

# Modelo de datos para la monitorización de cambios geométricos del parcelario catastral

Miguel Fernández Moreno <sup>1</sup>, Alfredo Ramón Morte <sup>2</sup>

---

## Abstract

El Catastro Inmobiliario español es un registro administrativo dependiente del Ministerio de Hacienda y Administraciones Públicas en el que se describen los bienes inmuebles rústicos, urbanos y de características especiales. En la actualidad ofrece información sobre las parcelas, aportando tanto datos alfanuméricos como geométricos. Además, esta información puede ser consultada mediante servicios web basados en protocolos estándar para intercambio de información cartográfica (WMS, WFS, entre otros), y otros protocolos para el intercambio de información alfanúmerica (p. ej. SOAP). Estos servicios proporcionan una vista estática de la realidad de una parcela, sin embargo, no permiten obtener información sobre el estado de las parcelas previo a una alteración catastral (agregación, segregación ó división). Como solución a este problema, la presente investigación propone un modelo de datos que puede ser utilizado en diversos tipos de aplicaciones relacionados con la estructura de la propiedad (Catastro, Registro de la propiedad, Notarías, etc). Este modelo definido en lenguaje universal de modelado (UML), e implementado sobre gestor de base de datos orientada a grafos (Neo4j), permite almacenar y consultar ese histórico, pudiendo ser explotado posteriormente tanto por aplicaciones de escritorio como por servicios web.

*Keywords:* Catastro, NoSQL, Bases de datos de grafos, geometría catastral, OGC, Neo4j

---

## 1. Introducción

El Catastro Inmobiliario español es un registro administrativo dependiente del Ministerio de Hacienda y Administraciones Públicas en el que se describen los bienes inmuebles rústicos, urbanos y de características especiales. Está regulado por el texto refundido de la Ley del Catastro Inmobiliario del 2003, y la inscripción de propiedades inmuebles en el mismo es obligatoria y gratuita, características que lo diferencian del Registro de la Propiedad.

La descripción catastral de los bienes inmuebles incluye sus características físicas, jurídicas y económicas, entre las que se encuentran su localización, referencia catas-

---

<sup>1</sup>Alumnodel máster de Software Libre de la UOC

<sup>2</sup>Director del proyecto final del presente trabajo

tral, superficie, uso, cultivo, representación gráfica, valor catastral y titular catastral (Dirección General de Catastro, 2012).

Desde el año 2007, la información almacenada por Catastro se ve condicionada por la Infrastructure for Spatial Information in Europe (INSPIRE) que es una iniciativa lanzada en el año 2001 por la Comisión Europea y desarrollada en colaboración con los Estados miembros y los países candidatos a la adhesión.

Según tal documento, los objetivos de INSPIRE son (INSPIRE, 2012):

- Poner a disposición de órganos responsables de toma de decisiones o aplicación de políticas comunitarias (esencialmente de Medio Ambiente) datos espaciales abundantes y fiables.
- Establecer servicios integrados de Información Geográfica (IG), basados en una red distribuida de bases de datos, enlazadas por normas comunes y protocolos que aseguren la interoperabilidad.
- Optimizar los datos disponibles mediante documentación de información espacial.
- Lograr la coherencia de la información espacial entre diferentes niveles y temas.
- Crear servicios destinados a mejorar, accesibilidad e interoperabilidad de los datos y a la eliminación de obstáculos para su utilización

Según INSPIRE la parcela catastral se define como un área individual de superficie de la tierra (terreno y/o agua), sujeta a derechos reales de propiedad homogéneos y de titularidad única. Como novedad a destacar en la directiva INSPIRE aplicada a catastro es la introducción de que cada parcela presente además del identificador correspondiente a cada registro nacional catastral, un identificador único exigido por INSPIRE y otros elementos de interés no obligatorios, introduciéndose la noción de conjunto de índices catastrales, índices que permiten a cada país vincular las parcelas catastrales con otro tipo de áreas territoriales de interés (como unidades administrativas, construcciones) propias de la realidad de cada estado miembro de la unión europea sujeto a INSPIRE (Consejo Superior Geográfico, 2012).

En Catastro, esta realidad empieza a tomar forma, sobre todo, a partir del año 2002, cuando se propone como directiva europea INSPIRE. Directiva que pretende velar por la interoperabilidad de los datos geoespaciales en los países miembros de la Unión Europea. Siguiendo esta directiva, la Dirección General del Catastro ofrece en forma de servicios web, la Cartografía Catastral de forma libre y gratuita de los datos públicos. Esta directiva se adoptará finalmente en el año 2007.

El Open Geospatial Consortium es un consorcio de la industria internacional formado por 474 empresas, agencias gubernamentales y universidades que participan en un proceso de consenso para desarrollar estándares de interfaz disponibles al público. Los estándares otorgan una tecnología a los desarrolladores para elaborar información espacial compleja y servicios accesibles y útiles para todo tipo de aplicaciones (Open Geospatial Consortium, 2012)<sup>3</sup>. Entre esos estándares destacamos los servicios WMS

---

<sup>3</sup><http://www.opengeospatial.org>

(Web Map Service), WFS (Web Feature Service), WPS (Web Processing Service), entre otros muchos.

