



## Consideraciones sobre la remisión de información en XML (Revisión 18-05-09)

1.	Procedimiento de generación de un archivo XML .....	1
2.	Herramientas para el tratamiento de documentos XML.....	2
3.	Errores más comunes en la generación de archivos XML .....	3
4.	Errores en el envío de archivos XML a la D.G. Catastro .....	9
5.	Herramientas de conversión .....	10
6.	Errores más frecuentes en la generación del texto plano.....	13
7.	Hoja de estilo (XSL).....	15

### 1. Procedimiento de generación de un archivo XML

A la hora de generar un documento XML para su posterior envío a la Dirección General del Catastro, se recomienda seguir los siguientes pasos para evitar la aparición de errores en el mismo.

- Consulta de documentos

Para que el fichero XML que se genere sea lo más correcto posible se deben consultar los siguientes documentos.

- Esquemas:  
Se deberían consultar tanto el esquema de XCAT (Idc.xsd) como el correspondiente al formato del archivo en cuestión. Ambos están disponibles en la web del catastro.  
Esto permitirá averiguar qué campos son obligatorios y cuáles son optativos, el formato de todos los elementos, las restricciones, etc.
- Instancia explicativa:  
En la web del Catastro hay disponible una instancia explicativa para cada formato de intercambio. En cada instancia se explica el contenido de cada campo y se proporciona información acerca del formato.
- Ejemplo instancia:  
También está disponible para cada formato una instancia de ejemplo, que puede servir de ayuda en el momento de generar el archivo.
- Especificaciones técnicas:  
El documento de especificaciones técnicas ayudará a comprender conceptos de catastro relacionados con el formato que se está utilizando.

- Verificar que está bien formado

Una vez que se ha generado el documento XML, se debe verificar que éste está bien formado, es decir, que sigue correctamente las especificaciones XML y su estructura es correcta.

En ocasiones no será necesario realizar esta comprobación, puesto que al realizar el paso siguiente (validación), las herramientas verifican también que el documento esté bien formado.

- Validar contra el esquema

Es necesario validar el documento XML para comprobar que se ajusta a su esquema correspondiente.

La validación es un paso imprescindible antes del envío del documento a la Dirección General del Catastro, puesto que no se admitirán ficheros no validados contra el correspondiente esquema.



## 2. Herramientas para el tratamiento de documentos XML

A continuación se proporciona un listado de herramientas que se pueden utilizar para facilitar la visualización, edición y validación de documentos XML.

- Validación online
  - <http://www.w3.org/2001/03/webdata/xsv>

- Editores XML

Permiten realizar validación contra el esquema.

- SW libre:
  - firstobject XML Editor - [http://www.firstobject.com/dn\\_editor.htm](http://www.firstobject.com/dn_editor.htm)  
Permite tratar archivos grandes (300 MB aproximadamente).
  - XML Copy Editor - <http://sourceforge.net/projects/xml-copy-editor/>  
Permite tratar archivos grandes (150 a 200 MB aproximadamente).
- SW propietario:
  - Altova XMLSpy - <http://www.altova.com>
- Librerías
  - MSXML (Microsoft) – <http://www.microsoft.com>
  - Xerces (Apache) - <http://xerces.apache.org>

Código Java para validar documentos XML utilizando Xerces:

```
//configuración del parser  
  
SAXParserFactory factory = SAXParserFactory.newInstance();  
SAXParser parser = factory.newSAXParser();  
XMLReader reader=parser.getXMLReader();  
reader.setFeature("http://xml.org/sax/features/namespace", true);  
reader.setFeature("http://apache.org/xml/features/validation/schema", true);  
reader.setFeature("http://xml.org/sax/features/validation", true);  
...  
  
//validación  
  
reader.parse(archivo);
```



### 3. Errores más comunes en la generación de archivos XML

A continuación se describen los errores más frecuentes que pueden darse en la generación de archivos XML.

