



---

# Normalización de datos de las IDE locales de Navarra

---

## GeoPackage, qué es y cómo se usa

### Participantes:

Fernando Alonso-Pastor

Álvaro Huarte

Pili Munárriz

José Manuel Vázquez

Yolanda Zuasti

**Editor:** Fernando Alonso-Pastor

**Estado:** Borrador

## Versiones:

N.º versión	Fecha	Autor	Comentarios
0.01	2019-02	José Manuel Vázquez	Redacción inicial
0.02	2019-02	Fernando Alonso-Pastor	Revisión
0.03	2019-03	José Manuel Vázquez	Estructura de un GeoPackage
0.04	2019-03	Fernando Alonso-Pastor	Nueva revisión
1.0	2019-03	José Manuel Vázquez	Cierre versión 1.0
1.01	2019-04	Álvaro Huarte, Yolanda Zuasti	Revisión
1.02	2019-07	José Manuel Vázquez	Simbología QGIS
1.03	2019-07	Fernando Alonso-Pastor	Revisión
2.0	2019-07	José Manuel Vázquez	Cierre versión 2.0

## Objeto del documento

Este documento trata de explicar qué es GeoPackage, por qué usarlo y cómo.

## Índice

<b>1 ¿TAN MALO ES EL SHAPEFILE?.....</b>	<b>4</b>
1.1. Ventajas.....	4
1.2. Inconvenientes.....	4
1.3. Alternativas.....	5
<b>2 GEOPACKAGE.....</b>	<b>6</b>
2.1. Ventajas.....	6
2.2. Inconvenientes.....	6
<b>3 ¿CÓMO UTILIZAR GEOPACKAGE?.....</b>	<b>7</b>
3.1. Importar datos a un proyecto QGIS.....	7
3.2. Crear capas nuevas en un GeoPackage.....	9
3.3. Añadir Información a una capa existente en un GeoPackage.....	11
3.4. Guardar la simbología en un Geopackage.....	14
3.5. Cómo es mi GeoPackage.....	15
<b>4 SOFTWARE NO GEOGRÁFICO PARA ACCEDER A UN GEOPACKAGE.....</b>	<b>17</b>
DB Browser for SQLite.....	17
Microsoft Access.....	18
<b>5 CÓMO ES POR DENTRO UN GEOPACKAGE.....</b>	<b>19</b>
<b>6 REFERENCIAS.....</b>	<b>21</b>



# 1 ¿TAN MALO ES EL SHAPEFILE?

Es un formato de trabajo y, sobre todo, de intercambio de datos geoespaciales que se viene utilizando desde principios de la última década del siglo XX.

Tiene unas pocas ventajas, y muchos inconvenientes.

## 1.1. Ventajas

- Es, de lejos, el formato más soportado por los distintos softwares de información geográfica.
- Aunque su formato es propietario, de ESRI, su especificación es abierta. (algo así, como el Microsoft Word).
- Para muchos casos de uso, puede ser suficiente. Los ficheros de índice (\*.shx) aceleran la lectura de estos archivos y es relativamente eficiente en términos de espacio, comparado con formatos de texto, como GML.
- Es muy eficiente en aplicaciones de publicación de mapas sin filtros alfanuméricos, porque la geometría no está empaquetada.
- Admite geometrías no válidas (con lazos, puntos duplicados,...). Nunca se sabe qué calidad tiene nuestra información...

Aquí se acaban las ventajas.

## 1.2. Inconvenientes

- Un shapefile se compone de varios archivos que un cliente SIG lee como uno único. El mínimo requerido es de tres:
  - .shp: almacena las entidades geométricas de los objetos.
  - .shx: almacena el índice de las entidades geométricas.
  - .dbf: tabla dBASE (formato presentado ¡en 1986!) donde se almacenan los atributos de los elementos geométricos.
  - Opcionalmente puede tener un .prj, .sbn, .sbx, .fbn, .fbx, .ain, .aih, .shp, .xml.
- Sin definición de sistema de referencia de coordenadas: Por defecto no hay definición del sistema de referencia de coordenadas utilizado. Se puede usar, por ejemplo, el archivo .prj, pero no es una parte estándar de la especificación
- Los nombres de atributos están limitados a 10 caracteres y con restricciones (sin tildes, ñ, etc)
- Solo 255 atributos. El archivo DBF no le permite almacenar más de 255 campos de atributos.
- Tipos de datos limitados: A float (decimal), integer (entero), date (fecha) y text (texto) con un máximo de 254 caracteres. No hay soporte para campos de datos más avanzados como blobs, imágenes o arrays.
- Conjunto de caracteres desconocido: No hay forma de especificar el conjunto de caracteres utilizado en la base de datos. Muchas aplicaciones están usando las viejas codificaciones de datos de Windows – \* o ISO- \*, mientras que hoy en día estamos tendiendo a usar UTF-8. Todavía no hay forma de especificar esto en el encabezado del archivo. El soporte para caracteres Unicode también es muy limitado. Puede existir un archivo .cpg pero no todos los SIG lo leen.
- Está limitado a 2 GB de tamaño de archivo. Aunque algunas herramientas pueden superar este límite, nunca pueden superar los 4 GB de datos.
- Sin topología en los datos: No hay forma de describir las relaciones topológicas en el formato.
- Solamente puede almacenar una geometría por archivo: No hay forma de guardar geometrías vectoriales mixtas, como puntos y polígonos.

- Las estructuras de datos más complicadas son imposibles de salvar. Es un formato de “tabla plana”.
- El soporte 3D es muy limitado: No hay forma de almacenar datos 3D con texturas o aspectos como las definiciones de materiales.