Los estándares definidos por el OGC siguen una línea temporal que han evolucionado y cristalizado en una serie de especificaciones, donde se explica cómo realizar su implementación, de manera que el primer documento publicado que especifica el WMS es del 2001, para el WFS del 2002, y ya más actual el WPS en el 2005.

El primero de estos estándares, el Web Map Service (WMS), proporciona una sencilla interfaz HTTP para solicitar imágenes de mapa georeferenciados de una o más bases de datos geoespaciales distribuidos. Una petición WMS define la capa geográfica (s) y el área de interés a ser procesados. La interfaz también permite la posibilidad de especificar si las imágenes enviadas deben ser transparentes para que las capas de varios servidores se pueden combinar o no (Open Geospatial Consortium, 2010).

En segundo lugar, el Web Feature Service (WFS), especifica el comportamiento de un servicio que permite realizar transacciones y tener acceso a elementos geográficos de manera independiente del almacén de datos subyacente (Open Geospatial Consortium, 2010). El estándar WFS permite lo siguiente a los clientes:

- Descubrir las colecciones de entidades disponibles (GetCapabilities).
- Describir los campos de atributos disponibles por entidades (DescribeFeatureType).
- Consultar una colección para un subconjunto de entidades basado en un filtro proporcionado (GetFeature).
- Añadir, editar o borrar entidades (Transaction).

En definitiva permite tener acceso a geometrías vectoriales y no sólo a una imagen como hace el WMS.

La Dirección General de Catastro siguiendo la línea impulsada por INSPIRE y usando los estándares desarrollados por el OGC, ha creado servicios para ser consumidos desde un navegador web o aplicaciones cliente de escritorio:

WMS, servicio de mapas en formato raster. Es un servicio libre y gratuito, pero no se puede editar. Es posible acceder a toda la información no protegida presente en Catastro. La petición "GETFEATUREINFO" del servicio WMS proporciona información sobre la Referencia Catastral de la parcela identificada, pudiéndose acceder con un hipervínculo a la información de acceso libre presente para esa parcela en la Oficina Virtual del Catastro (Alonso Peña et al., 2009).

WFS, servicio de mapas que publica cartografía en formato vectorial proporcionando un medio de gestión y visualización de datos geográficos a través de la red en un formato editable. Este servicio a diferencia del anterior no es accesible libremente, sino que requiere de registro previo donde se debe justificar su necesidad. Las peticiones se realizan acotando una zona por sus coordenadas geográficas (e incluso por fecha de modificación), obteniendo como resultado un fichero GML<sup>4</sup> (especificación XML)

---

<sup>4</sup>GML o Geography Markup Language es un estándar de codificación basada en XML para la información geográfica desarrollado por el OpenGIS Consortium (OGC). Permite el intercambio de datos geográficos de forma clara y transparente (Open Geospatial Consortium, 2012).

cuyo contenido gráfico vectorial puede integrarse en el correspondiente sistema del solicitante. Se consigue de esta forma satisfacer la demanda de información catastral de aquellos agentes colaboradores que precisan ir más allá de una imagen a superponer y su correspondiente acceso a Oficina Virtual del Catastro (Alonso Peña, 2006).

Un avance, dentro de los servicios del Catastro español, fue sobre todo, la implantación del servicio WFS, que nos permite descargar la cartografía catastral en formato vectorial. Al hacer la petición al servidor de una zona determinada, este devuelve un GML (Alonso Peña, 2006).

En ambos casos, la información muestra la realidad actual de la cartografía catastral, es decir, podemos consultar la información relativa a usos, superficie, geometría, referencia catastral, clase, etc, siempre datos de libre difusión o públicos (Cano et al., 2006).

A partir de la Ley de Economía Sostenible en 2011<sup>5</sup>, los servicios cartográficos ofrecidos por el Catastro Español toman mayor importancia. La interpretación de la nueva normativa debe hacerse “atendiendo fundamentalmente a su espíritu y finalidad” que, según el preámbulo de la citada ley 2/2011, es la de mejorar la coordinación con el Registro de la Propiedad; agilizar tramitaciones; reducir cargas administrativas a los ciudadanos mediante la colaboración de notarios y registradores; y mejorar, tras la intervención notarial, la conciliación entre la base de datos catastral y la realidad física inmobiliaria (Suárez Pinilla, 2012). Las principales novedades introducidas por la reforma son: certificación catastral descriptiva y gráfica, presunción de veracidad y, subsanación de discrepancias.

Este conjunto de normativas, iniciativas y estándares, constituyen el marco en el que se plantea el proyecto “Ramón Llull”, en el que se desarrolla una aplicación para gestionar las alteraciones catastrales, y en el que se pretende que, tanto las Notarías como los técnicos trabajen conjuntamente y se use la cartografía catastral de base sobre la que trabajar (Llorens Cobos et al., 2005). Este proyecto tiene en cuenta la parte jurídica y técnica, y por ello, el proyecto se desarrolla en distintas fases donde se crea un software FOSS<sup>6</sup> que cumple con todo lo anterior (INSPIRE, OGC, etc.), y da transparencia a los servicios usados por la aplicación, ofrece la posibilidad de estudio por parte de los usuarios, y supone por ende, un avance tanto educativo como tecnológico para la sociedad (Laboratorio de Geomática, 2011). Permitiendo en un futuro poder iniciar certificaciones catastrales desde las notarías de forma más ágil y eficaz, cumpliendo así con la Ley de Economía Sostenible.