#### 1. Fechas

Toda fecha deberá ser válida y seguir el formato AAAA-MM-DD

AAAA: año (4 dígitos)

MM: mes (2 dígitos)

DD: día (2 dígitos)

Ejemplo:

Correcto	Incorrecto
2007-01-04	04/01/2007

#### 2. Horas

Toda hora deberá ser válida y cumplir con el formato HH:MM:SS

HH: hora (2 dígitos)

MM: minutos (2 dígitos)

SS: segundos (2 dígitos)

Ejemplo:

Correcto	Incorrecto
14:24:18	14:24 14h24m18s

#### 3. Longitud de los campos

La longitud de los campos debe ser la que se indique en el esquema correspondiente, si ésta viene especificada de forma explícita o debe pertenecer al rango indicado, en caso de que se indique un valor máximo y/o mínimo.

Ejemplo:

```
<xsd:length value="2"/>
```

La longitud del campo deberá ser 2 obligatoriamente.

```
<xsd:maxLength value="25"/>
```

```
<xsd:minLength value="10"/>
```

La longitud deberá estar comprendida entre 10 y 25.

#### 4. Rango de valores

Cuando se indique un rango de valores, el valor del campo deberá estar comprendido entre el valor mínimo y máximo especificados.

Ejemplo:



```
<!--Descripción de la referencia catastral-->  
<xsd:simpleType name="erc">  
  <xsd:annotation>  
    <xsd:documentation>Descripción de la referencia  
    catastral</xsd:documentation>  
  </xsd:annotation>  
  <xsd:restriction base="xsd:integer">  
    <xsd:minInclusive value="1"/>  
    <xsd:maxInclusive value="3"/>  
  </xsd:restriction>  
</xsd:simpleType>
```

Correcto	Incorrecto
<erc>1</erc>	<erc>0</erc>
<erc>2</erc>	<erc>25</erc>

## 5. Decimales

En los números decimales la parte entera y la decimal deben estar separadas por una coma (,). La parte decimal debe tener la longitud indicada, no puede acortarse o suprimirse por el hecho de que sólo haya ceros a la derecha.

Ejemplo:

Correcto	Incorrecto
20,30	20.30
20,42	20,3

## 6. Mayúsculas

El contenido de casi todos los campos alfanuméricos debe ir obligatoriamente en mayúsculas. Como excepción se puede citar al tipo cspr (código de subparcela), que puede ir en minúsculas.

## 7. Enumeraciones

El contenido de un campo de tipo enumeración debe ser obligatoriamente una de las opciones indicadas en el esquema.

Ejemplo:

```
<!--Clase de naturaleza del bien inmueble-->  
<xsd:simpleType name="cn">  
  <xsd:annotation>  
    <xsd:documentation>Código de naturaleza del bien inmueble UR: Urbana,  
    Rústica, BI: BICES,</xsd:documentation>  
  </xsd:annotation>  
  <xsd:restriction base="xsd:string">  
    <xsd:enumeration value="UR"/>  
    <xsd:enumeration value="RU"/>  
    <xsd:enumeration value="BI"/>  
  </xsd:restriction>  
</xsd:simpleType>
```



Correcto	Incorrecto
<code>&lt;cn&gt;UR&lt;/cn&gt;</code>	<code>&lt;cn&gt;NB&lt;/c&gt;</code>

### 8. Nombres de las etiquetas

Los nombres de las etiquetas se deben corresponder exactamente con los indicados en el esquema correspondiente.

### 9. Orden de los elementos

Se debe respetar exactamente el orden de los elementos tal y como se especifican en el esquema correspondiente.