## 1.3. Alternativas

Podemos considerar el shapefile como un formato caduco. Veamos qué alternativas tenemos para sustituirlo:

- **KML.** La popularidad de KML fue alta hace años, pero desde 2015 su popularidad ha disminuido. Como está basado en XML, no es eficiente para almacenar conjuntos de datos grandes. Un KML combina la cartografía junto con la geometría de los datos en un solo archivo, lo cual es problemático cuando los datos tienen el potencial de ser utilizados de múltiples maneras. Dado que oficialmente solo admite el sistema de referencia de coordenadas (WGS-84), no es adecuado para una serie de aplicaciones.
- **GeoJSON.** En el caso de GeoJSON, el problema es que no se pueden representar todas las geometrías y los sistemas avanzados de referencia de coordenadas no son del todo compatibles. Recomendamos utilizar GeoJSON para el intercambio de datos particularmente para servicios web y visores webmapping.
- **GML** es un candidato para el reemplazo de Shapefile para el intercambio de datos en situaciones donde los datos son demasiado complejos para ser representados por GeoJSON. Sin embargo, una desventaja importante de GML es que es un estándar increíblemente complejo. Pocos paquetes de software admiten todo el estándar y el soporte para partes individuales del estándar varía ampliamente. Sin embargo, para la gran mayoría de los conjuntos de datos GML es excesivo.
- **ESRI GeoDatabase.** Formato propietario de ESRI. Es muy complejo y con características avanzadas, pero solo utilizable con herramientas ESRI.
- **GeoPackage**, es la alternativa más recomendable, que describiremos a continuación.

## 2 GEOPACKAGE

GeoPackage es un formato abierto, basado en estándares, independiente de la plataforma, auto descriptivo y compacto para el intercambio de información geoespacial.

### 2.1. Ventajas

- Es un [estándar](#) del OGC (Open Geospatial Consortium).
- Se basa en [SQLite 3](#), es decir contiene, además de los datos, un [sistema gestor de base de datos](#) con sus funciones, sus relaciones, su lenguaje de consultas y definición de datos... todo, excepto la gestión de usuarios (es monousuario). No es un formato desconocido, en cualquier smartphone de hoy en día tenemos unas cuantas bases de datos SQLite (los contactos, las fotografías, el whatsapp, etc).
- Dispone de índices espaciales. Se recomienda el uso de índices espaciales cuando el número de registros excede del millar. El uso de índices espaciales incrementa la velocidad de búsquedas espaciales y su visualización en los SIG de escritorio.
- Se trata de un único archivo **.gpkg**, por lo que es ideal para transferir información geoespacial.
- Desaparece el problema del conjunto de caracteres, siempre es UTF-8.
- Podemos almacenar multitud de tipos de geometrías en un mismo archivo **.gpkg**: Point, Line, Polygon, MultiPoint, MultiLine, MultiPolygon, CompoundCurve, CurvedPolygon, MultiCurve y MultiSurface.
- En el mismo archivo también podemos almacenar tablas sin geometría, que pueden utilizarse, por ejemplo, para listas controladas de valores.
- En el mismo archivo, también podemos almacenar una matriz de teselas e imágenes ráster a diferentes escalas.
- Cada geometría tiene definido su sistema de referencia de coordenadas, y pueden ser distintas unas de otras.
- Soporta el uso directo, lo que significa que podemos acceder y actualizar sus datos en un formato de almacenamiento nativo sin necesidad de conversiones intermedias. Podemos acceder a los datos de GeoPackage de forma “nativa” sin traducciones de formato intermedio.
- GeoPackage destaca por su flexibilidad y se puede utilizar de muchas maneras, por lo que puede reemplazar al formato shapefile. GeoPackage, a diferencia de un shapefile, se ha diseñado para almacenar datos complejos y voluminosos (hasta 140 TB). Además los atributos de las geometrías pueden contener nombre muy largos.
- GDAL soporta entidades de GeoPackage. Esta librería de código abierto escrita en C sirve para convertir formatos de datos. GDAL es la librería más ampliamente utilizada. 100 programas de GIS utilizan la librería GDAL internamente en sus procesos espaciales, lo que nos lleva al siguiente punto:
- Casi todos los software GIS de referencia son capaces de leer archivos GeoPackage. Por ejemplo ArcGIS o QGIS pueden leer y escribir. Y se pueden utilizar en ArcGIS Pro, GeoServer (importando y sirviendo los datos como estándar web de la OGC), o Leaflet. Otros programas que soportan GeoPackage son: GeoTools, OpenJUMP PLUS, Skyline, Luciad, Envia MapLink, SpatiaLite, FME Desktop y FME Server, TerraGO. Es decir, no vamos a tener problema para compartir nuestros datos con otros colegas de trabajo.
- Los GeoPackages son particularmente útiles en dispositivos móviles y en entornos de comunicación en los que hay conectividad y ancho de banda limitados.

### 2.2. Inconvenientes

- La base de datos SQLite subyacente es un formato binario complejo.

### 3 ¿CÓMO UTILIZAR GEOPACKAGE?

Nuestra recomendación, al igual que con un shapefile, es utilizar QGIS.

Vamos a utilizar como ejemplo, un proyecto para gestionar los aparcamientos de un municipio.

En esta URL

[http://ww2.pcypsitna.navarra.es/SITNA/EntornosColaboracion/IDeMunicipales/Biblioteca%20de%20documentos/2.%20Proyectos%20QGIS%20de%20trabajo/IDEMUN\\_Pol\\_Aparcamientos.zip](http://ww2.pcypsitna.navarra.es/SITNA/EntornosColaboracion/IDeMunicipales/Biblioteca%20de%20documentos/2.%20Proyectos%20QGIS%20de%20trabajo/IDEMUN_Pol_Aparcamientos.zip), encontraremos un .zip con dos archivos:

- IDEMUN\_Pol\_Aparcamientos.qgs: el proyecto QGIS para tratar la información
- IDEMUN\_Pol\_Aparcamientos.gpkg: el GeoPackage con los datos.