Son pues, los sistemas de información que gestionan y administran las propiedades unos sistemas de importante relevancia para los gobiernos. Como consecuencia se han realizado estudios e investigaciones que proponen modelos catastrales que ayuden a manejar mejor los datos.

A nivel europeo se han llevado a cabo otra serie de estudios relacionados con el Catastro, siempre dando forma a un modelo funcional, que pretende ser un estándar común a multitud de países (van Oosterom et al., 2006).

En este sentido, y dada la importancia de los sistemas territoriales y de la propiedad

---

<sup>5</sup><http://www.boe.es/boe/dias/2011/03/05/pdfs/BOE-A-2011-4117.pdf>

<sup>6</sup><http://freepensourcesoftware.org/>

en cualquier país, y ante la necesidad de generar modelos transparentes y compatibles entre los distintos Catastros se han llevado a cabo importantes estudios e iniciativas.

De manera que ya en el 2006 se genera un modelo para la gestión catastral, siempre bajo la directiva INSPIRE, para que sea compatible entre diversos países, aunque se parte de los Países Bajos. En este estudio, ya se tiene en cuenta la variable tiempo, además de las variables persona, parcela y derechos sobre esta (van Oosterom et al., 2006) pero no es hasta el año 2012, en el que se publica la nueva norma ISO para definir un modelo de parcelas catastrales (Land Property Domain Model). Esta variable tiempo, en la mayor parte de las ocasiones sólo se usa para crear una imagen en una fecha determinada del parcelario, son en definitiva instantáneas de una geometría en una fecha determinada (Triglav et al., 2011), variable a la que se le da solución de manera sencilla en esta investigación.

Se han generado estudios que buscan un diseño general y estándar que abarque las necesidades de cualquier sistema catastral, pero aún no permiten un seguimiento de la geometría a través del tiempo. Todo lo que se alcanza a observar es una imagen estática, una singular instantánea de la realidad catastral en la actualidad. Según (Ho et al., 2000), en el año 2000, se hace una aproximación al objeto del que es estudio el actual proyecto de investigación. Se genera un modelo relacionado con bases de datos espacio-temporales, que viene determinado por un dominio espacial de dos dimensiones y un dominio temporal, que es la fecha válida y la fecha de la transacción. La novedad en este estudio es la generación de unas operaciones topológicas temporales que permiten una visualización de las operaciones o alteraciones en las que ha desembocado la actual geometría parcelaria.

En general, están destinados a gestionar la información, con lo cual un sistema relacional de base de datos funciona muy bien, pero ofrece un modelo que no es extensible ni escalable. Y si se pretende dar un servicio público mediante servicios web, la información puede ser demasiado extensa como para dar un servicio eficiente que funcione correctamente. En el presente proyecto, se crea una alternativa a los modelos presentados en estudios previos que sea simple, ágil y eficiente, y sobre todo, que pueda acoplarse a sistemas públicos y privados que trabajen con información catastral.

Tradicionalmente los modelos de bases de datos GIS catastrales no permiten ver los cambios generados por las sucesivas operaciones de actualización de la cartografía Sperry (1999), aunque se han propuesto modelos de bases de datos espaciales que contengan metadatos de estos cambios.

Se busca un diseño que pueda acomodarse al sistema actual de la información catastral, y que sea un diseño que ofrezca la solidez necesaria para poder dar un servicio estable y útil (Çağdaş & Stubkjær, 2011).

Aunque lo que nos ofrece Catastro es una información bastante completa, en algunos casos se requiere un historial de las sucesivas actualizaciones de una parcela que aporte datos relativos a geometría, fecha o tipo de alteración que le ha llevado al estado presente. Un caso que se puede dar (figura 1), es el de una parcela heredada por un ciudadano, que tiene olvidada pues no está en el lugar donde reside.

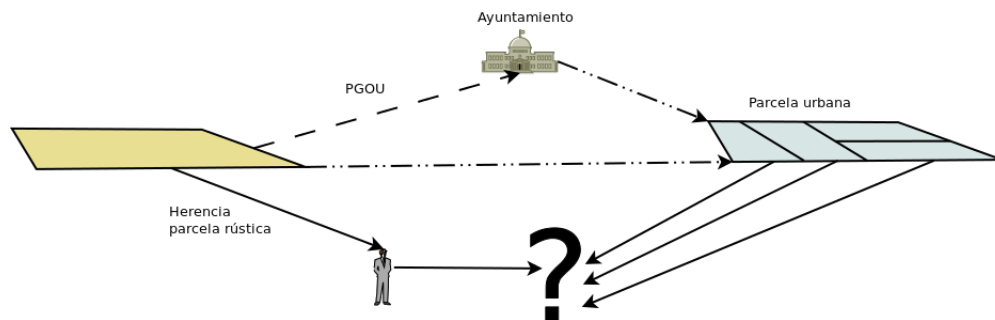


Figura 1: Supuesto de un caso real

Cuando el propietario pretende realizar algún tipo de operación sobre la parcela (venta, alteración catastral, etc), y ésta se ha visto afectada por un plan de ordenación urbana, y su parcela ha cambiado tanto su referencia catastral como su geometría. Aunque en el Ayuntamiento le faciliten la nueva referencia catastral, mediante entrevistas con los colindantes no queda claro que parcela es suya.

