OSGeo FDO y Apache Tika: construyendo una plataforma para la descripción de recursos multimedia

Beltran, Arturo; Granell, Carlos; Huerta, Joaquín

Resumen

La información geográfica juega un papel fundamental en la sociedad actual, y el interés de los usuarios por ella crece día a día. Sin embargo, aún resulta demasiado complicado encontrar contenidos geográficos que sean relevantes (actualizados, de calidad y veraces), pese a los esfuerzos realizados en generar grandes catálogos de metadatos.

Para hacer que la información esté disponible a nivel global y llegue fácilmente al mayor número de personas posible resulta esencial organizar, publicitar y facilitar el acceso a dicha información. Y para que esto sea posible, es decir, para que un recurso sea encontrado como resultado de una búsqueda es necesario describirlo según sus propiedades. Es en este contexto donde los metadatos cobran sentido y se convierten en la pieza central de cualquier sistema de información.

Con el objetivo de conseguir descripciones de los recursos, se analizaron diferentes herramientas de extracción de metadatos. Se consideró como la propuesta más interesante la del proyecto Apache Tika. Apache Tika es un conjunto de herramientas para detectar y extraer metadatos y texto estructurado del contenido de varios tipos de documentos usando librerías de parseo existentes. Actualmente, soporta diferentes formatos de texto, audio, imagen y video. Pero tiene una arquitectura que permite ampliar los formatos soportados mediante lo que ellos llaman proveedores. Resulta una interesante plataforma común de extracción de metadatos para recursos multimedia, el problema es que no soporta formatos de información geoespacial.

En consecuencia, se buscaron y evaluaron diferentes plataformas comunes de acceso a datos geográficos. Entre las diferentes opciones analizadas, se eligió la plataforma FDO. FDO (FDO Data Access Technology) es una API desarrollada por OSGeo para la manipulación, la definición y análisis de información geoespacial, independientemente de dónde se encuentre almacenada. FDO utiliza un modelo basado en proveedores para soportar gran variedad de fuentes de datos geoespaciales.

En este trabajo se pretende conseguir una plataforma que permita obtener información y acceder de forma lo más homogénea posible a recursos heterogéneos para poder extraer metadatos. Viendo lo que ofrecen los proyectos OSGeo FDO y Apache Tika, se quiere ir un paso más adelante y se pretende conseguir una plataforma de extracción de metadatos para todo tipo de recursos multimedia, con especial interés en los recursos geoespaciales. Este sería el primer paso para la descripción de los recursos, mediante la cual, posteriormente se podrá abordar la publicación, ya sea mediante la indexación respecto a sus características o la inclusión en un servidor de catálogo.

PALABRAS CLAVE

Metadatos, Extracción, Multimedia, OSGeo FDO, Apache Tika

1. INTRODUCCIÓN

Hoy en día la información juega un papel fundamental en la sociedad donde vivimos, llegando incluso al punto de la dependencia. Esto ha motivado, y ha sido motivado por, la era digital en la que nos encontramos inmersos. La cantidad de sistemas de información que manejamos actualmente es incontable: Bibliotecas Digitales, Sistemas de Información Geográfica (SIG)/Infraestructuras de Datos Espaciales (IDE), directorios y buscadores de internet, etc. Todos ellos motivados por el deseo de que la información esté disponible a nivel global y llegue fácilmente al mayor número de personas posible en un entorno colaborativo. Para ello resulta esencial organizar, publicitar y facilitar el acceso a dicha información.

Es en este contexto en el que los metadatos cobran sentido y resultan ser de gran importancia, pues para que un recurso sea encontrado como resultado de una búsqueda debemos ser capaces de describirlo según sus propiedades. Por lo tanto, los metadatos juegan un papel fundamental en cualquier sistema de información que podamos imaginar. Permitiéndonos indexar o catalogar los recursos en base a la descripción de sus características (tipo de dato, contenido, origen, calidad, fecha de creación, etc.) y a su contexto, para posteriormente poder ser encontrados. El problema reside en que la tarea de generación de metadatos resulta tediosa y poco gratificante, siendo necesario dedicar gran cantidad de tiempo y recursos tanto económicos como humanos. Por ello se considera necesario investigar técnicas y metodologías que permitan generar la mayoría de estos metadatos de forma automática.

Cuando nos encontramos ante un recurso desconocido, lo primero que podemos hacer es examinarlo y acceder a su contenido y extraer tanta información como nos sea posible tanto del propio recurso y su contexto como de su contenido. Pero esto no siempre resulta fácil, ya que la extracción automática metadatos implica el conocimiento de las estructuras internas de los datos que utilizan los formatos de almacenamiento de la Información Geográfica (IG) y realizar, para cada uno de ellos, una correspondencia con los distintos elementos que componen un metadato de acuerdo a un determinado estándar (DublinCore[1], ISO19115[2]…). Además, debemos tener en cuenta que el elevado número de formatos de almacenamiento existentes para IG dificulta que un misma aplicación pueda manejarlos todos y, que por tanto, se deban desarrollar soluciones integradoras que combinen las capacidades de lectura de la meta-información de distintas librerías para los múltiples formatos.

Por otra parte, si no nos conformamos solamente con la IG y queremos generalizar el proceso de extracción de metadatos para cualquier tipo de recurso multimedia el problema se agrava dada la enorme cantidad de formatos con los que vamos a tener que lidiar. Por ello, como veremos en este artículo, se ha realizado un estudio de herramientas y plataformas que facilitan el acceso a datos y metadatos que nos permitan describir los recursos. Por lo tanto, el objetivo de este trabajo es lograr una plataforma que nos permita obtener información y acceder de forma lo más homogénea posible a recursos heterogéneos.