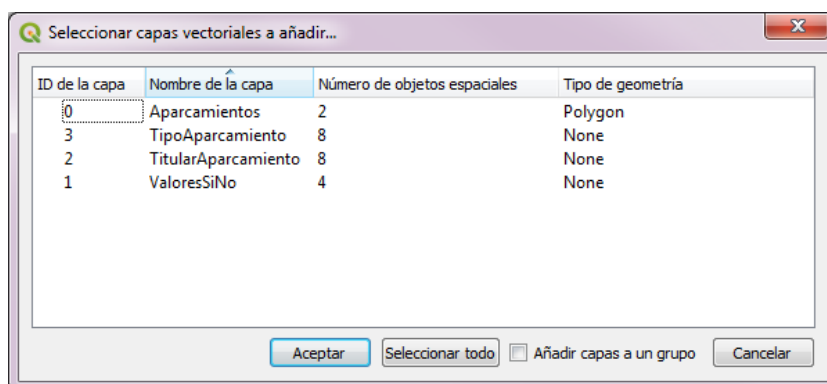
Para entender mejor el contexto del ejemplo, se puede consultar este [documento](#).

Por cierto, si no utilizásemos geopackage, en lugar de un único .gpkg, necesitaríamos bastantes más archivos:

- IDEMUN\_Pol\_Aparcamientos.cpg: conjunto de caracteres del shapefile.
- IDEMUN\_Pol\_Aparcamientos.dbf: atributos de los objetos del shapefile.
- IDEMUN\_Pol\_Aparcamientos.prj: sistema de coordenadas (ESRI) del shapefile.
- IDEMUN\_Pol\_Aparcamientos.qpj: sistema de coordenadas (QGIS) del shapefile.
- IDEMUN\_Pol\_Aparcamientos.shp: geometrías de los objetos del shapefile.
- IDEMUN\_Pol\_Aparcamientos.shx: índice de las geometrías de los objetos del shapefile.
- TipoAparcamiento.ods: lista controlada de valores.
- TitularAparcamiento.ods: lista controlada de valores.
- ValoresSiNo.ods: lista controlada de valores.

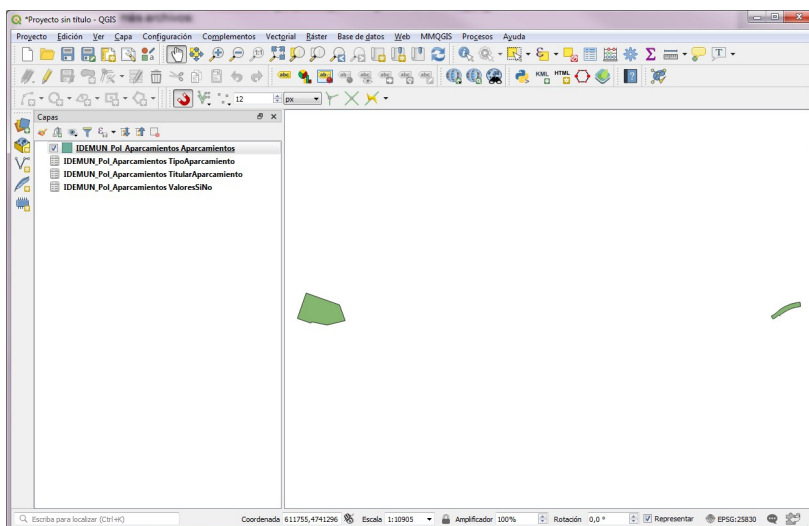
#### 3.1. Importar datos a un proyecto QGIS

Tanto si lo añadimos desde el administrador de fuentes de datos como si arrastramos el fichero .gpkg al panel de capas de QGIS, nos preguntará qué capas queremos añadir al proyecto.