## 2. Modelo para la monitorización del parcelario catastral

### 2.1. Selección del sistema gestor de base de datos

La mayoría de los motores de bases de datos usados para la gestión territorial, son sistemas de bases de datos geográficas y relacionales. Pero para la gestión de grandes volúmenes de datos y, sobre todo, si la finalidad es que pueda ser un servicio web, cabe destacar los sistemas de bases de datos NoSQL. Éstos permiten un manejo rápido de la información almacenada, puesto que no debe hacer una comprobación registro a registro, reducen al mínimo los accesos a disco, y pueden almacenar cantidades enormes de datos y a la vez dar servicio a multitud de peticiones. Este tipo de bases de datos se han popularizado gracias sobre todo a las modernas redes sociales, y al fin y al cabo, se pretende un modelo de relaciones entre unas parcelas y otras debido a las continuas actualizaciones. En este sentido, cada parcela tiene una relación con otra a partir de actualizaciones topológicas de su geometría, lo que nos lleva a un red parcelaria con conexiones entre ellas.

El modelo más difundido está comúnmente asociado a las bases de datos relacionales, pero este tipo de sistema aunque podría dar solución al modelo que se quiere plantear, que en vista de estudios anteriores, no ofrece la sencillez que se pretende en esta investigación. Las bases de datos NoSQL, sobre todo ofrece un uso de los recursos más reducido que las bases de datos relacionales, dándose estudios de velocidad entre bases de datos relacionales y de tipo NoSQL (Vicknair, 2010), con resultados de mayor velocidad conforme aumenta el número de registros.

Algunos de los tipos de sistemas gestores de bases de datos que podemos encontrar son: bases de datos documentales, de grafos, clave/valor, multivalor, orientadas a objetos, tabulares, etc. Un ejemplo comparativo de bases de datos NoSQL con licencia free software (table 1):

<b>Datastore</b>	<b>Data Model</b>	<b>Query API</b>
Cassandra	Columnfamily	Thrift
CouchDB	Document	map/reduce views
HBase	Columnfamily	Thrift, REST
MongoDB	Document	Cursor
Neo4j	Graph	Graph
Redis	Collection	Collection
Riak	Document	Nested hashes
Scalaris	Key/value	get/put
Tokyo Cabinet	Key/value	get/put
Voldemort	Key/value	get/put

Cuadro 1: Comparación de almacenes de datos NoSQL y APIs de consulta (Ellis, 2012)

Se barajaron otras opciones como el uso de una base de datos NoSQL de tipo clave / valor, pero lo que mejor se ajusta y que permite una monitorización ya no sólo temporal, sino de relaciones entre unos objetos y otros es la base de datos orientada a grafos.

Por ende, el motor de base de datos por el que se opta es uno orientado a grafos, por dar solución al problema planteado de forma sencilla y sobre todo bastante rápida a un tipo de contenedor de datos que se espera contengan millones de registros.

Este tipo de bases de datos están basadas en la teoría de grafos, que define un grafo (G) como un conjunto (V(G)) de elementos llamados vértices, un conjunto (E(G)) de elementos llamados aristas (edges), y una relación de incidencia, que asocia cada arista con uno o dos vértices en cada uno de sus extremos (Tutte, 2001).

Dentro de la base de datos orientada a grafos los nodos o vértices representan cualquier tipo de objeto definido por el usuario, y los arcos o aristas son las relaciones que hay entre dos o más nodos. Cada nodo representa un objeto con sus propiedades, y las relaciones que hay entre ellos pueden tener también sus propiedades. Estas relaciones son dirigidas o no dirigidas, las primeras tendrán una única dirección o ser bidireccional, y la segunda no tiene una dirección definida. En la figura 2 se puede ver un grafo con relaciones dirigidas.

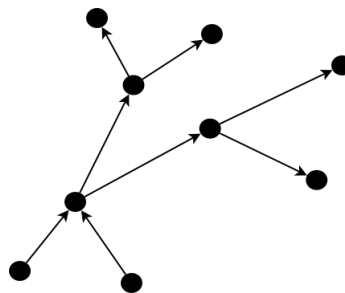


Figura 2: Ejemplo de grafo

## 2.2. Modelo de datos

El modelo de datos trata cada parcela como un nodo dentro del conjunto y se conectan entre ellos por la relación que hay entre pares de parcelas y cualifica las relaciones entre pares de parcelas en función de una alteración catastral que afecta a la parcela de origen y produce una parcela resultante (figura 3):

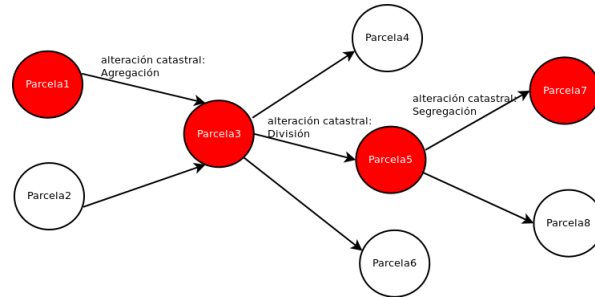


Figura 3: Estructura en grafo del modelo propuesto

De esta manera se puede trazar una línea histórica de los cambios acaecidos entre parcelas, es decir con cada actualización geométrica.