Ejemplo:

```
<xsd:element name="notreg">  
  <xsd:annotation>  
    <xsd:documentation>Formato de Notarios y Registradores </xsd:documentation>  
  </xsd:annotation>  
  <xsd:complexType>  
    <xsd:sequence>  
      <xsd:element name="cab" type="cabnr">  
        <xsd:annotation>  
          <xsd:documentation>Cabecera</xsd:documentation>  
        </xsd:annotation>  
      </xsd:element>  
      <xsd:element name="luden" type="lunr">  
        <xsd:annotation>  
          <xsd:documentation>Lista de Unidades de datos de  
          entrada</xsd:documentation>  
        </xsd:annotation>  
      </xsd:element>  
    </xsd:sequence>  
  </xsd:complexType>  
</xsd:element>
```

Correcto	Incorrecto
<pre>&lt;notreg ...&gt;   &lt;cab&gt;     ...   &lt;/cab&gt;   &lt;luden&gt;     ...   &lt;/luden&gt; &lt;/notreg&gt;</pre>	<pre>&lt;notreg ...&gt;   &lt;luden&gt;     ...   &lt;/luden&gt;   &lt;cab&gt;     ...   &lt;/cab&gt; &lt;/notreg&gt;</pre>



## 10. Etiquetas abiertas

Es un error dejar etiquetas abiertas. Cada campo debe tener su etiqueta de inicio y de cierre.

Ejemplo:

Correcto	Incorrecto
<pre>&lt;cpp&gt;   &lt;cpo&gt;     ...   &lt;/cpo&gt; &lt;/cpp&gt;</pre>	<pre>&lt;cpp&gt;   &lt;cpo&gt;     ... &lt;/cpp&gt;</pre>

## 11. Contenido de los campos

Es un error dejar campos vacíos. Entre las etiquetas de inicio y de cierre el contenido debe ser del tipo esperado.

Ejemplo:

Correcto	Incorrecto
<pre>&lt;lorus&gt;   &lt;cpp&gt;     &lt;cpo&gt;       ...     &lt;/cpo&gt;     &lt;cpa&gt;       ...     &lt;/cpa&gt;   &lt;/cpp&gt;   &lt;npa&gt;     ...   &lt;/npa&gt; &lt;/lorus&gt;</pre>	<pre>&lt;lorus&gt;   &lt;cpp&gt;     &lt;cpo/&gt;     &lt;cpa/&gt;   &lt;/cpp&gt;   &lt;npa/&gt; &lt;/lorus&gt;</pre>
	<pre>&lt;cpo&gt; &lt;/cpo&gt;</pre>

## 12. Campos obligatorios

Es un error no incluir en el archivo los campos obligatorios. Campos obligatorios son todos aquellos cuya definición no incluye `minOccurs="0"`.

Ejemplo:

```
<xsd:element name="cutit" type="cutit" minOccurs="0">
  <xsd:annotation>
    <xsd:documentation>Cuenta de titulares</xsd:documentation>
  </xsd:annotation>
</xsd:element>
```

El elemento cutit no es obligatorio.



### 13. Ocurrencia máxima

La ocurrencia máxima de un campo puede ser mayor que 1, es decir, hay campos que pueden repetirse un número determinado de veces, incluso infinitas veces.

Ejemplo:

```
<xsd:complexType name="liddf">
  <xsd:sequence>
    <xsd:element name="iddf" type="iddf" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

La ocurrencia máxima de iddf es infinita, puede repetirse un número ilimitado de veces.

### 14. Opciones exclusivas

Cada vez que se ofrezcan diversas alternativas (choice), hay que decantarse por una de las opciones ofrecidas. No se puede poner más de una de ellas.

Ejemplo:

```
<xsd:complexType name="rcof">
  <xsd:annotation>
    <xsd:documentation>Referencia catastral o de finca</xsd:documentation>
  </xsd:annotation>
  <xsd:choice>
    <xsd:element name="rc" type="rc">
      <xsd:annotation>
        <xsd:documentation>Referencia Catastral</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
    <xsd:element name="rfin" type="pc">
      <xsd:annotation>
        <xsd:documentation>Referencia de finca</xsd:documentation>
      </xsd:annotation>
    </xsd:element>
  </xsd:choice>
</xsd:complexType>
```

Correcto	Incorrecto
<pre>&lt;rcof&gt;   &lt;rfin&gt;     ...   &lt;/rfin&gt; &lt;/rcof/&gt;</pre>	<pre>&lt;rcof&gt;   &lt;rc&gt;     ...   &lt;/rc&gt;   &lt;rfin&gt;     ...</pre>
<pre>&lt;rcof&gt;   &lt;rc&gt;     ...   &lt;/rc&gt; &lt;/rcof/&gt;</pre>	<pre>&lt;rfin&gt;   ... &lt;/rfin&gt; &lt;/rcof/&gt;</pre>



## 15. Otras validaciones

Aparte de las ya mencionadas, algunos campos realizan una validación más compleja, que será necesario consultar en el esquema correspondiente para evitar errores.

Ejemplo:

```
<xsd:simpleType name="nif">  
  <xsd:annotation>  
    <xsd:documentation>Nif</xsd:documentation>  
  </xsd:annotation>  
  <xsd:restriction base="xsd:string">  
    <xsd:length value="9"/>  
    <xsd:pattern value="([A-Z][0-9]){9}"/>  
  </xsd:restriction>  
</xsd:simpleType>
```

## 16. Referencia al esquema

Cada archivo XML debe contener una referencia al esquema correspondiente, mediante el atributo `xsi:schemaLocation`, al que se darán 2 valores: un espacio de nombres (namespace) y la URL del esquema.

Ejemplo:

```
<notreg xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance  
xsi:schemaLocation="http://www.catastro.meh.es/ http://www.catastro.meh.es/ws/esquemas/notreg.xsd"  
xmlns="http://www.catastro.meh.es/">
```

```
<finentrada xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance  
xsi:schemaLocation="http://www.catastro.meh.es/  
http://www.catastro.meh.es/ws/esquemas/finentrada.xsd" xmlns="http://www.catastro.meh.es/">
```

## 17. Campos en blanco

El contenido de ningún campo podrá consistir únicamente en espacios en blanco. En el caso de que se desconozca algún dato obligatorio de tipo alfanumérico, se deben introducir los caracteres NC para indicar esta circunstancia. Si el dato obligatorio que se desconoce corresponde a un año, se debe introducir el valor 0001.

## 18. Codificación

La codificación o 'encoding' de los ficheros XML debe ser UTF-8. Se deben cumplir dos condiciones:

- La primera línea debe contener la siguiente etiqueta, que informa acerca de la codificación utilizada:

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

- El archivo, al guardarse, debe estar codificado con el juego de caracteres UTF-8. La mayoría de editores XML y de texto permiten elegir esta codificación al guardar un archivo. También existen programas, como 'recode' (<http://recode.progiciels-bpi.ca>), que pueden cambiar la codificación de los archivos. Otras herramientas, como 'UTF-8 Validator' (<http://bolek.techno.cz/UTF8-Validator>), permiten comprobar si un fichero está codificado con UTF-8.



#### 4. Errores en el envío de archivos XML a la D.G. Catastro

Cada vez que se envíe un fichero a Catastro, se comprobará que es válido. Si no lo es, se devolverá al usuario una lista con los errores que contiene dicho fichero.

Dicha lista de errores tendrá el siguiente aspecto:

```
The value of the 'http://www.catastro.meh.es:/tft' element does not equal its fixed value.
The element 'http://www.catastro.meh.es:/cab' has incomplete content. Expected 'http://www.catastro.meh.es:/cuexp'.
The 'http://www.catastro.meh.es:/texp' element has an invalid value according to its data type.
The element 'http://www.catastro.meh.es:/expg' has incomplete content. Expected 'http://www.catastro.meh.es:/aexpg'.
The element 'http://www.catastro.meh.es:/nbi' has incomplete content. Expected 'http://www.catastro.meh.es:/nbn'.
The element 'http://www.catastro.meh.es:/dt' has invalid child element 'http://www.catastro.meh.es:/nm'. Expected 'http://www.catastro.meh.es:/np'.
The 'http://www.catastro.meh.es:/texp' element has an invalid value according to its data type.
The element 'http://www.catastro.meh.es:/expg' has incomplete content. Expected 'http://www.catastro.meh.es:/aexpg'.
The element 'http://www.catastro.meh.es:/nbi' has incomplete content. Expected 'http://www.catastro.meh.es:/nbn'.
The element 'http://www.catastro.meh.es:/pos' has incomplete content. Expected 'http://www.catastro.meh.es:/dp'.
The element 'http://www.catastro.meh.es:/dt' has invalid child element 'http://www.catastro.meh.es:/nm'. Expected 'http://www.catastro.meh.es:/np'.
The element 'http://www.catastro.meh.es:/pos' has incomplete content. Expected 'http://www.catastro.meh.es:/dp'.
The element 'http://www.catastro.meh.es:/pos' has incomplete content. Expected 'http://www.catastro.meh.es:/dp'.
The element 'http://www.catastro.meh.es:/texp' element has an invalid value according to its data type.
The element 'http://www.catastro.meh.es:/expg' has incomplete content. Expected 'http://www.catastro.meh.es:/aexpg'.
The element 'http://www.catastro.meh.es:/nbi' has incomplete content. Expected 'http://www.catastro.meh.es:/nbn'.
The element 'http://www.catastro.meh.es:/dt' has invalid child element 'http://www.catastro.meh.es:/nm'. Expected 'http://www.catastro.meh.es:/np'.
The element 'http://www.catastro.meh.es:/dir' has invalid child element 'http://www.catastro.meh.es:/pnp'. Expected 'http://www.catastro.meh.es:/tv'.
The 'http://www.catastro.meh.es:/texp' element has an invalid value according to its data type.
The element 'http://www.catastro.meh.es:/expg' has incomplete content. Expected 'http://www.catastro.meh.es:/aexpg'.
The element 'http://www.catastro.meh.es:/nbi' has incomplete content. Expected 'http://www.catastro.meh.es:/nbn'.
The element 'http://www.catastro.meh.es:/dt' has invalid child element 'http://www.catastro.meh.es:/nm'. Expected 'http://www.catastro.meh.es:/np'.
The element 'http://www.catastro.meh.es:/dir' has invalid child element 'http://www.catastro.meh.es:/pnp'. Expected 'http://www.catastro.meh.es:/tv'.
```

Cada error de la lista contendrá la siguiente información:

- El elemento en el que ha producido el error, identificado por su namespace (en este caso, 'http://www.catastro.meh.es/') y por el nombre de la etiqueta.
- La explicación del error encontrado, en inglés.





3. Al finalizar la ejecución:

- a. Si la ejecución finaliza correctamente, lo indica mediante el mensaje 'Finalizado OK'.

El XML generado estará en el fichero que hayamos indicado como destino, y será necesario validarlo contra el esquema correspondiente para comprobar que no haya errores.