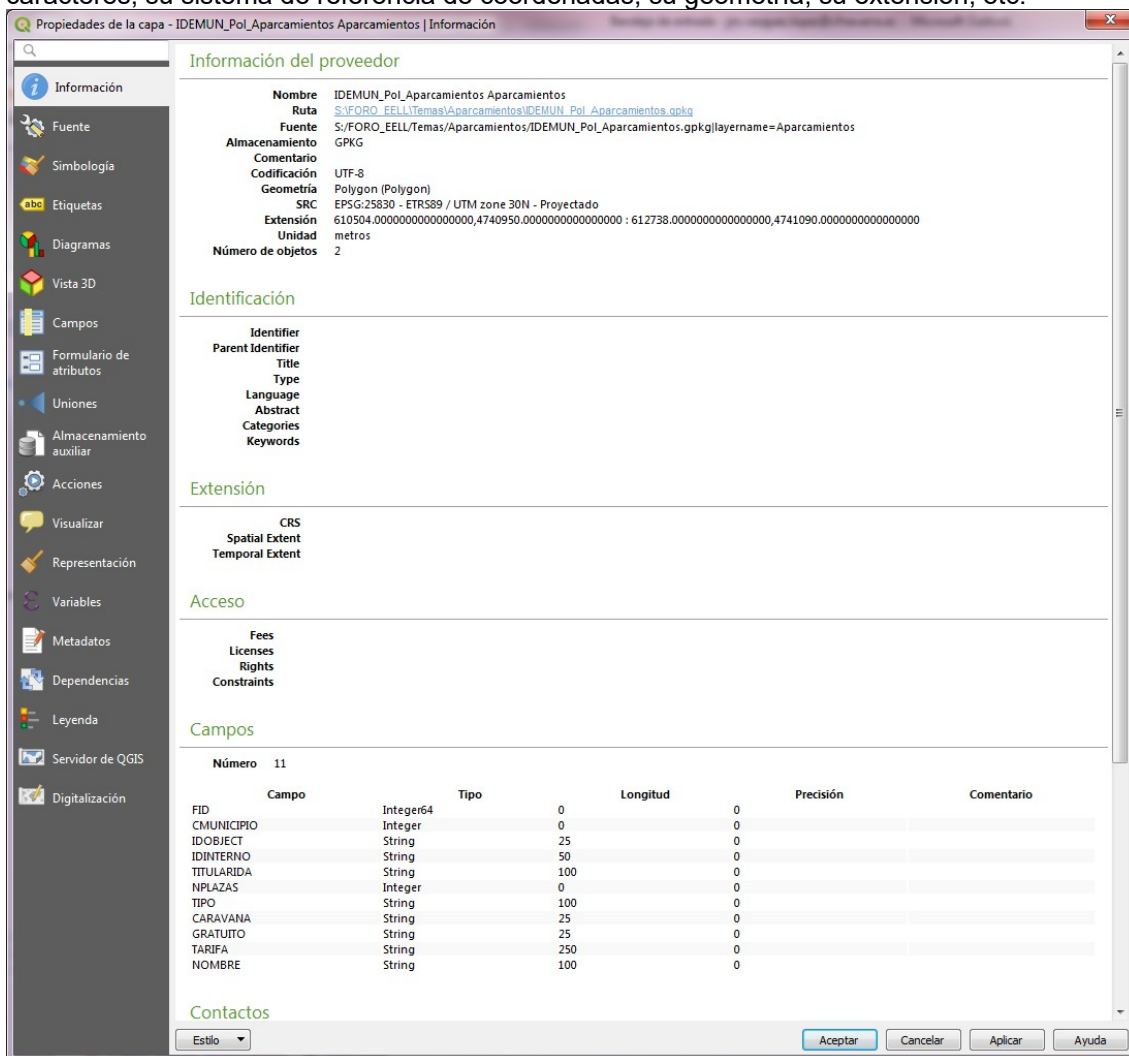


Vemos que, en este caso, tenemos una capa vectorial (polígono) y tres capas sin geometría.

Si seleccionamos todas, las tendremos disponibles para visualizar los datos, transformarlos, añadir y eliminar atributos, modificar geometrías, como si de un shapefile se tratase.



Por ejemplo, la capa Aparcamientos, aparece perfectamente definida, con su codificación de caracteres, su sistema de coordenadas, su geometría, su extensión, etc.



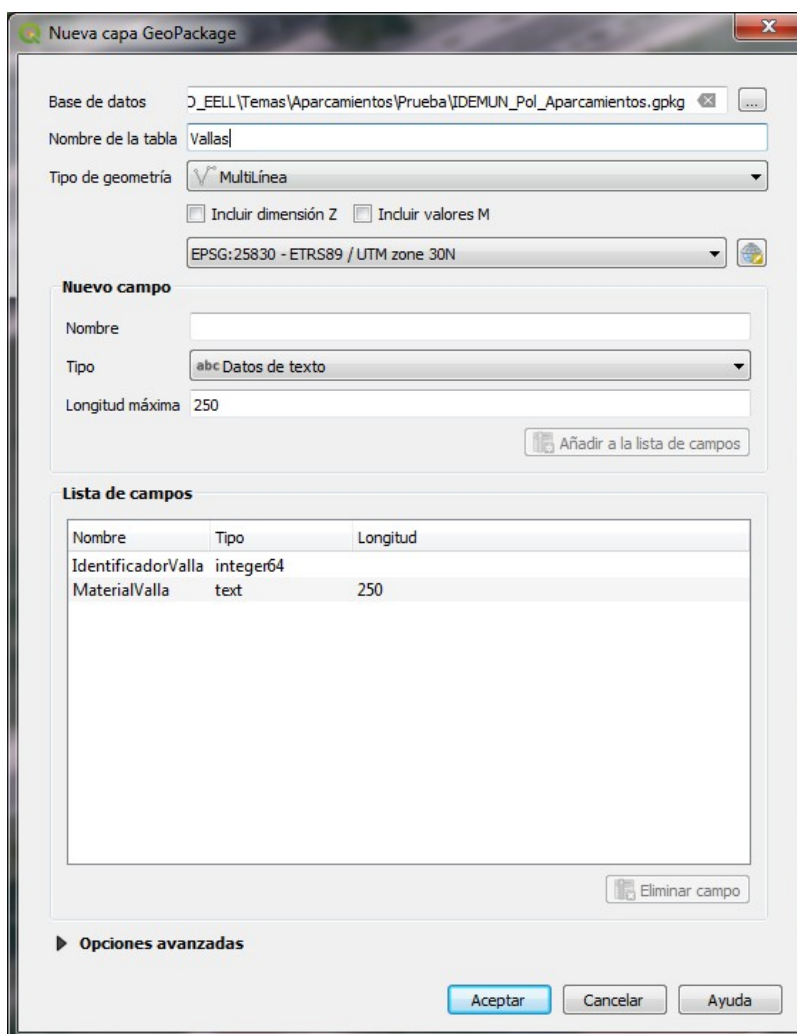


## 3.2. Crear capas nuevas en un GeoPackage

A un GeoPackage se le pueden añadir tantas capas como se desee, de cualquier tipo de geometría (o ninguna) y en el sistema de referencia de coordenadas que se quiera.

Supongamos que queremos añadir una capa para gestionar el vallado de los aparcamientos, para ello, vamos a crear en ese GeoPackage una capa nueva de tipo multilinea, a la que llamaremos *Vallas*.

Para ello, pulsamos el botón *Nueva capa GeoPackage*:



**Nueva capa GeoPackage**

Base de datos: C:\EELL\Temas\Aparcamientos\Prueba\IDEMUN\_Pol\_Aparcamientos.gpkg

Nombre de la tabla: Vallas

Tipo de geometría: MultiLinea

Incluir dimensión Z  Incluir valores M

EPSG:25830 - ETRS89 / UTM zone 30N

**Nuevo campo**

Nombre: [ ]

Tipo: abc Datos de texto

Longitud máxima: 250

Añadir a la lista de campos

**Lista de campos**

Nombre	Tipo	Longitud
IdentificadorValla	integer64	
MaterialValla	text	250

Eliminar campo

Opciones avanzadas

Aceptar Cancelar Ayuda

**Base de datos:** Seleccionamos nuestro geopackage

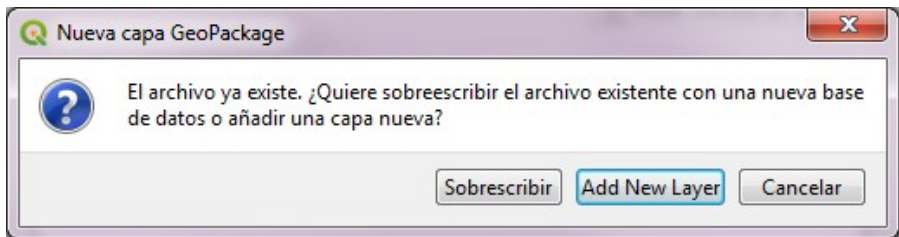
**Nombre de la tabla:** Se refiere al nombre de la capa dentro de nuestro geopackage. **Muy importante**, por defecto, tendremos el nombre de nuestro geopackage. Escribir ahí el nombre que queremos que tenga la capa o tabla.