2. ESTUDIO DE HERRAMIENTAS PARA EL ACCESO A DATOS Y METADATOS

Como se ha comentado en la introducción, la necesidad de acceder y obtener información de tan gran cantidad de formatos ha motivado la realización de un estudio que analiza y evalúa diferentes plataformas comunes de acceso a datos de IG y de diferentes herramientas para la extracción de metadatos. A continuación se detallan las soluciones más destacadas:

GeoTools

GeoTools[3] es un conjunto de librerías de código libre (LGPL)[4] escritas en Java[5] que proporcionan métodos adaptados a los estándares para la manipulación de datos geoespaciales, para, por ejemplo, implementar Sistemas de Información Geográfica (SIG). Proporciona una implementación de las especificaciones del *Open Geospatial Consortium* (OGC)[6] tal y como éstas se van desarrollando.

Aunque sus principales características y las primeras pruebas de extracción de metadatos hicieron ver este proyecto como una buena solución, al intentar ampliar el rango de formatos de recursos soportados no resultó una solución tan satisfactoria.

[1] http://dublincore.org
[2] http://www.iso.org/iso/catalogue_detail.htm?csnumber=39229
[3] http://www.geotools.org
[4] http://www.opensource.org/licenses/lgpl-license.php
[5] http://www.java.com
[6] http://www.opengeospatial.org

DAL de gvSIG

Es la capa de acceso a datos de *gvSIG*[7] denominada *Data Access Library* (DAL) tras la reingeniería realizada para la versión 2.0. Con DAL se pretende dotar a *gvSIG* de una capa de abstracción que permita al núcleo de la aplicación operar de forma homogénea con diferentes fuentes y formatos de datos. Su arquitectura se basa en la flexibilidad y la robustez, y por ello aspectos como el desacoplamiento y la trazabilidad son esenciales en su diseño. Los componentes que formarían parte de la librería de acceso a datos serían: la librería de acceso a datos, los distintos almacenes o proveedores de datos y las operaciones de manejo de leyendas específicos de algunos formatos.

En un principio, se decidió optar por lo que ofrece el proyecto *gvSIG* y reutilizar su capa de acceso a datos (DAL) en este proyecto. Sin embargo, para que el DAL de *gvSIG* sea la plataforma de acceso a datos que se desea habrá que trabajar mucho en ella y habrá que ampliar su funcionalidad, especialmente en la parte relacionada con proporcionar información (metadatos) sobre el recurso. En este sentido, se realizó un estudio y posterior propuesta para ampliar dicha funcionalidad. Además, aún falta por integrar gran parte de los proveedores de datos, entre ellos todos los que proporcionan el acceso a datos de naturaleza ráster.

Librerías GDAL/OGR

La *Geospatial Data Abstraction Library* (GDAL)[8] es una librería que permite traducir entre diferentes formatos geoespaciales de naturaleza ráster. Es desarrollada por la *Open Source Geospatial Foundation* (OSGeo)[9] bajo una licencia X11[10]/MIT[11] de código abierto. Como librería, presenta un único modelo de datos abstracto mediante el cual cualquier aplicación puede acceder a los diferentes formatos soportados. Además, viene con una amplia variedad de utilidades de línea de comandos para la traducción y el procesado de datos.

Por su parte, la *OGR Simple Feature Library* (OGR)[12] es una librería que proporciona funcionalidad y utilidades similares a GDAL pero para datos de naturaleza vectorial.

Estas librerías serían una buena base se quisiera empezar a desarrollar una plataforma común de acceso a datos geoespaciales. Pero si existiera algo un nivel por encima se ahorraría mucho esfuerzo.

FDO Data Access Technology

FDO Data Access Technology (FDO)[13] es una API para la manipulación, la definición y análisis de información geoespacial, independientemente de dónde se encuentre almacenada. FDO utiliza un modelo basado en proveedores para soportar gran variedad de fuentes de datos geoespaciales, donde cada proveedor normalmente soporta un formato de datos o almacén de datos en particular. FDO es desarrollada por OSGeo, siendo libre y de código abierto bajo la licencia LGPL.

La API de FDO proporciona una interfaz genérica basada en comandos para un gran número de formatos de datos para guardar, recuperar, actualizar y analizar información geoespacial. Además, FDO proporciona un modelo para extender su interfaz a otros formatos de datos. La API genérica, es extensible, y es posible añadir comandos personalizados a un proveedor particular. Un proveedor es la implementación especifica de de la API que proporciona acceso a los datos almacenados en un determinado formato.

Los principales proveedores que incorpora FDO son:

- **Provider for ArcSDE**[14]: Acceso de lectura/escritura a datos cuya fuente de datos se encuentra en el formato ArcSDE de ESRI[15].
- **Provider for MySQL**[16]: Acceso de lectura/escritura a datos cuya fuente de datos se encuentra en una base de datos MySQL. La arquitectura de MySQL soporta varios motores de almacenamiento, características y capacidades.