Tanto los nodos como las relaciones tendrán unas propiedades que ofrezcan una información sencilla y útil, para observar esos cambios geométricos.

Los nodos tendrán los siguientes atributos (figura 4):

- Referencia Catastral: es el identificador único que hace referencia a la parcela catastral.
- Geometría: geometría de la parcela catastral.
- Fecha: fecha en la que se añade al conjunto del grafo ese nodo.

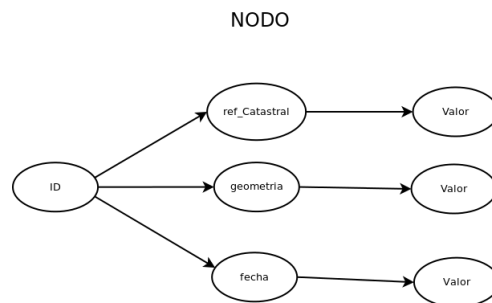


Figura 4: Descripción de un nodo en el modelo de datos

Las relaciones que hay entre los nodos tendrán los siguientes atributos:



- Tipo: tipo de alteración catastral que ha provocado la actualización.
- Fecha: Fecha en la que se produce la actualización.

El modelo es totalmente **escalable** y se pueden añadir más atributos en función de las necesidades.

### 3. Implementación del modelo

#### 3.1. *Neo4j*

Como marco de pruebas se ha elegido el motor de base de datos Neo4j<sup>7</sup>, una base de datos orientada a grafos. Neo4j está desarrollada por Neo Technologies, Inc, y con lenguaje de programación Java, y tiene una licencia GPL v3.0 en su versión community (Technology Neo, 2012).

Una característica de este tipo de bases de datos, es que no existen las consultas ni las transacciones tal cual se hacen en bases de datos relacionales. Para tener un control sobre las acciones que involucran a este tipo de base de datos NoSQL, es necesario desarrollar una API que se encargue de las transacciones de entrada y las consultas.

Por ello, y para poder hacer una implementación del modelo propuesto ha sido necesario desarrollar un software propio que se encargue de hacerlo, una API propia que puede ser usada por terceros y que se aproxime a lo que se pretende hacer con el modelo, que es el poder seguir históricamente una parcela catastral a través de las continuas operaciones de actualización. Se detalla el diseño UML de la API, que ha sido desarrollada totalmente en lenguaje de programación JAVA, cuyo código se distribuirá bajo licencia GPL.

Como se ilustra en la figura 5, el diagrama de la API desarrollada consta de clases internas y públicas.