**SRC:** En este caso, hemos seleccionado *EPSG:25830*.

**Geometría:** seleccionar *Multilinea*.

**Nuevo campo y Lista de campos:** vamos definiendo el nombre, tipo y longitud de los atributos de la capa. Para cada campo, lo escribimos en el apartado *Nuevo Campo* y pulsamos el botón *Añadir a la lista de campos*.

Cuando hayamos terminado, pulsamos el botón *Aceptar*.

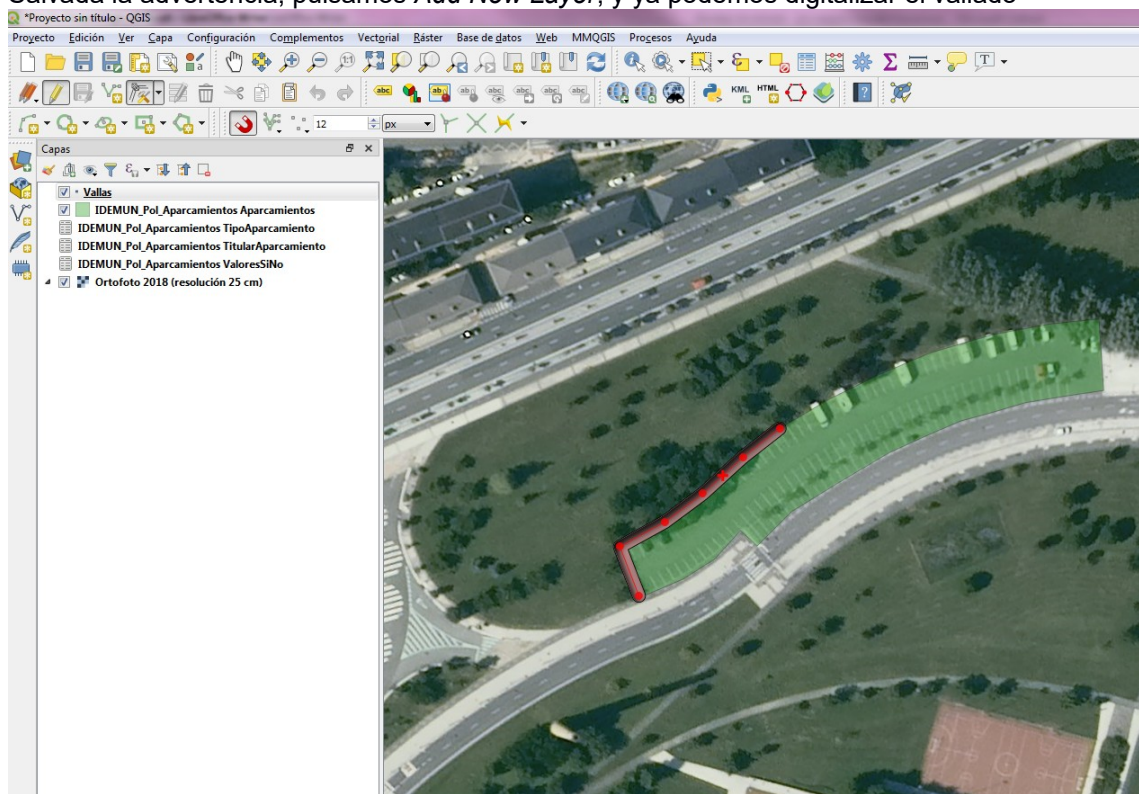


## ¡¡ Atención !!

Tenemos que pulsar **Add New Layer**, para crear la capa Vallas dentro de nuestro GeoPackage.

Si pulsamos *Sobrescribir*, lo haremos sobre el geopackage entero (es decir, en lugar de un fichero con un montón de capas, nos quedaremos con un fichero con una única capa, la de Vallas y **perderemos todo lo demás**).

Salvada la advertencia, pulsamos *Add New Layer*, y ya podemos digitalizar el vallado

