicio Web síncrono. La plataforma .NET framework .NET 3.5, permite que un Servicio Web síncrono puede ser llamado tanto de forma síncrona como asíncrona, (utilizando el patrón de diseño Begin/End). En este anexo se describen un par de ejemplos de clientes que realizan llamadas síncronas y asíncronas al Servicio Web.

### DEFINICIÓN DEL SERVICIO WEB INDEXSEARCH

La interfaz del Servicio Web de búsqueda de indexSearch, que funciona como intermediario entre los clientes y el índice de Exalead, tiene definido el siguiente formato:

```
[WebMethod]
public ResponseSummary indexSearch(Petitioner petitioner, SearchType searchType, Query query,
SortCriteria sortCriteria, PaginationCriteria pagCriteria, bool returnIn) {
    ResponseSummary
    responseSummary = new ResponseSummary();

    return responseSummary;
}
```

### INVOCACIÓN AL SERVICIO WEB INDEXSEARCH DE FORMA SÍNCRONA

Cuando vamos a crear el cliente, la manera común de invocación a un Servicio Web es de forma síncrona, esto es, se va a realizar la operación de invocación al servicio y el proceso va a esperar hasta que esta se haya completado.

Esta forma es efectiva y práctica, cuando se completa la operación rápidamente y el tiempo de espera no es crítico en la ejecución de la aplicación.

Para este tipo de llamadas únicamente se va a disponer de método de invocación:  
exaleadClient.< indexSearch >.

Los pasos a realizar para implementar un cliente de forma síncrona, se detallan a continuación:

### REFERENCIAS AL SERVICIO

1. Indicar referencia al Servicio Web de IndexSearch.  
using NetExaleadWebServiceClient.ExaleadWebServiceRef;

### TAREA PRINCIPAL

1. Instanciación del Servicio Web IndexSearch.  
NetExaleadWebServiceClient.ExaleadWebServiceRef.ExaleadWebServiceS OapClient exaleadClient  
= new  
NetExaleadWebServiceClient.ExaleadWebServiceRef.ExaleadWebServiceS oapClient ();
2. Invocación al Servicio Web de forma síncrona  
IAsyncResult ar = exaleadClient.indexSearch(petitioner, searchType, query, sortCriteria, pagCriteria,  
true, cbf,exaleadClient);

## EJEMPLO DE INVOCACIÓN SÍNCRONA

```
...
using ExaleadConsoleApplication.ExaleadServiceReference;

static void Main(string[] args) {

    Petitioner petitioner = getPetitioner();
    SearchType searchType = getSearchType();
    Query query = getQuery();
    SortCriteria sortCriteria = getSortCriteria();
    PaginationCriteria pagCriteria = getPagCriteria();
    try {

        Console.WriteLine("Inicio del proceso...");

        //Instancia del Web Service IndexSearch
        NetExaleadWebServiceClient.ExaleadWebServiceRef.ExaleadWebServiceSoapClient exaleadClient =
        new NetExaleadWebServiceClient.ExaleadWebServiceRef.ExaleadWebServiceSoapClient();
        //Invocación al Servicio Web de forma síncrona
        Console.WriteLine("Invocar al servicio de búsqueda de IndexSearch...");    ResponseSummary
        responseSummary =
            exaleadClient.indexSearch(petitioner, searchType, query, sortCriteria, pagCriteria, true);

        //Finalización de la operación síncrona
        Console.WriteLine("Búsqueda finalizada.");
        Console.WriteLine("Proseguimos con las siguientes tareas...");
        System.Threading.Thread.Sleep(10000);
        Console.WriteLine("Fin de las tareas.");

        // Tratamiento de Excepciones
    } catch (Exception e) {
        Console.WriteLine("Error:" + e.Message);
        Console.WriteLine("Detalles:" + e.ToString());    return;
    }
}
```

## INVOCACIÓN AL SERVICIO WEB INDEXSEARCH DE FORMA ASÍNCRONA

Otra de las formas de invocar al cliente es de manera asíncrona, así, no será necesario por parte del proceso que realiza la petición o peticiones, esperar a que el Servicio Web de indexSearch retorne el resultado. Para ello se define una función de CallBack en la cual se delegará la espera de la búsqueda.

Esta función de CallBack se invocará a la finalización de la búsqueda, y será la encargada de realizar el tratamiento del resultado de los datos de la operación.

Para realizar este tipo de operativa, será necesario generar una aplicación cliente y obtener referencias del servicio asíncronas.