- b. Si hay algún error en la generación, se indicará por pantalla, y se cancelará la ejecución. Esto ocurre cuando el fichero plano tiene algún error grave.

La información que se puede obtener del error es la siguiente:

- Posición en el fichero plano del registro de "Expediente" al que pertenece el elemento que ha producido el error.
- Identificación de dicho expediente:
  - o Referencia del expediente origen de la alteración.
  - o Entidad origen de la alteración.
  - o Tipo de intercambio.
  - o Tipo de expediente.
- Tipo de registro que contiene el error: Normalmente, el error se ha producido exactamente en el tipo de registro indicado, aunque en ocasiones se indica de manera aproximada, como ocurre en el caso de registros de titularidad.
- Descripción del error en inglés.

• Requisitos técnicos

No se necesita ninguna adaptación especial para poder ejecutar la herramienta de transformación, puede ejecutarse en cualquier ordenador, si bien, se ha localizado la siguiente incidencia:

Las herramientas de transformación necesitan la librería de Microsoft **MSXML 4.0** para poder traducir a XML el fichero plano de entrada, pero esta librería contiene un fallo en su código que provoca una excepción `_com_error`, que aborta dicha conversión no siendo posible la obtención del fichero de salida en XML.

Para solucionar este problema, sólo si se ha detectado este error en alguna ejecución de la aplicación, se debe actualizar **MSXML 4.0** con **MSXML 4.0 sp2**, que se puede descargar gratuitamente desde la Web de Microsoft.

Se puede acceder directamente desde el siguiente enlace:

<http://www.microsoft.com/downloads/details.aspx?displaylang=es&FamilyID=3144B72B-B4F2-46DA-B4B6-C5D7485F2B42>

Esta solución es compatible para los sistemas operativos: Windows 2000, Windows 98, Windows ME, Windows NT, Windows Server 2003, Windows XP.

**NOTA:** Las pruebas para esta solución y las pruebas de las herramientas de transformación han sido realizadas con éxito para un PC con Windows XP SP2.

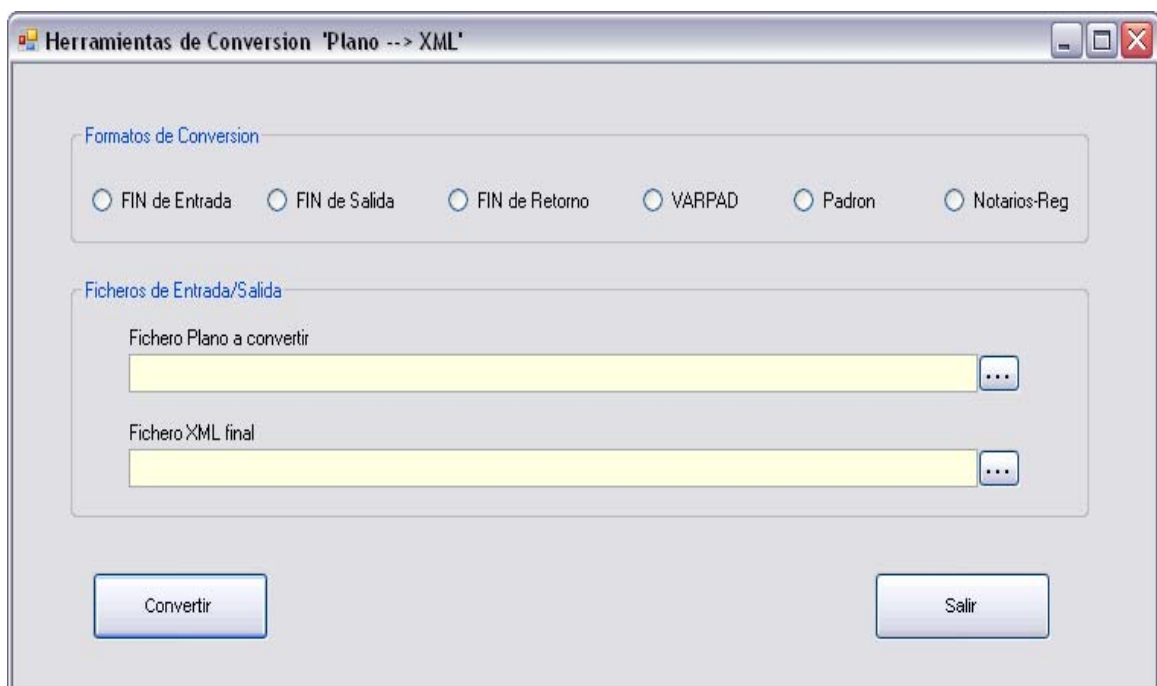


- Aplicación gráfica

Se ha desarrollado una aplicación gráfica que facilita el uso de todas las herramientas de conversión existentes. Se encuentra disponible en la página web del Catastro.

Esta aplicación permite realizar la conversión de un fichero plano a XML en cualquiera de los formatos de intercambio de Catastro disponibles.

Consta de una interfaz gráfica muy sencilla de utilizar que evita los inconvenientes de tener que trabajar mediante la línea de comandos.





## 6. Errores más frecuentes en la generación del texto plano.

Se describen a continuación un conjunto de errores frecuentes en los que se suele incurrir al generar el fichero plano. Algunos de estos errores hacen imposible la generación del XML mediante la herramienta de conversión, otros la permiten, pero generan documentos inválidos.

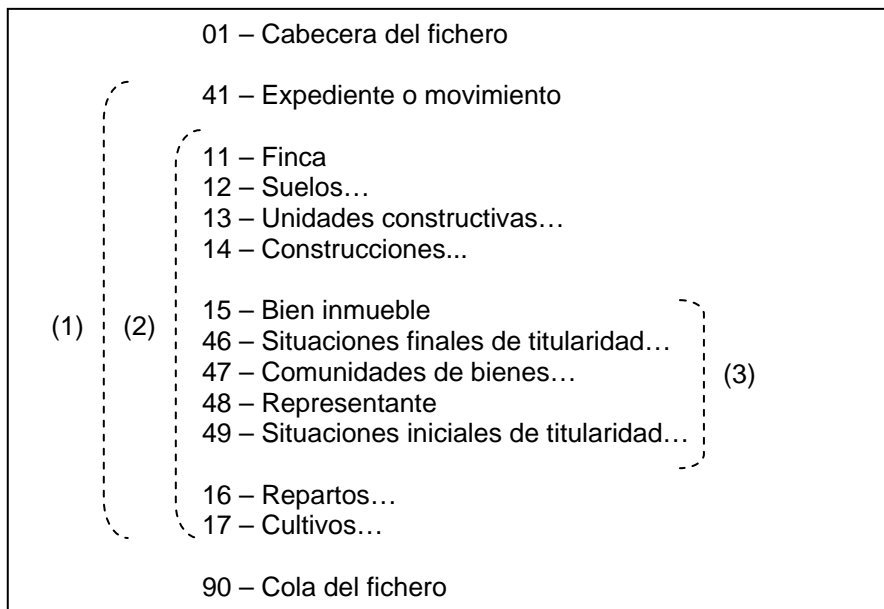
### 1. Formato del fichero

El formato del fichero plano tiene que coincidir con una de las siguientes opciones:

- Todo el contenido en una única línea, y cada registro con un tamaño de 1000 caracteres.
- Cada registro en una línea de 1000 caracteres como mínimo, con el carácter de fin de fichero en una línea aparte.

### 2. Orden de los registros

El orden de los registros debe ser el siguiente:



Nota: No todos los registros aparecen en todos los formatos, ni todos los que aparecen son obligatorios.