---

<sup>7</sup><http://www.neo4j.org>

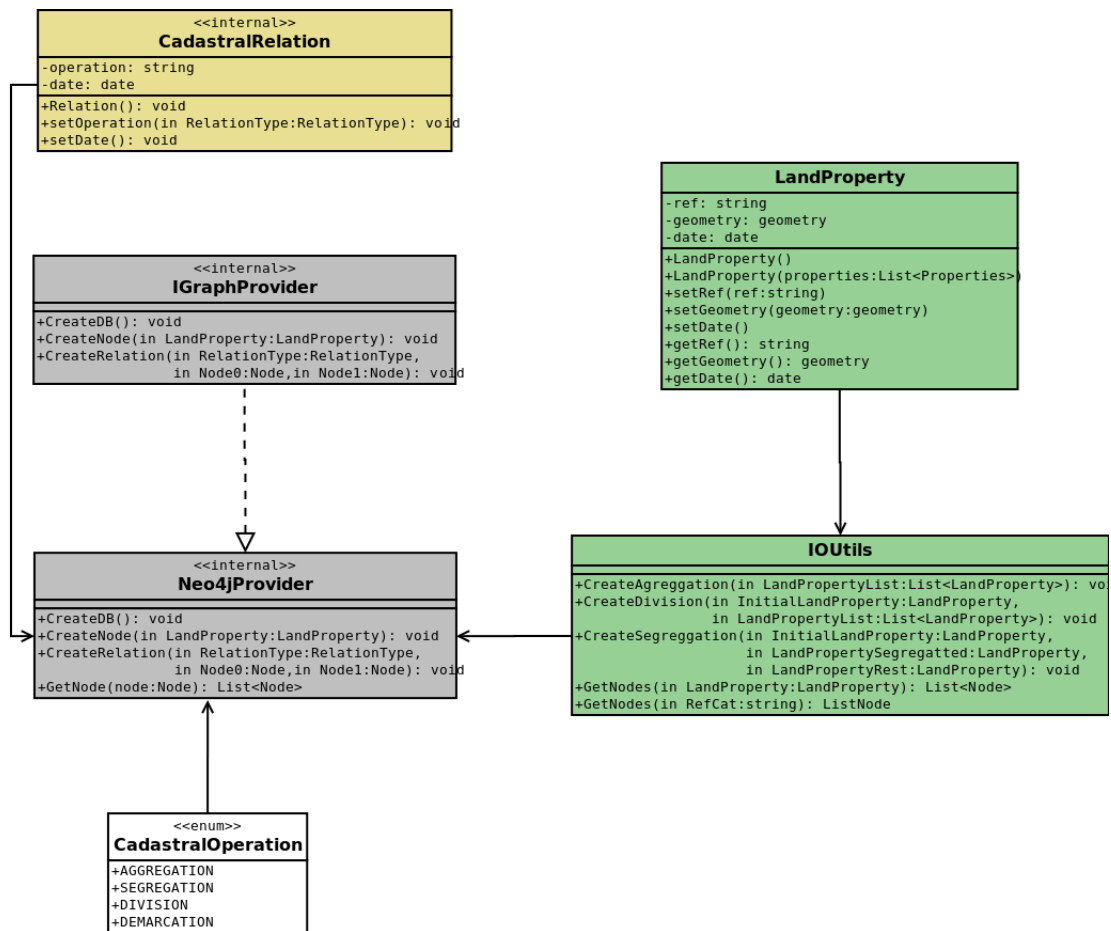


Figura 5: Diagrama uml de la API

Los métodos públicos están contenidos en las clases IOUtils y LandProperty. *IOUtils* se encarga de crear cada una de las operaciones catastrales en nodos y relaciones, creando primero objetos de tipo *LandProperty* que internamente serán convertidos a nodos.

Para el banco de pruebas se omite el uso de la clase CadastralRelation, puesto que no es relevante para la demostración que se realiza, pero que para una implementación real será necesaria para saber la fecha en que se produjo esa alteración catastral.

La API deja la puerta abierta a usarse con otros motores de base de datos orientada a grafos, de manera que sólo es necesario ir creando nuevos proveedores adaptados a la base de datos. En el diagrama UML se puede ver el Neo4jProvider para la base de datos elegida y que hace uso de Neo4j.

### 3.2. Alteración catastral de agregación

A continuación vamos a ver como trabaja el código creado para una operación de agregación. Para hacer una agregación de varias parcelas, IOUtils usa el método AddAggregation,

```
1 public void AddAggregation(  
2     List<LandProperty> landpropertyList, LandProperty  
3     landProperty){  
4     nProvider.CreateAggregation(  
5         landpropertyList, lp);  
6 }  
7
```

que hace una llamada a la clase Neo4jProvider pasando como parámetros una lista con las parcelas que se van a agregar y la parcela resultante de esa operación de agregación.

El método CreateAggregation de Neo4jProvider, se encarga de crear un nodo por cada una de las parcelas entrantes.

```
1 public void CreateSegregation(LandProperty lp,  
2     LandProperty lpSegregated, LandProperty lpRest){  
3     Node node0 = this.toNode(lp);  
4     CadastralOperation c0 = CadastralOperation.  
5         SEGREGATED_T0;  
6     Node node1 = this.toNode(lpSegregated);  
7     Node node2 = this.toNode(lpRest);  
8     this.CreateRelation(c0, node0, node1, node2);  
9 }  
10
```

En la línea 2 del método CreateAggregation, se llama a un método privado que crea los nodos a partir de las parcelas catastrales que se le pasa, es el método toNode:

```
1 private Node toNode(LandProperty lp){  
2     tx = this.neodb.beginTransaction();  
3     Node newNode;  
4     Index<Node> landProperties = index.forNodes("  
5         landProperties");  
6     try{  
7         newNode = this.neodb.createNode();  
8         this._lp = lp;  
9         newNode.setProperty("refCat", _lp.getRefCat());  
10        newNode.setProperty("geometry", _lp.  
11            getGeometry());  
12        newNode.setProperty("date", _lp.getDate());  
13        ;  
14        landProperties.add(newNode, "refCat",  
15            newNode.getProperty("refCat"));  
16        landProperties.add(newNode, "geometry",  
17            newNode.getProperty("geometry"));  
18    }  
19 }  
20
```

```

13         landProperties.add(newNode, "date", newNode.
14             getProperty("date"));
15             tx.success();
16     }
17     finally{
18         tx.finish();
19     }
20     return newNode;
21 }

```

A su vez, dentro de la clase Neo4jProvider se ha creado método privado que se encarga de crear las relaciones entre un nodo y otro. De manera que se le pasa el tipo de operación catastral, el nodo inicial y el nodo final:

```

1 private void CreateRelation(CadastralOperation
2     cOperation, Node n0, Node n1){
3     Relationship rel;
4     RelationshipType relType = cOperation;
5     RelationshipIndex iRelationship = index.
6         forRelationships("iRelationship");
7     tx = neodb.beginTx();
8     try{
9         if(cOperation == CadastralOperation.
10             AGGREGATED_TO){
11             rel = n0.createRelationshipTo(n1,
12                 relType);
13             iRelationship.add(rel, "AGGREGATED_TO",
14                 cOperation);
15             tx.success();
16         }
17         else if(cOperation == CadastralOperation.
18             DIVIDED_TO){
19             rel = n0.createRelationshipTo(n1,
20                 relType);
21             iRelationship.add(rel, "DIVIDED_TO",
22                 cOperation);
23             tx.success();
24         }
25     }
26     finally{
27         tx.finish();
28     }
29 }

```

Los nodos y las relaciones que se han ido añadiendo a la base de datos, son indexadas para posteriormente poder consultar a partir de una de sus propiedades (p. ej.: referencia catastral).