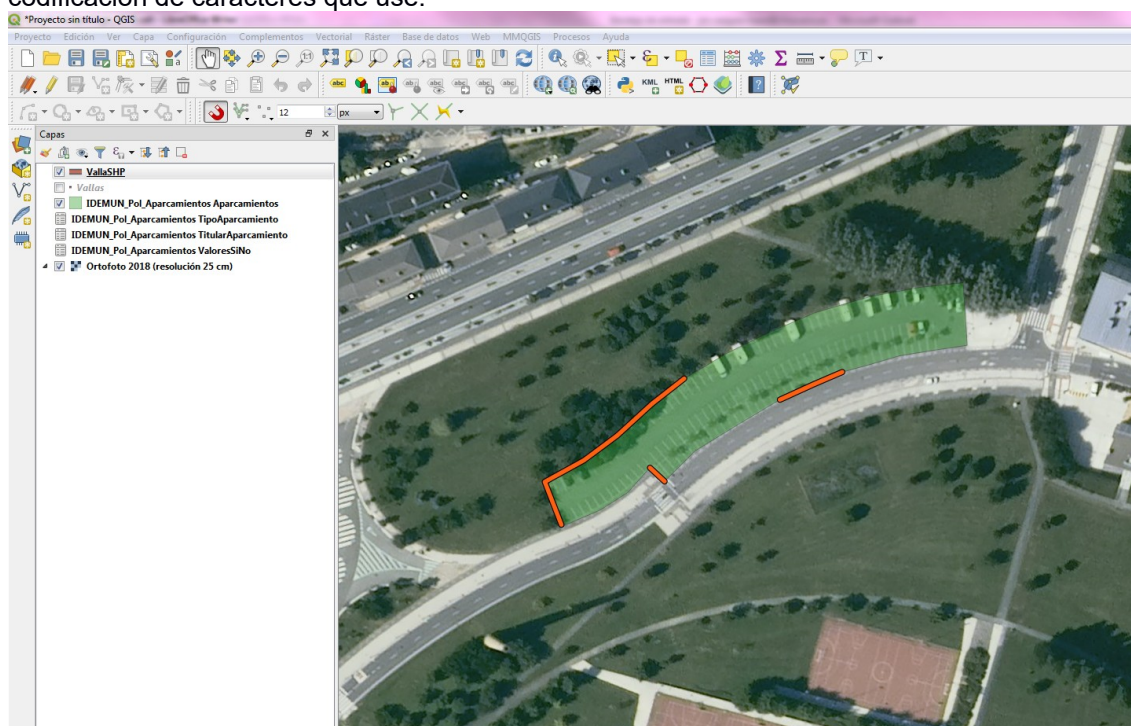


### 3.3. Añadir Información a una capa existente en un GeoPackage

Como hemos dicho, a un GeoPackage se le pueden añadir tantas capas como se desee, de cualquier tipo de geometría (o ninguna) y en el sistema de referencia de coordenadas que se quiera.

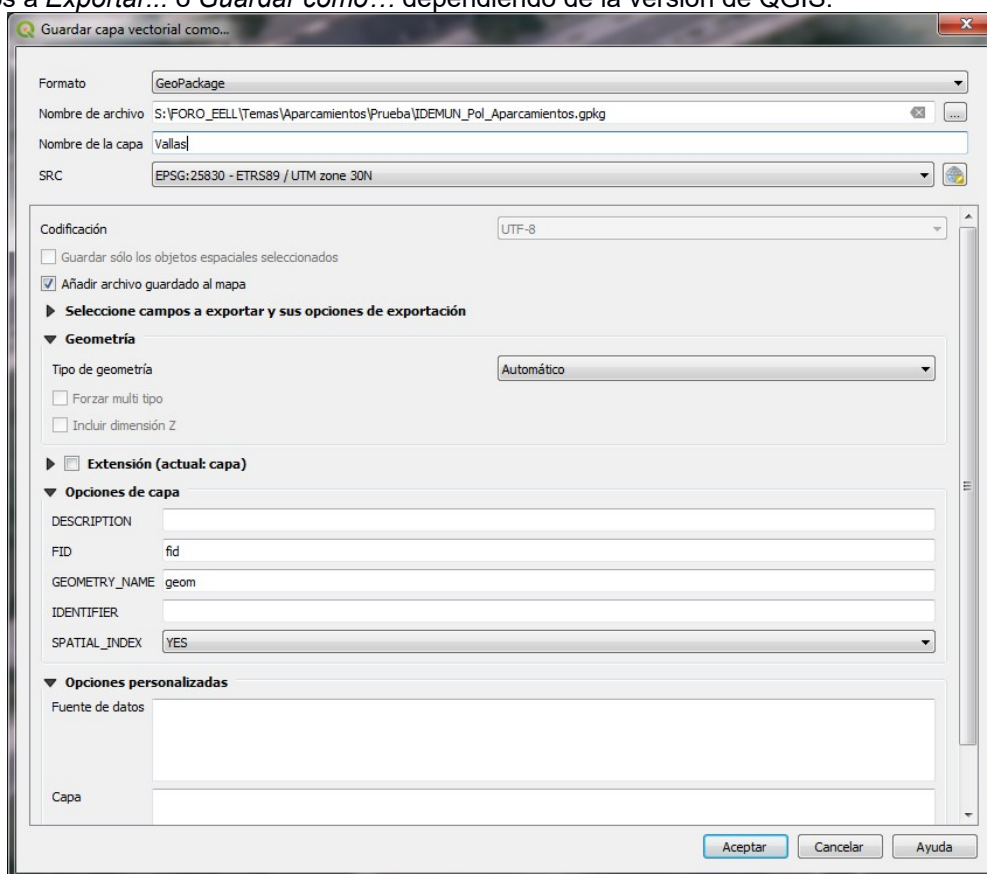
Siguiendo con el caso anterior, supongamos que ya teníamos el vallado de los aparcamientos en un shapefile (o un KML o un GML o un excel con coordenadas), y lo que queremos es volcar esa información a la capa que acabamos de crear en nuestro GeoPackage.

Para ello, añadimos ese shapefile (en nuestro caso *VallasSHP*) a nuestro proyecto QGIS. Hay que tener cuidado en el sistema de referencia de coordenadas en el que lo tengamos y en la codificación de caracteres que use.



Ahora tenemos que conseguir tener los mismos campos en la capa origen y destino. La forma más sencilla es crear campos virtuales con la calculadora de campos.

Una vez que tenemos, como mínimo, los mismos atributos en la capa shapefile que los existentes en la capa GeoPackage, seleccionamos la capa shapefile y, con botón derecho, nos vamos a *Exportar...* o *Guardar como...* dependiendo de la versión de QGIS:



**Formato:** Seleccionamos GeoPackage (o *PaqueteGeo*, en otras versiones).

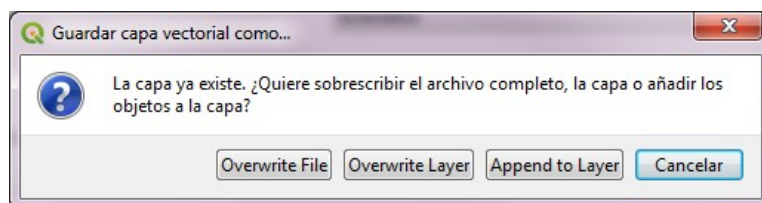
**Nombre de archivo:** Con el botón Explorar, seleccionamos el geopackage sobre el que estamos trabajando.

**Nombre de la capa:** Se refiere al nombre de la capa dentro de nuestro geopackage. **Muy importante**, poner bien escrito el nombre de la capa existente (en nuestro caso Vallas), si no creará una nueva dentro del geopackage.