- (1) Se repite por cada expediente del fichero
- (2) Se repite por cada finca
- (3) Se repite por cada bien inmueble

### 3. Posiciones

Se deben rellenar los campos en las posiciones adecuadas. El descuadre de posiciones provoca un error grave.

### 4. Fechas



Las fechas son especialmente importantes. Para una correcta transformación del fichero plano deben estar correctamente rellenas y en las posiciones adecuadas.

La existencia de una fecha errónea provocaría un error de formato o de fuera de rango en la generación del XML.

Ejemplo:

Correcto	Incorrecto
20070101	2006464
Si la fecha es opcional: 00000000	Si la fecha es obligatoria: 00000000

## 5. Campos numéricos

Los campos numéricos deben estar rellenos con números, nunca con espacios en blanco.

Un campo numérico cuyo valor tiene una longitud menor que la indicada debe rellenarse con ceros a la izquierda, puesto que si se ponen a la derecha su significado será numérico.

Un campo numérico opcional, cuando no exista, podrá rellenarse, o completamente con ceros, o con espacios.

Es posible que un único campo del formato en texto plano se corresponda con varias etiquetas en el XML (como la parcela catastral y el número de finca registral). Si esto sucediese hay que tener en cuenta que si el campo origen es numérico todas sus posiciones se han de rellenar con números, aunque parte del campo no exista.

Ejemplo:

Un campo numérico de 19 posiciones.

Correcto	Incorrecto
0000012345678911234	12345678911234

## 6. Campos alfanuméricos

Los campos alfanuméricos deben estar rellenos con espacios en blanco a su derecha.

## 7. Enumeraciones

Los campos que sólo se pueden rellenar con una determinada enumeración de valores han de estar rellenos sólo con uno de esos valores.

Ejemplo:

La clase de naturaleza sólo acepta los valores UR, RU o BI. Cualquier otro valor es incorrecto.

Algunos de los errores que se acaban de describir imposibilitan la generación del XML, no obstante, el hecho de generar el documento XML no implica que éste sea válido. Por tanto, después de la generación, se ha de validar el fichero resultante a fin de comprobar la inexistencia de otros errores.



## 7. Hoja de estilo (XSL)

Algunos de los formatos de intercambio de Catastro disponen de una hoja de estilo que permite visualizar de manera inmediata un resumen de la información presente en un fichero en formato XML.

Esta hoja de estilo puede utilizarse de dos maneras: o bien utilizando el navegador para abrir el archivo XML, o utilizando una herramienta que genere un documento HTML a partir del archivo XML correspondiente.

- Navegador

Para que el navegador pueda aplicar la hoja de estilo al archivo XML es necesario que este último tenga incluida una referencia a la hoja de estilo correspondiente.

En algunos casos, el fichero XML ya se creará con la referencia a la hoja de estilo, como es el caso de los formatos DOC, Padrón BICE y Errores.

En otros, sin embargo, el fichero no tendrá incluida dicha referencia por defecto, aunque se podrá modificar manualmente para incluirla si se desea visualizar el fichero de esta manera. Este sería el caso de FIN de salida, FIN de retorno y Padrón de urbana o rústica.

Para referenciar una hoja de estilo en un fichero XML, después de la primera línea (`<?xml version="1.0" encoding="UTF-8"?>`) se debe incluir una etiqueta como esta:

```
<?xml:stylesheet href="hoja_estilo" type="text/xsl"?>
```

donde *hoja\_estilo* sería la ruta completa del fichero de la hoja de estilo correspondiente

Por ejemplo, en el caso del formato padrón sería así:

```
<?xml version="1.0" encoding="UTF-8" ?>
<?xml:stylesheet href=" http://www.catastro.meh.es/ws/esquemas/padron.xsl" type="text/xsl"?>
<padron xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.catastro.meh.es/
http://www.catastro.meh.es/ws/esquemas/padron.xsd"
xmlns="http://www.catastro.meh.es/">
```

Cuando el fichero es excesivamente grande no es recomendable utilizar el navegador para aplicar la hoja de estilo, ya que puede consumir demasiados recursos de la máquina.

- Procesador XSL

Los procesadores XSL son capaces de generar documentos en diferentes formatos a partir de un archivo XML y una hoja de estilo (XSL).

Para que una herramienta de este tipo genere el correspondiente fichero en formato HTML, solamente será necesario indicar el nombre del archivo XML y el nombre de la hoja de estilo, así como el nombre que deberá tener el nuevo documento HTML.

Existen muchas de estas herramientas disponibles en Internet. Una de ellas es 'Instant saxon' (<http://saxon.sourceforge.net/saxon6.5.2/instant.html>).



Esta opción es la más apropiada si el fichero XML es muy grande. Por lo general, el documento HTML resultante tendrá un tamaño bastante inferior al del fichero XML, y por tanto podrá visualizarse sin problemas mediante el navegador.