### 3.3. Implementación y prueba del modelo

Para las pruebas realizadas se parte de un escenario ficticio en forma de cuadrícula, cargando por tanto, primero los datos, y dando como resultado un sistema que contiene cincuenta mil nodos, que corresponden con las parcelas, y cuyas relaciones entre unas y otras son las alteraciones catastrales que se han producido a lo largo del tiempo. Desde la consola de Neo4j se hace una consulta a un nodo determinado dada su referencia catastral, consulta que es posible, gracias a la indexación de los nodos a medida que se añaden en la base de datos.

```
neo4j-sh (0)$ START a = node:landProperties(refCat= "1") RETURN a;
```

El resultado en consola de la consulta sería un nodo con esa referencia catastral, junto a los atributos de su referencia catastral, la fecha en que se añadió, y la geometría en el formato estándar WKT<sup>8</sup>, que podrá ser visualizada en programas SIG de escritorio o mediante un servidor de mapas como MapServer<sup>9</sup>.

```
+-----+
| a                                           |
+-----+
| Node[1]{refCat:"1",geometry:"POLYGON((0 500000 0,0 1000000 0,500000 1000000 0,500000 500000 0,0 500000 0))",date:"27/12/12 19:06"} |
+-----+
```

Con el nodo devuelto por la consulta anterior, se sabe qué nodos están conectados al anterior y su direccionalidad, en este caso se trata de una división catastral, y nos devuelve los atributos y valores asociados a los nodos conectados:

```
neo4j-sh (0)$ START a = node(1) MATCH a-[r]->b RETURN r,b;
```

La salida en consola de la consulta anterior serían todos los nodos que tienen relación con el Node[1] devuelto por la consulta anterior:

```
+-----+
| r           | b                                           |
+-----+
| :DIVIDED_TO[3] {} | Node[5]{refCat:"1.3",geometry:"POLYGON((250000 750000 0,250000 1000000 0,500000 1000000 0,500000 750000 0,250000 750000 0))",date:"..."} |
| :DIVIDED_TO[2] {} | Node[4]{refCat:"1.2",geometry:"POLYGON((250000 500000 0, 250000 750000 0, 500000 750000 0, 500000 500000 0,250000 500000 0))",date:"..."} |
| :DIVIDED_TO[1] {} | Node[3]{refCat:"1.1",geometry:"POLYGON((0 750000 0,0 1000000 0,250000 1000000 0,250000 750000 0,0 750000 0))",date:"..."} |
| :DIVIDED_TO[0] {} | Node[2]{refCat:"1.0",geometry:"POLYGON((0 500000 0,0 750000 0,250000 750000 0,250000 500000 0,0 500000 0))",date:"..."} |
+-----+
```

En la figura 6 vemos como quedarían gráficamente el resultado de dos operaciones catastrales si se da el caso en que una parcela viene precedida de una operación de división, y luego se agrega junto a diversas parcelas más.

---

<sup>8</sup>Well-know Text

<sup>9</sup><http://mapserver.org/>

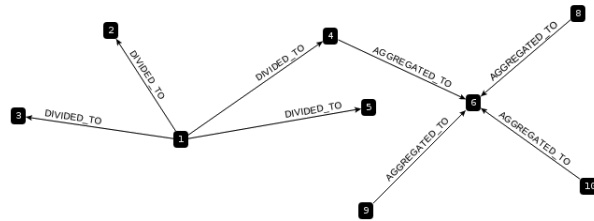


Figura 6: Resultado de sucesivas operaciones de actualización

### 3.4. Aplicabilidad

Mediante servicios REST<sup>10</sup> se realiza una doble acción, por un lado desde organismos oficiales, se añadirán parcelas al conjunto del grafo, y por otro lado, permitir consultar esta información y que se muestre de forma inteligible al usuario.

En definitiva, el modelo presentado está pensado para poder ser explotado mediante servicios web (REST, WPS...) y ofrezca un acceso sencillo al usuario.

Volviendo al caso real citado al principio del artículo, bastaría con realizar una petición web usando como parámetro de consulta la referencia catastral nueva o la antigua de la que dispone el propietario, para mediante servicios web, procesarse la respuesta, tal como se muestra en la figura 7.

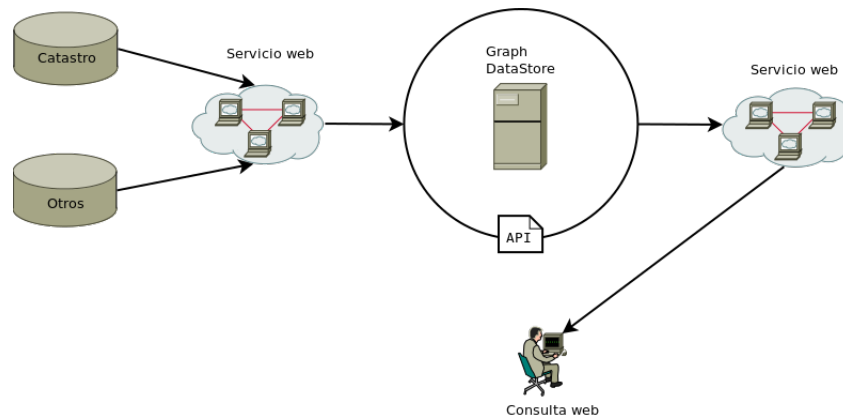


Figura 7: Uso del modelo de datos

Como respuesta se obtendrán 'n' parcelas que se han visto involucradas en algún tipo de alteración catastral dada la referencia catastral que le hemos pasado, y se podría seguir un hilo histórico de su parcela desde que la heredó hasta el presente.