[7] http://www.gvsig.gva.es
[8] http://www.gdal.org
[9] http://www.osgeo.org
[10] http://en.wikipedia.org/wiki/X11_License
[11] http://www.opensource.org/licenses/mit-license.php
[12] http://www.gdal.org/ogr/index.html
[13] http://fdo.osgeo.org
[14] http://www.esri.com/software/arcgis/arcsde/index.html
[15] http://www.esri.com
[16] http://www.mysql.com

- **Provider for SDF**[17]: Acceso de lectura/escritura a datos cuya fuente de datos se encuentra en el formato SDF. Este formato geoespacial de Autodesk[18], soporta múltiples atributos, proporciona gran rendimiento para grandes conjuntos de datos e interoperabilidad con otros productos de Autodesk.
- **Provider for SHP**[19]: Acceso de lectura/escritura a datos cuya fuente de datos se encuentra en el formato SHP de ESRI, que consiste en archivos separados para geometría, índices y atributos. Cada archivo SHP y su archivo asociado DBF son tratados como una clase con una geometría común.
- **Provider for ODBC**[20]: Acceso de lectura/escritura a datos cuya fuente de datos se encuentra en una base de datos que ofrezca la API estándar ODBC.
- **Provider for WFS**[21]: Acceso de lectura a datos cuya fuente de datos se encuentra en un servicio WFS de OGC. Soporta un entorno cliente/servidor y recupera los datos geoespaciales codificados en GML de uno o más servicios.
- **Provider for WMS**[22]: Acceso de lectura a datos cuya fuente de datos se encuentra en un servicio WFS de OGC. WMS produce mapas de datos espacialmente referenciados de forma dinámica, los cuales son normalmente renderizados en los formatos PNG, GIF o JPEG.
- **Provider for GDAL**: Acceso de lectura a datos ráster cuya fuente de datos se encuentra en uno de los formatos soportados por la librería GDAL.
- **Provider for OGR**: Acceso de lectura/escritura a datos vectoriales cuya fuente de datos se encuentra en uno de los formatos soportados por la librería OGR.
- **SL-King FDO Provider for Oracle**[23]: Proporciona acceso a datos Oracle Spatial/Locator in Oracle 10G, Oracle XE and Oracle 9i.

En la Figura 1 se puede observar la arquitectura general del proyecto FDO y se aprecia que ofrece la funcionalidad que se desea. La única pega que se encuentra, a priori, es que la API que ofrece está basada en los lenguajes de programación C++[24] y .Net[25] y este desarrollo desde un principio se había planteado en Java.

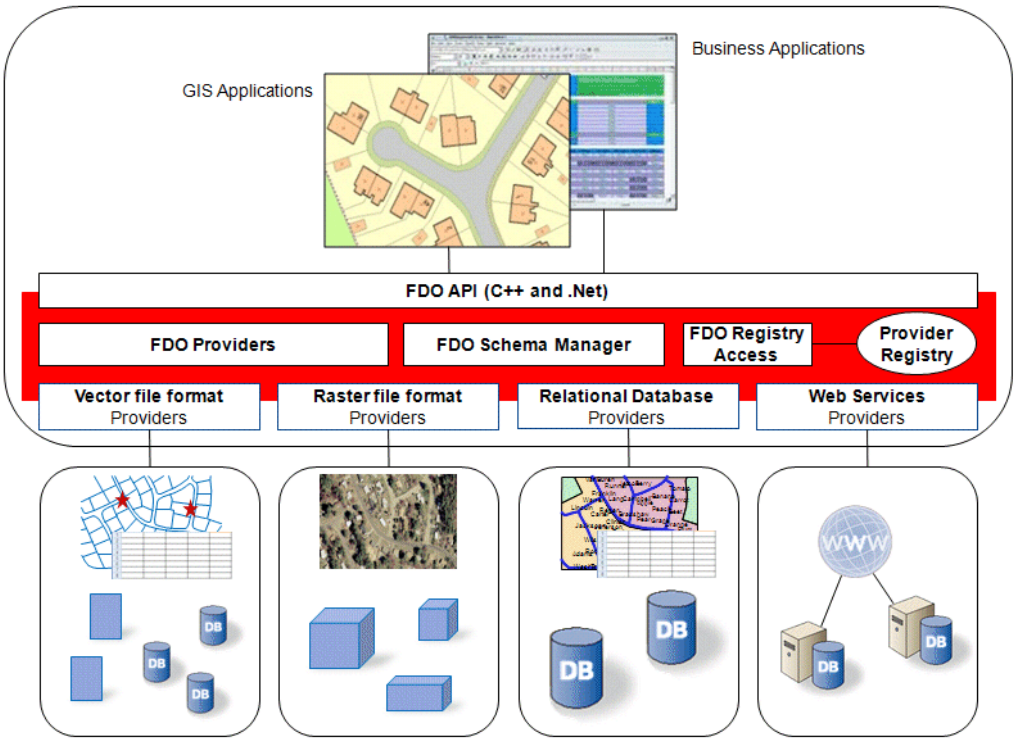

Figura 1: Arquitectura general de FDO.

[17] http://en.wikipedia.org/wiki/Spatial_data_file
[18] http://www.autodesk.com
[19] http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf
[20] http://en.wikipedia.org/wiki/Open_Database_Connectivity
[21] http://www.opengeospatial.org/standards/wfs
[22] http://www.opengeospatial.org/standards/wms
[23] http://www.oracle.com
[24] http://www.cplusplus.com
[25] http://www.microsoft.com/NET

CatMDEdit

La aplicación CatMDEdit[26] permite la gestión de recursos a través de los metadatos asociados a los mismos, prestando especial atención a la gestión y documentación de recursos de información geográfica. Entre sus características, encontramos la generación automática de metadatos para algunos formatos de datos. Y esa es la parte que se ha estudiado.

En la sección 3.4 del manual de usuario[27] de CatMDEdit aparece una tabla donde se analizan los elementos de metadatos extraídos para los distintos formatos soportados. Estos elementos son rellenados con valores obtenidos del análisis del contenido y formato de transferencia de esos datos. En la Tabla 1 se pueden ver los formatos de los cuales CatMDEdit es capaz de extraer información y la correspondencia de los metadatos extraídos con los campos ISO.