Al tomar referencias asíncronas, la definición del Servicio Web de IndexSearch, va a ser transformado en dos métodos:

- **Método Begin**, iniciará la comunicación asíncrona y retornará inmediatamente, el formato va a ser:

- **Método End**, indicará la finalización asíncrona del mensaje devolviendo el resultado de la llamada, el formato va a ser:  
`<exaleadClient>.End< indexSearch >.`

Los pasos para realizar el cliente de forma asíncrona se detallan a continuación

## REFERENCIAS AL SERVICIO

---

1. Indicar referencia al Servicio Web de IndexSearch

```
using NetExaleadWebServiceClient.ExaleadWebServiceRef;
```

## TAREA PRINCIPAL

---

1. Definición de función de Callback, que será invocada cuando termine la petición de búsqueda

```
AsyncCallback cbf = new AsyncCallback(callbackFnc);
```

2. Instanciación del Servicio Web IndexSearch.

```
NetExaleadWebServiceClient.ExaleadWebServiceRef.ExaleadWebServiceS  
oapClient exaleadClient = new  
NetExaleadWebServiceClient.ExaleadWebServiceRef.ExaleadWebServiceS oapClient ();
```

3. Invocación al Servicio Web de forma asíncrona:

La llamada al servicio va a incluir:

- Parámetros de entrada:
  - ✓ Parámetros por referencia de la petición del servicio
  - ✓ Nombre de la función de callback
  - ✓ Object que permite al proceso que invoca tener referencias sobre su estado.
- Retorno de la función
  - ✓ Valor de retorno como objeto de la interfaz de tipo IAsyncResult.

```
IAsyncResult ar = exaleadClient.BeginindexSearch(petitioner, searchType, query,  
sortCriteria, pagCriteria, true, cbf, exaleadClient);
```

4. Posibilidad de controlar cuando la tarea esta completada  
`ar.IsCompleted`

## FUNCIÓN DE CALLBACK

---

1. En la función de Callback, se obtiene una instancia del Cliente a través de la interfaz IAsyncResult

```
NetExaleadWebServiceClient.ExaleadWebServiceRef.ExaleadWebServiceS  
oapClient exaleadClient = (  
NetExaleadWebServiceClient.ExaleadWebServiceRef.ExaleadWebServiceS  
oapClient)ar.AsyncState;
```

2. Obtenemos los datos y tratamiento de los mismos.  
`ResponseSummary responseSummary = exaleadClient.EndindexSearch (ar);`

## EJEMPLO DE INVOCACIÓN ASÍNCRONA

---

```
...
using ExaleadConsoleApplication.ExaleadServiceReference;

static void Main(string[] args) {

    Petitioner petitioner = getPetitioner();
    SearchType searchType = getSearchType();
    Query query = getQuery();
    SortCriteria sortCriteria = getSortCriteria();
    PaginationCriteria pagCriteria = getPagCriteria();
    try {

        Console.WriteLine("Realizando tarea principal...");
        // Referencia a la función de callback
        AsyncCallback cbf = new AsyncCallback(callbackFnc);
        //Instancia del Web Service IndexSearch
        NetExaleadWebServiceClient.ExaleadWebServiceRef.ExaleadWebServiceSoapClient exaleadClient =
new NetExaleadWebServiceClient.ExaleadWebServiceRef.ExaleadWebServiceSoapClient();
        //Invocación al Servicio Web de forma asincrona
        Console.WriteLine("Invocar al inicio del servicio de búsqueda de IndexSearch..."); IAsyncResult ar =
exaleadClient.BeginindexSearch(petitioner,
                                searchType, query, sortCriteria, pagCriteria, true, cbf,exaleadClient);
        //Realización de tareas mientras se realiza la búsqueda
        Console.WriteLine("Proseguimos con nuestra tarea principal...");

        int start = DateTime.Now.Second; int
currentSecond = start;

        // Controlar cuando esta completada la tarea
        Console.WriteLine("Comprobar si la búsqueda ha finalizado...");
        while (!ar.IsCompleted) {
            if (currentSecond < DateTime.Now.Second) { currentSecond =
DateTime.Now.Second;
                Console.WriteLine("Espera..." + (currentSecond - start).ToString());
            }
        }

        //Finalización de la operación asincrona
        Console.WriteLine("Búsqueda finalizada.");
        Console.WriteLine("Proseguimos en la tarea principal...");

        //Realizamos funcionalidad de la tarea.
        System.Threading.Thread.Sleep(10000);
        Console.WriteLine("Fin de la tarea principal.");
    }
}
```



```

return;
}

// Definición de función de call-back, invocada cuando la operación asincrona se ha completado.
static void callbackFnc(IAsyncResult ar) {

    string results;
    try{
        // Obtención de instancia del Cliente a través de la interfaz IAsyncreult
        NetExaleadWebServiceClient.ExaleadWebServiceRef.ExaleadWebServiceSoapClient exaleadClient =
        ( NetExaleadWebServiceClient.ExaleadWebServiceRef.ExaleadWebServiceSoapClient)ar.AsyncState;

        // Completamos los datos.
        ResponseSummary responseSummary = exaleadClient.EndindexSearch(ar);

        //Output the results.
        Console.WriteLine("Obtenidos resultados y tratamiento de estos.");
        Console.WriteLine("Resultado de la búsqueda:" + responseSummary.ToString());
        // Tratamiento de Excepciones
    } catch (Exception e) {
        Console.WriteLine("Error:" + e.Message);
        Console.WriteLine("Detalles:" + e.ToString());
        return;
    }
}

// Tratamiento de Excepciones
} catch (Exception e) {
    Console.WriteLine("Error:" + e.Message);
    Console.WriteLine("Detalles:" + e.ToString());
}

```