<sup>10</sup>Representational State Transfer

#### 4. Conclusiones

El ejemplo desarrollado en este trabajo pone de manifiesto que el modelo de datos se hace sencillo de manejar y de explotar una vez creada una API que maneje el motor de base de datos. La implementación llevada a cabo sobre Neo4j permite ver que parcelas han tenido alguna relación con otra dada su referencia catastral. Permite recuperar estados de las parcelas anteriores a una alteración catastral llevada a cabo sobre una parcela concreta. Parece rápido, puesto que, para ejecutar una consulta no debe ir comprobando registro a registro. Se antoja un sistema especialmente útil para pruebas periciales o ámbitos legales, donde se necesita ver de donde proviene una parcela con una referencia catastral determinada, caso por ejemplo de un plan parcial.

El modelo desarrollado permitirá que más adelante se pueda explotar mediante servicios web públicos, para mostrar una línea histórica de las operaciones de actualización de una parcela determinada, aunque indudablemente se ha de desarrollar una API más adecuada para que los métodos públicos sean más útiles para terceros, ya que las pruebas realizadas han sido sobre una aplicación de escritorio.

Una de la aplicabilidad más inmediata es junto con la aplicación desarrollada Ramón Llull (Llorens Cobos et al., 2005), desde la cual se podrían hacer consultas sobre parcelas que se han visto afectadas por algún tipo de alteración catastral.

Este modelo sigue dispuesto a modificarse, sobre todo, con la aprobación de la norma ISO 19152:2012, deberá adecuarse a este estándar para parcelas catastrales (International Organization for Standardization, 2012).

Por tanto, y como trabajos futuros, se crearan nuevos métodos de consulta y, se mejorará y adecuará el código para nuevos casos de estudio, que puedan ser usados por servicios web.

#### Referencias

- Alonso Peña, C. (2006). Nuevos escenarios de colaboración en el mantenimiento de la información catastral. *CT: Catastro*, 56, 41–56.
- Alonso Peña, C., Fernández Gómez, R., & Yáñez Morante, E. (2009). Consulta y actualización de información catastral mediante servicios web. *CT: Catastro*, (pp. 37–46).
- Cano, A., Virgós, L., Olivares, J. M., Quintana, F. J., & Cepeda, F. G. (2006). Servicio WFS de la D.G. del Catastro. Infraestructura de Datos Espaciales. *CT/Catastro*, 56.
- Dirección General de Catastro, D. (2012). Portal de la Dirección General de Catastro.
- Çağdaş, V., & Stubkjær, E. (2011). Design research for cadastral systems. *Computers, Environment and Urban Systems*, 35, 77–87.
- Consejo Superior Geográfico, C. (2012). Geoportal de Infraestructura de Datos Espaciales.
- Ellis, J. (2012). NoSQL Ecosystem.

- Ho, D., Ho, K., & Soo, H. (2000). A spatiotemporal database model and query language. *Journal of Systems and Software*, 55, 129–149.
- INSPIRE (2012). Portal de INSPIRE.
- Laboratorio de Geomática, U. d. A. (2011). Proyecto Ramón Llull.
- Llorens Cobos, F., Manuel, J., Martínez, M., Tomás, J., & Carrión, N. (2005). Proyecto Ramon Llull : sistema de gestión de alteraciones catastrales para las notarías de la Comunidad Valenciana. In *I Jornadas de SIG Libre 1* (pp. 1–12).
- van Oosterom, P., Lemmen, C., Ingvarsson, T., van der Molen, P., Ploeger, H., Quak, W., Stoter, J., & Zevenbergen, J. (2006). The core cadastral domain model. *Computers, Environment and Urban Systems*, 30, 627–660.
- Open Geospatial Consortium (2010). *Open Geospatial Consortium Geography Markup Language ( GML ) simple features profile*. Technical Report Open Geospatial Consortium.
- Open Geospatial Consortium (2012). *Geography Markup Language ( GML ) — Extended schemas and encoding rules*. Technical Report.
- Sperry, C. C. . L. T., L. (1999). A lineage metadata model for the temporal management of a cadastre application. In *Database and Expert Systems Applications, 1999. Proceedings. Tenth International Workshop on*. (pp. 466–474).
- International Organization for Standardization, I. (2012). ISO.
- Suárez Pinilla, J. I. (2012). Notarios y registradores.
- Technology Neo, I. (2012). Neo4j.
- Triglav, J., Petrovič, D., & Stopar, B. (2011). Spatio-temporal evaluation matrices for geospatial data. *International Journal of Applied Earth Observation and Geoinformation*, 13, 100–109.
- Tutte, W. (2001). *Graph Teory*.
- Vicknair, M. M. Z. Z. . N. X., C. (2010). A comparison of a graph database and a relational database: a data provenance perspective. In *ACM SE '10 Proceedings of the 48th Annual Southeast Regional Conference* (pp. 466–474).