Campo ISO	SHP	DGN	ECW	FICC	GeoTIFF	GIF/ GFW	JPG/ JGW	PNG/ PGW
MD_Metadata.identificationInfo> MD_DataIdentification.spatialRepresentationType	X	X	X	X	X	X	X	X
MD_Metadata.identificationInfo> MD_DataIdentification.extent> EX_Extent.geographicElement> EX_GeographicBoundingBox.northBoundLatitude, EX_GeographicBoundingBox.SouthBoundLatitude, EX_GeographicBoundingBox.eastBoundLongitude, EX_GeographicBoundingBox.westBoundLongitude	X	X	X	X	X	X	X	X
MD_Metadata.contentInfo> MD_FeatureCatalogueDescription.featureTypes MD_Metadata.applicationSchemaInfo> MD_ApplicationSchemaInformation.schemaAsci	X	X		X				
MD_Metadata.applicationSchemaInfo> MD_ApplicationSchemaInformation.schemaAscii	X							
MD_Metadata.applicationSchemaInfo> MD_ApplicationSchemaInformation.schemaAscii	X							
MD_Metadata.spatialRepresentationInfo> MD_VectorSpatialRepresentation.geometricObjects> MD_GeometricObjects.geometricObjectType	X	X		X				
MD_Metadata.spatialRepresentationInfo> MD_VectorSpatialRepresentation.geometricObjects> MD_GeometricObjects.geometricObjectCount	X	X		X				
MD_Metadata.distributionInfo> MD_Distribution.transferOptions> MD_DigitalTransferOptions.onLine> CI_OnlineResource.linkage	X	X	X	X	X	X	X	X
MD_Metadata.distributionInfo> MD_Distribution.transferOptions> MD_DigitalTransferOptions.transferSize	X	X	X	X	X	X	X	X
MD_Metadata.distributionInfo> MD_Distribution.distributionFormat> MD_Format.name	X	X	X	X	X	X	X	X
MD_Metadata.spatialRepresentationInfo> MD_GridSpatialRepresentation.numberOfDimensions			X		X	X	X	X
MD_Metadata.spatialRepresentationInfo> MD_GridSpatialRepresentation.axisDimensionProperties> MD_Dimension.dimensionName			X		X	X	X	X
MD_Metadata.spatialRepresentationInfo> MD_GridSpatialRepresentation.axisDimensionProperties> MD_Dimension.dimensionSize			X		X	X	X	X

Tabla 1: Metadatos extraídos automáticamente por CatMDEdit

Apache Tika

Apache Tika[28] es un conjunto de herramientas para detectar y extraer metadatos y texto estructurado del contenido de varios tipos de documentos usando librerías de parseo existentes. Tika es un proyecto de la *Apache Software Foundation*[29] escrito en Java y se distribuye bajo la licencia Apache v2.0[30].

Actualmente, soporta formatos como: HTML, XML y derivados, documentos de Microsoft Office, formato OpenDocument, PDF, EPUB, RTF, diferentes formatos de compresión, diferentes formatos de texto, audio, imagen y video, algunos archivos de Java y el formato mbox. En la siguiente figura (Figura 2) podemos ver los metadatos que Apache Tika es capaz de extraer de un documento de MSWord.

[26] http://catmdedit.sourceforge.net
[27] http://iaaa.cps.unizar.es/software/index.php/CatMDEdit_Spanish_user_manual
[28] http://lucene.apache.org/tika
[29] http://www.apache.org
[30] http://www.apache.org/licenses/LICENSE-2.0

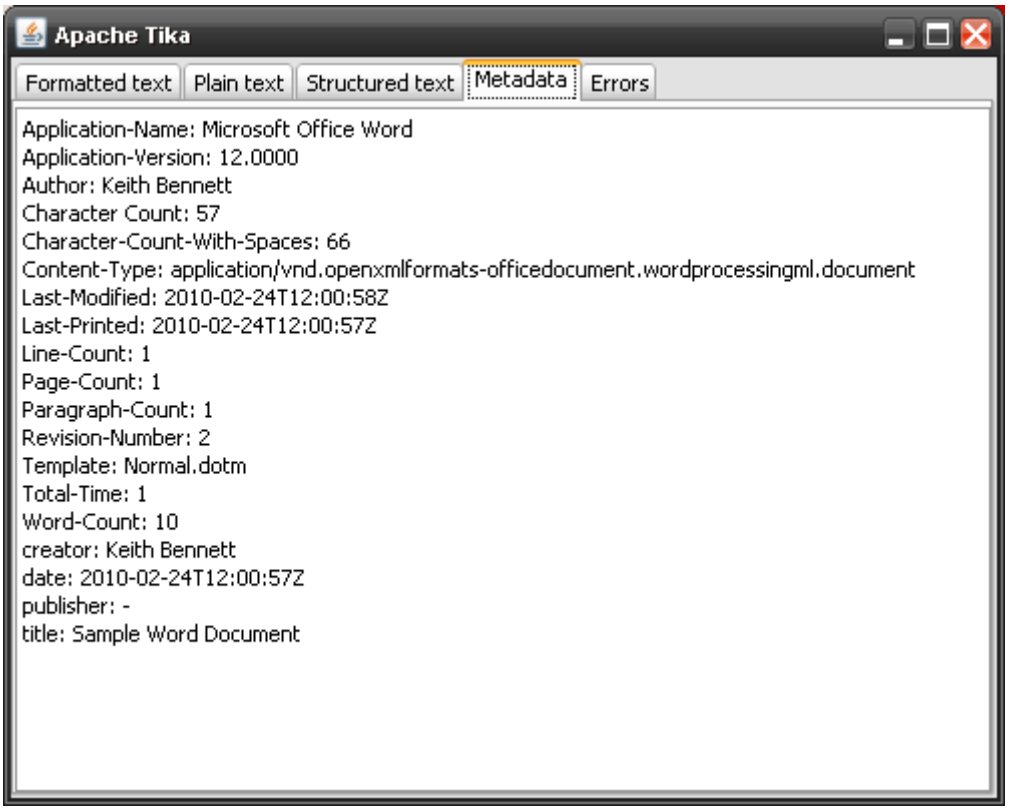

Figura 2: Ejemplo de metadatos extraídos de un documento de MSWord

Resulta interesante una plataforma común de extracción de metadatos para recursos multimedia. Pero como se puede observar no soporta formatos de información geoespacial.

3. INTEGRACIÓN DE FDO Y APACHE TIKA

El objetivo que se planteó fue conseguir una plataforma que nos permitiera obtener información y acceder de forma lo más homogénea posible a recursos heterogéneos para poder extraer metadatos. Viendo lo que nos ofrecen los proyectos FDO y Apache Tika, queremos ir un paso más adelante y pretendemos conseguir una plataforma de extracción de metadatos para todo tipo de recursos multimedia. Este sería el primer paso para la descripción de los recursos, mediante la cual, posteriormente podremos plantearnos diversos métodos de publicación, entre ellos la indexación respecto a sus características o la inclusión en un servidor de catálogo.

Como hemos comentado, la integración de FDO y Apache Tika nos permitirá conseguir una plataforma de extracción de metadatos para todo tipo de recursos multimedia, con especial interés en los recursos geoespaciales.

Apache Tika tiene una arquitectura que nos permite ampliar los formatos soportados mediante lo que ellos llaman proveedores, normalmente son *parsers* para formatos específicos. Por lo tanto, el desarrollo consistirá en añadir uno o varios *parsers* para que al detectar ciertos formatos de datos se realice la extracción de metadatos usando la API de FDO tal y como se puede apreciar en la Figura 3.

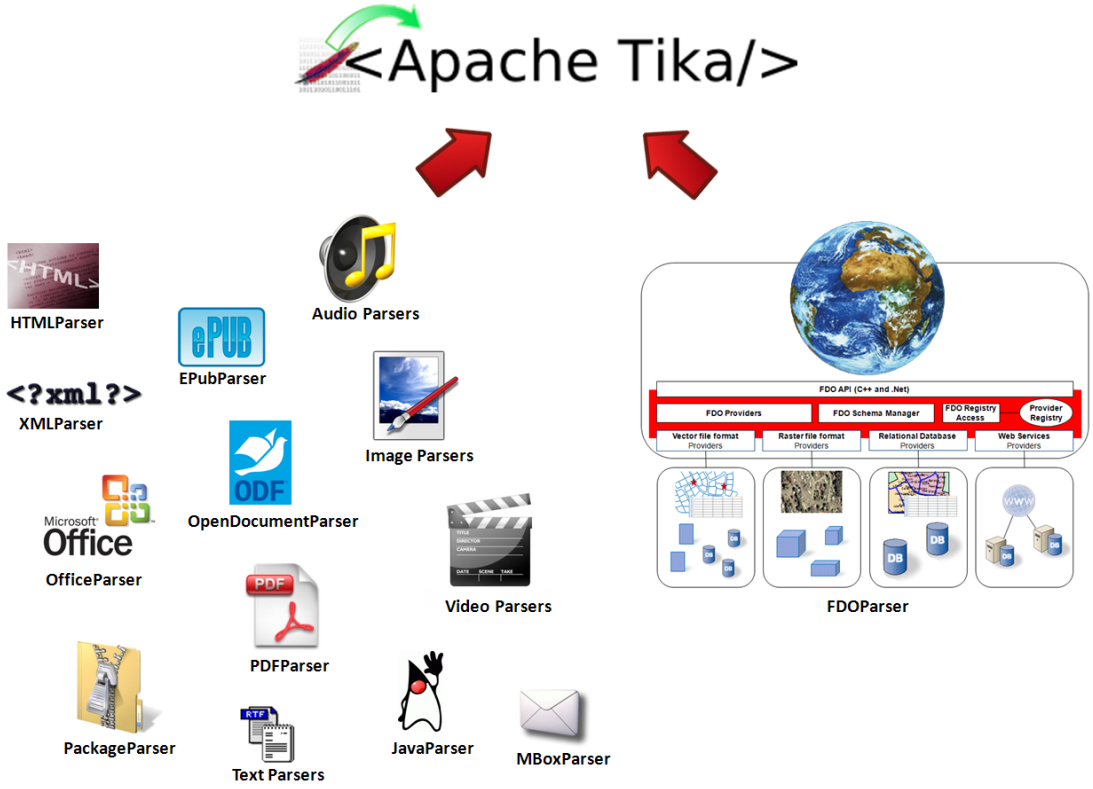

Figura 3: Integración de Apache Tika y FDO

Aunque conceptualmente puede parecer sencillo, el problema encontrado fue que la API que ofrece FDO se encuentra en los lenguajes de programación C++ y .Net, y el proyecto Apache Tika está desarrollado totalmente en Java.

La solución que se suele elegir en este tipo de situaciones es el uso de *Java Native Interface* (JNI)[31]. JNI es un *framework* de programación que permite que un programa escrito en Java ejecutado en la máquina virtual java (JVM) pueda interactuar con programas escritos en otros lenguajes como C o C++. Para facilitar la creación de código JNI, existen herramientas de desarrollo como el *Simplified Wrapper and Interface Generator* (SWIG)[32].

4. RESULTADOS

La integración de FDO y Apache Tika proporciona una plataforma de extracción de metadatos para todo tipo de recursos multimedia. En la Tabla 2 se pueden apreciar los tipos de recursos soportados inicialmente por Apache Tika y en la Tabla 3 los tipos de recursos que soporta FDO. Tras la integración, la nueva plataforma soportará todos esos tipos de recursos.

[31] http://java.sun.com/javase/6/docs/technotes/guides/jni/index.html
[32] http://www.swig.org

Parser	Formatos
HtmlParser	HTML, XHTML, ASP
XMLParser	XML, XML+SVG
OfficeParser & OOXMLParser	Documentos de Word, Excel, PowerPoint, Visio y mensajes de Outlook
OpenDocumentParser	Todo tipo de documentos ODF
PDFParser	PDF
EpubParser	Epub
RTFParser	RTF
PackageParser	GZIP, BZIP, BZIP2, CPIO, GTAR, GZIP, TAR, ZIP
TXTParser	Documentos de texto plano
AudioParser	AU, WAV, AIFF
MidiParser	MIDI
MP3Parser	MP3
ImageParser	BMP, GIF, PNG, TIFF, ICON, PSD, XCF
JpegParser	JPEG
FLVParser	FLV
ClassParser	Documentos Java
ZipParser	Archivos JAR
MboxParser	E-mails en format MBOX

Tabla 2: Formatos soportados por Apache Tika

Proveedor	Formatos
ArcSDE	ArcSDE de ESRI
SDF	SDF de Autodesk
SHP	SHP de ESRI
GDAL	Más de 108 formatos ráster Ver: http://www.gdal.org/formats_list.html
OGR	Más de 43 formatos vectoriales Ver: http://www.gdal.org/ogr/ogr_formats.html
MySQL	Conexión con bases de datos MySQL con soporte espacial (datos y consultas)
SL-King Oracle	Conexión con bases de datos Oracle Spatial
ODBC	Conexión con bases de datos con interfaz ODBC
WMS	Acceso a servicios WMS
WFS	Acceso a servicios WFS

Tabla 3: Formatos soportados por FDO

Una vez conocidos los tipos de recursos que soporta la plataforma, veamos el nivel de detalle de las descripciones que es capaz de proporcionar, esto es, los metadatos que consigue extraer de cada tipo de recurso.

Respecto a los *parsers* ya implementados en Apache Tika, la cantidad de metadatos que consiguen extraer depende directamente del tipo de recurso. Por ejemplo, en las siguientes figuras (Figura 4 y Figura 5) se pueden ver los metadatos que se han extraído de un documento de texto de Microsoft Word y de un archivo de audio en formato MP3.

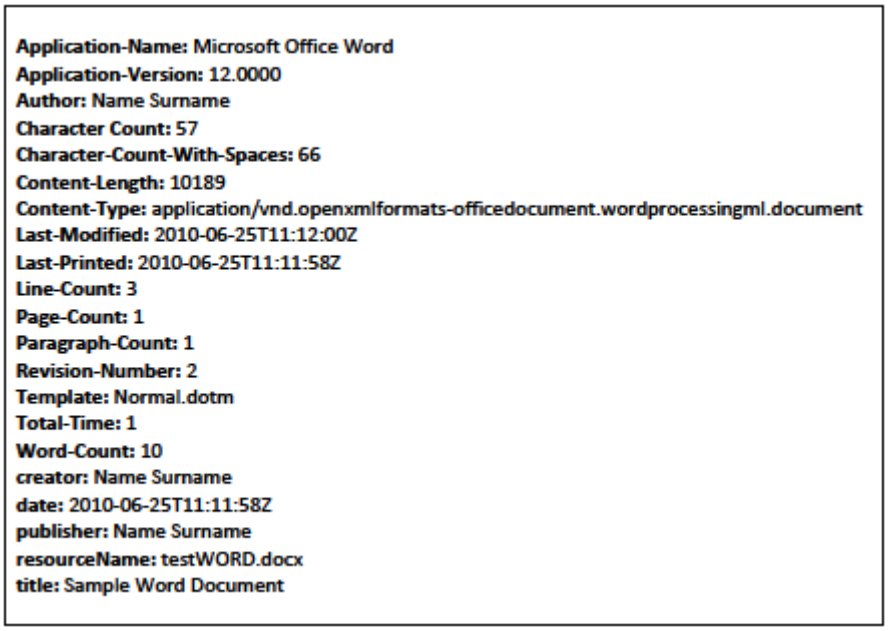

Figura 4: Metadatos extraídos por Apache Tika de un documento de MS Word

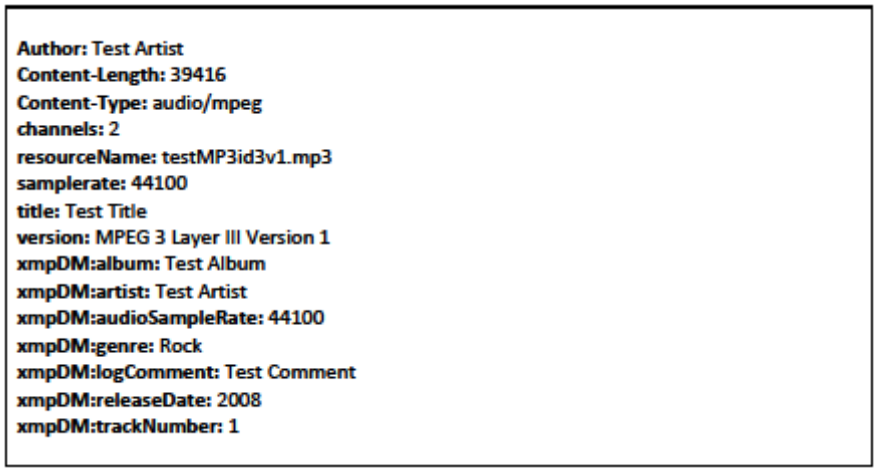

Figura 5: Metadatos extraídos por Apache Tika de un archivo de audio MP3

Respecto al nuevo *parser* desarrollado, dado que la API de FDO permite analizar información geoespacial independientemente de dónde se encuentre almacenada, los metadatos extraídos para la multitud de formatos que soporta tendrán una forma homogénea. Eso sí, reflejando las características específicas de cada formato. En la Tabla 4, se puede ver un resumen de los metadatos que el nuevo *parser* extrae mediante la API de FDO. Más adelante, en la Figura 7 se mostrará a modo de ejemplo una instancia más detallada de estos metadatos.

Etiqueta	Significado
Resource	Inicia la descripción de un recurso
FormatName	Formato en el que se encuentra el recurso analizado
ResourceType	Tipo de recurso
Provider	Proveedor de FDO utilizado para analizar el recurso
Source	Origen de los datos
ResourceName	Nombre del recurso
ConnectionString	Cadena de conexión al recurso
dateStamp	Fecha de la creación de los metadatos
keywords	Palabras clave en la descripción del recurso
SpatialContexts	Descripción de los contextos espaciales del recurso. Incluyendo nombre, sistema de coordenadas, extensión…
Schemas	Descripción de los esquemas de datos del recurso. Incluyendo nombre, atributos, *features*…
SchemaAttributes	Descripción de los atributos del esquema. Incluyendo nombre y valor de los mismos.
FeatureClasses	Descripción de los *features* del recurso. Incluyendo nombre, restricciones, sus propiedades…
BaseIdentityProperties y Properties	Descripción de las propiedades de cada *feature*. Incluyendo nombre, tipo y diferente información dependiente del tipo, como tamaño, valor por defecto, precisión…

Tabla 4: Resumen de los metadatos extraídos desde la API de FDO

Por otra parte, se ha desarrollado un módulo que permite incluir en las descripciones automáticas de los recursos ciertos metadatos preconfigurados. Este módulo basa su funcionamiento en un archivo de configuración en formato XML, en el cual se añadirán las nuevas etiquetas de los metadatos que se deseen incluir en las descripciones. De este modo, se consigue añadir cualquier información dependiente del contexto de una forma rápida y cómoda en todas las descripciones. Los metadatos basados en el contexto pueden incluir información tan relevante como el autor de los datos o la empresa a la que pertenecen entre otros. En la Figura 6 se puede ver un ejemplo escueto del posible contendido de este archivo de configuración, aunque se debe tener en cuenta que se puede incluir cualquier información que parezca relevante y se insertará en las descripciones de los recursos de forma automática (dentro de la etiqueta *PreConfiguredMD*).

```
<?xml version="1.0" encoding="UTF-8"?>
<config>
  <MDLanguage>en</MDLanguage>
  <characterSet>utf8</characterSet>
  <useLimitation>Not to be used for commercial purposes</useLimitation>
  <accessConstraints>copyright</accessConstraints>
  <otherConstraints></otherConstraints>
  <metadata>
    <type>Autogenerated</type>
    <author>FDO API test</author>
  </metadata>
</config>
```

Figura 6: Ejemplo del archivo XML para metadatos preconfigurados

Finalmente, en la Figura 7 se puede apreciar un ejemplo de las descripciones de recursos que se pueden conseguir una vez integrados Apache Tika y FDO. Hay que tener en cuenta, que la descripción incluye los metadatos que se han conseguido extraer de un archivo de información vectorial en formato SHP usando el desarrollo basado en la API de FDO, además de los preconfigurados.

```
<?xml version="1.0" encoding="UTF-8"?>
<Resource>
  <PreConfiguredMD>
    <MDLanguage>en</MDLanguage>
    <characterSet>utf8</characterSet>
    <uselimitation>Not to be used for commercial purposes</uselimitation>
    <accessConstraints>copyright</accessConstraints>
    <otherConstraints/>
    <metadata>
      <type>Autogenerated</type>
      <author>FDO API test</author>
    </metadata>
  </PreConfiguredMD>
  <FormatName>SHP</FormatName>
  <ResourceType>Vectorial</ResourceType>
  <Provider>SHP</Provider>
  <Source>C:\ExampleData\shp_world_countries\country.shp</Source>
  <ResourceName>country.shp</ResourceName>
  <ConnectionString>DefaultFileLocation=
            C:\ExampleData\shp_world_countries\country.shp;</ConnectionString>
  <dateStamp>22/06/2010 0:00:00</dateStamp>
  <dateType>creation</dateType>
  <keywords>Vectorial, SHP, SHP, country.shp</keywords>
  <SpatialContexts>
    <SpatialContext>
      <Name>GCS_WGS_1984</Name>
      <CoordinateSystem>GCS_WGS_1984</CoordinateSystem>
      <CoordinateSystemWkt>GEOGCS["GCS_WGS_1984",DATUM["D_WGS_1984",
                          SPHEROID["WGS_1984",6378137,298.257223563]],
                          PRIMEM["Greenwich",0],
                          UNIT["Degree",0.0174532925199433]]
      </CoordinateSystemWkt>
      <Description/>
      <Geometry>OSGeo.FDO.Geometry.IPolygonImp</Geometry>
      <Extent>POLYGON ((-180 -90, 180 -90, 180 83,6235961914063,
              -180 83,6235961914063, -180 -90))</Extent>
      <ExtentType>SpatialContextExtentType_Dynamic</ExtentType>
      <XYTolerance>0,001</XYTolerance>
      <ZTolerance>0,001</ZTolerance>
    </SpatialContext>
  </SpatialContexts>
  <Schemas num="1">
    <Schema>
      <SchemaName>Default</SchemaName>
      <SchemaQualifiedName>Default</SchemaQualifiedName>
      <SchemaDescription>Default schema.</SchemaDescription>
      <SchemaAttributes num="0"/>
      <FeatureClasses num="1">
        <FeatureClass>
          <FeatureClassName>country</FeatureClassName>
          <FeatureClassAttrCount>0</FeatureClassAttrCount>
          <FeatureClassDescription/>
          <FeatureClassIsComputed>False</FeatureClassIsComputed>
          <FeatureClassUniqueConstraintsCount>0
          </FeatureClassUniqueConstraintsCount>
          <BaseIdentityProperties num="0"/>
```

```
          <Properties>
            <Property>
              <PropertyName>FeatId</PropertyName>
              <PropertyDescription/>
              <PropertyRefCount>3</PropertyRefCount>
              <PropertyType>PropertyType_DataProperty</PropertyType>
              <PropertyDataType>DataType_Int32</PropertyDataType>
              <PropertyAttrLength>0</PropertyAttrLength>
              <PropertyLength>0</PropertyLength>
              <PropertyDefaultValue/>
              <PropertyNullable>False</PropertyNullable>
              <PropertyPrecision>0</PropertyPrecision>
              <PropertyScale>0</PropertyScale>
              <PropertyValueConstraint/>
            </Property>
            <Property>
              <PropertyName>FIPS_CNTRY</PropertyName>
              <PropertyDescription/>
              <PropertyRefCount>2</PropertyRefCount>
              <PropertyType>PropertyType_DataProperty</PropertyType>
              <PropertyDataType>DataType_String</PropertyDataType>
              <PropertyAttrLength>0</PropertyAttrLength>
              <PropertyLength>2</PropertyLength>
              <PropertyDefaultValue/>
              <PropertyNullable>True</PropertyNullable>
              <PropertyPrecision>0</PropertyPrecision>
              <PropertyScale>0</PropertyScale>
              <PropertyValueConstraint/>
            </Property>
-
            <Property>
              <PropertyName>Geometry</PropertyName>
              <PropertyDescription/>
              <PropertyRefCount>3</PropertyRefCount>
              <PropertyType>PropertyType_GeometricProperty</PropertyType>
              <PropertyGeometryTypes>4</PropertyGeometryTypes>
              <PropertyAttrLength>0</PropertyAttrLength>
              <PropertyHasElevation>False</PropertyHasElevation>
              <PropertyHasMeasure>False</PropertyHasMeasure>
              <PropertySpatialContextAssociation>GCS_WGS_1984
              </PropertySpatialContextAssociation>
              <PropertySpecificGeometryTypesLength>1
              </PropertySpecificGeometryTypesLength>
            </Property>
          </Properties>
        </FeatureClass>
      </FeatureClasses>
    </Schema>
  </Schemas>
</Resource>
```

Figura 7: Ejemplo de descripción de recurso de tipo SHP

5. CONCLUSIONES

Teniendo en mente el objetivo final de facilitar al usuario el acceso a los recursos, el primer paso para conseguirlo es la descripción de los recursos, en este caso mediante la generación automática de metadatos, para su posterior catalogación o indexación que permita organizar los recursos y finalmente publicarlos para poder ser encontrados.

Tras estudiar varias posibles soluciones, en este trabajo se han integrado los proyectos FDO y Apache Tika como capa de acceso a datos que posibilita la extracción de metadatos que permitan describir todo tipo de recursos multimedia. Observando los resultados conseguidos se puede concluir que la nueva plataforma da soporte para describir recursos multimedia de una gran variedad de formatos e incluso servicios, especialmente de formatos de IG. Por lo tanto, se considera que se ha cumplido el objetivo inicial de este trabajo, esto es lograr una plataforma que nos permita obtener información y acceder de forma lo más homogénea posible a recursos heterogéneos.

Actualmente, resulta muy complicado conseguir un sistema que permita la descripción de recursos totalmente autónomo, pues siempre será necesaria la participación del usuario para introducir o por lo menos validar los campos de metadatos menos intuitivos, hay que tener en cuenta que no todos los datos son fáciles de averiguar, por ejemplo el resumen o el título. Deberemos empezar por rellenar los campos básicos de descubrimiento de forma que se puedan ejecutar búsquedas mínimas con éxito, por ejemplo, en un catálogo. Más tarde podremos dedicar esfuerzos a completar rigurosamente el metadato. Es preferible tener todos los metadatos "a medias" que "atascarse" intentando rellenar exhaustivamente uno de ellos.

En este sentido, las descripciones conseguidas para los recursos de información geográfica en base a los metadatos extraídos de forma automática parecen ser bastante completas. Si a esto se le suma la participación del usuario a la hora de incluir más metadatos, ya sean como metadatos relativos al contexto preconfigurados o rellenando a mano los metadatos menos intuitivos, se puede conseguir un sistema que facilite en gran medida las rutinarias y poco motivadoras labores de los creadores de metadatos, reduciendo además los errores que se producen al escribir directamente los metadatos.

Finalmente, se considera esencial impulsar la investigación en todos los campos relacionados con la generación y gestión de metadatos dado el papel clave que estos juegan en cualquier sistema de información. Estos metadatos permiten indexar y catalogar los recursos de una forma más exacta en los sistemas de información y, en consecuencia, aumentan la capacidad de proporcionar resultados más relevantes y exactos a las búsquedas realizadas por los usuarios. Con todo esto, se consigue mejorar la accesibilidad a la información, el objetivo principal.

6. CONTACTOS

Arturo BELTRAN
arturo.beltran@uji.es
Universitat Jaume I (UJI)
Institute of New Imaging Technologies (INIT)

Carlos GRANELL
carlos.granell@uji.es
Universitat Jaume I (UJI)
Institute of New Imaging Technologies (INIT)

Joaquín HUERTA
huerta@uji.es
Universitat Jaume I (UJI)
Institute of New Imaging Technologies (INIT)