**SRC:** el que deseemos, en nuestro caso EPSG:25830.

**Geometría:** seleccionar *Automático*.

Una vez pulsemos *Aceptar*, nos aparecerá el siguiente mensaje:



## ¡¡ Atención !!

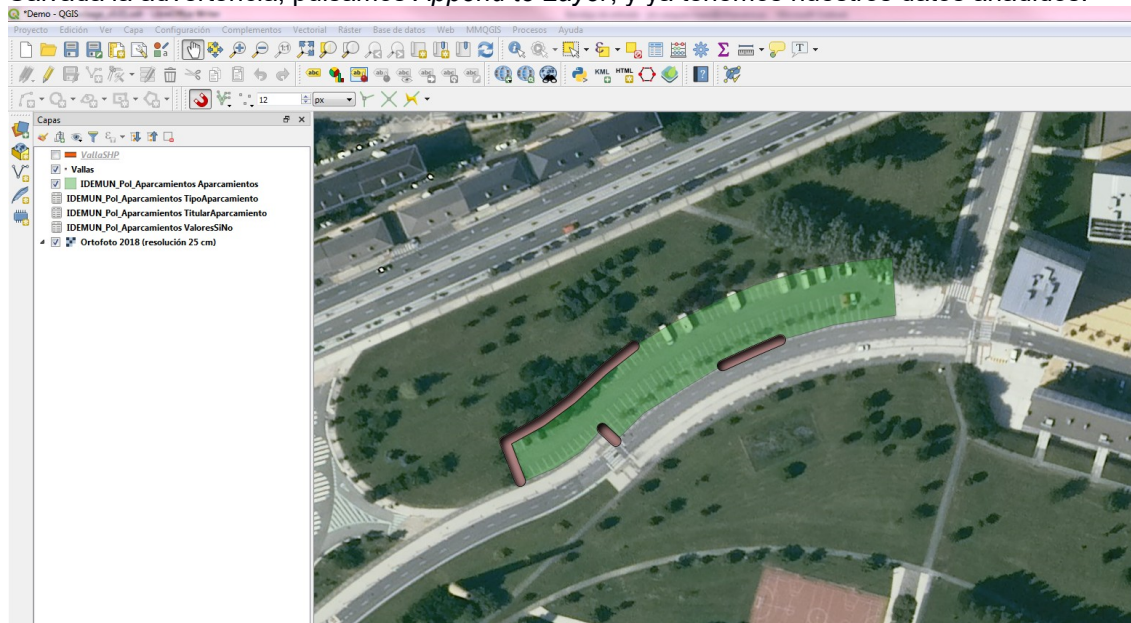
Tenemos que pulsar **Append to Layer**, para añadir nuestros datos a los ya existentes en la capa Vallas.

Si pulsamos *Overwrite layer*, sustituiremos la capa que ya existe con ese nombre (es decir si,

por lo que sea hemos cometido algún error en los nombre de campos, lo trasladamos al geopackage).

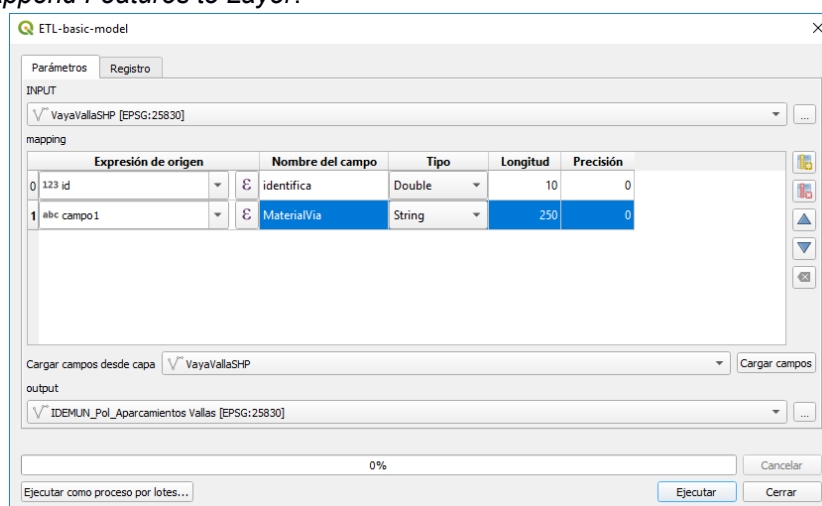
Si pulsamos *Overwrite file*, reescribimos el geopackage entero (es decir, en lugar de un fichero con un montón de capas, nos quedaremos con un fichero con una única capa, la de Vallas y **perderemos todo lo demás**).

Salvada la advertencia, pulsamos *Append to Layer*, y ya tenemos nuestros datos añadidos:



Siguiendo con el mismo ejemplo del vallado, si el modelo de datos que teníamos en el shapefile nos parece adecuado, podemos obviar el paso 3.2 y añadir directamente la capa Vallas a nuestro geopackage tal y como lo hemos hecho en el paso 3.3.

Otra forma de añadir datos a una capa existente desde otra (u otro archivo) utilizar el plugin *Append Features to Layer*:

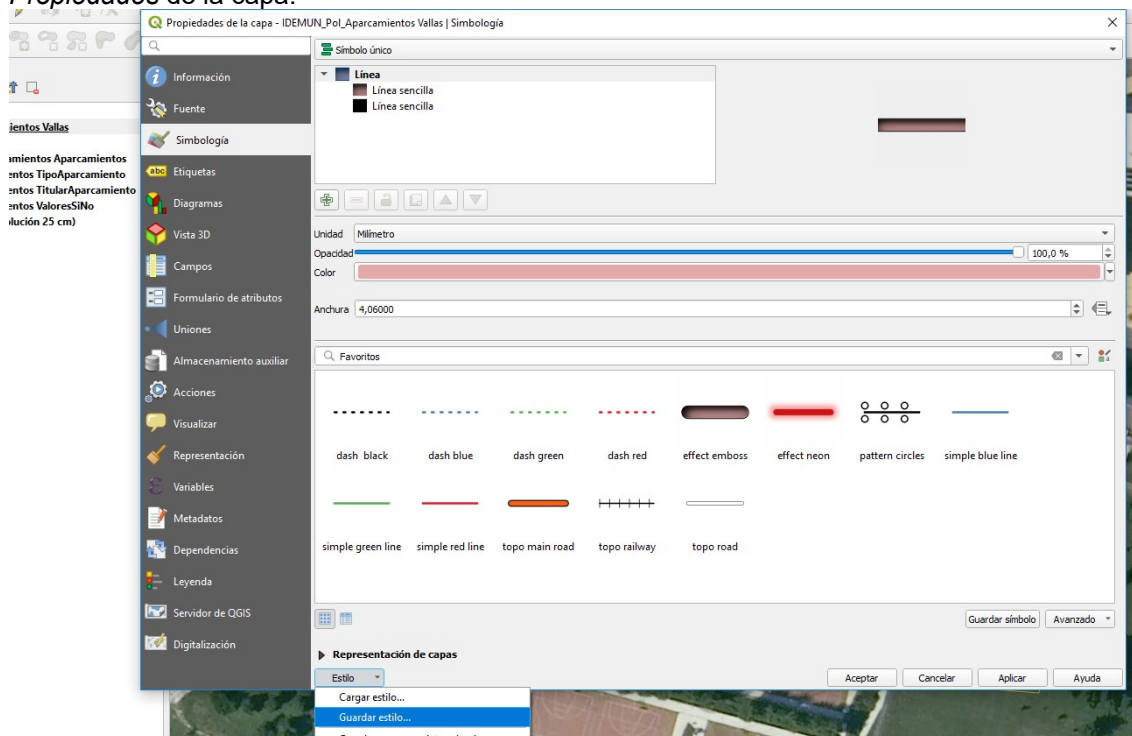


Todo lo explicado en los puntos 3.1., 3.2 y 3.3. es válido para capas sin geometría. La única diferencia es que da igual lo que pongamos en *SRC* y que en el *Tipo de geometría* tendremos que seleccionar *Ninguna geometría*.

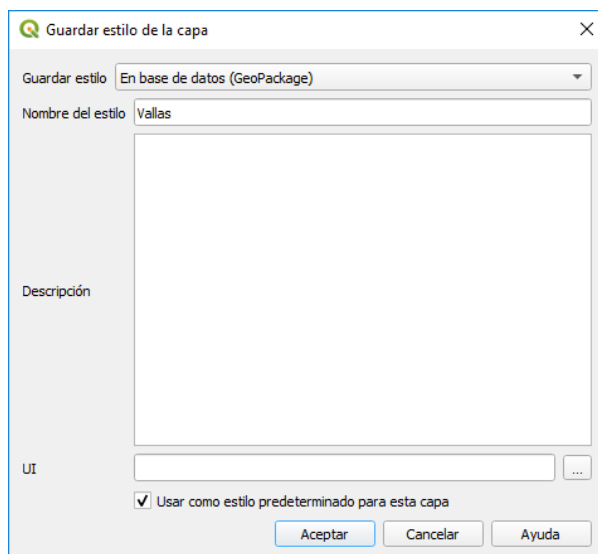
## 3.4. Guardar la simbología en un Geopackage

A partir de la versión 3, podemos guardar la simbología de nuestras capas QGIS en el mismo geopackage donde tenemos los datos o en otro, como deseemos). Así, podremos abrir el geopackage desde un proyecto QGIS en blanco y las capas se representarán con la simbología que hayamos guardado en él.

Para hacerlo, con doble click (o botón derecho) sobre la capa deseada, abrimos la ventana de *Propiedades de la capa*:



En la parte inferior, pinchamos *Estilo* y seleccionamos *Guardar estilo...*



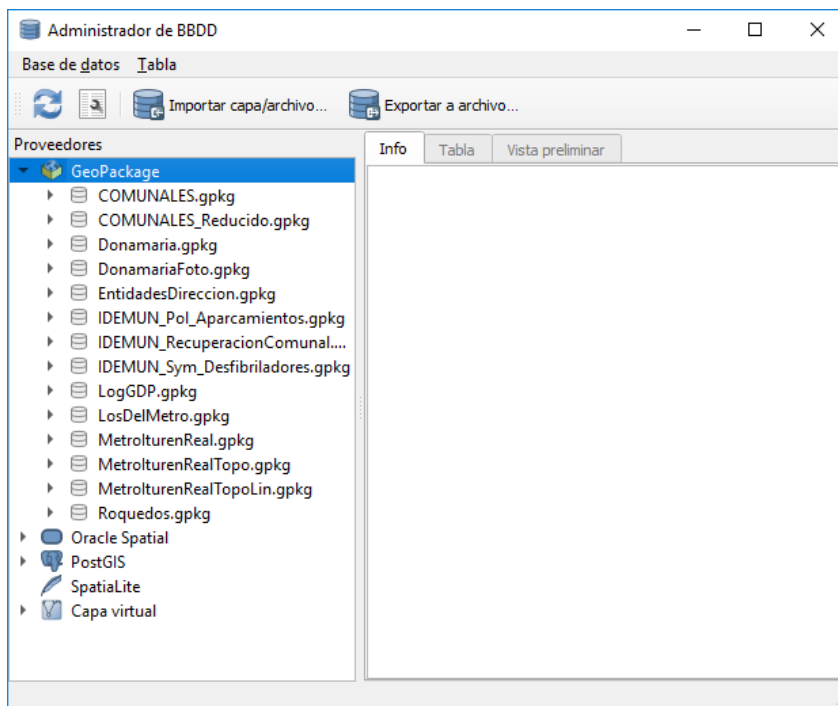
Y le damos un nombre, en este caso Vallas. También podemos decirle que sea el estilo por defecto de la capa.

De esta forma, quedará almacenado en el geopackage la simbología (y el etiquetado, y las acciones y el formulario de datos... ¡todo!) que hayamos definido para esta capa durante nuestra sesión de QGIS.

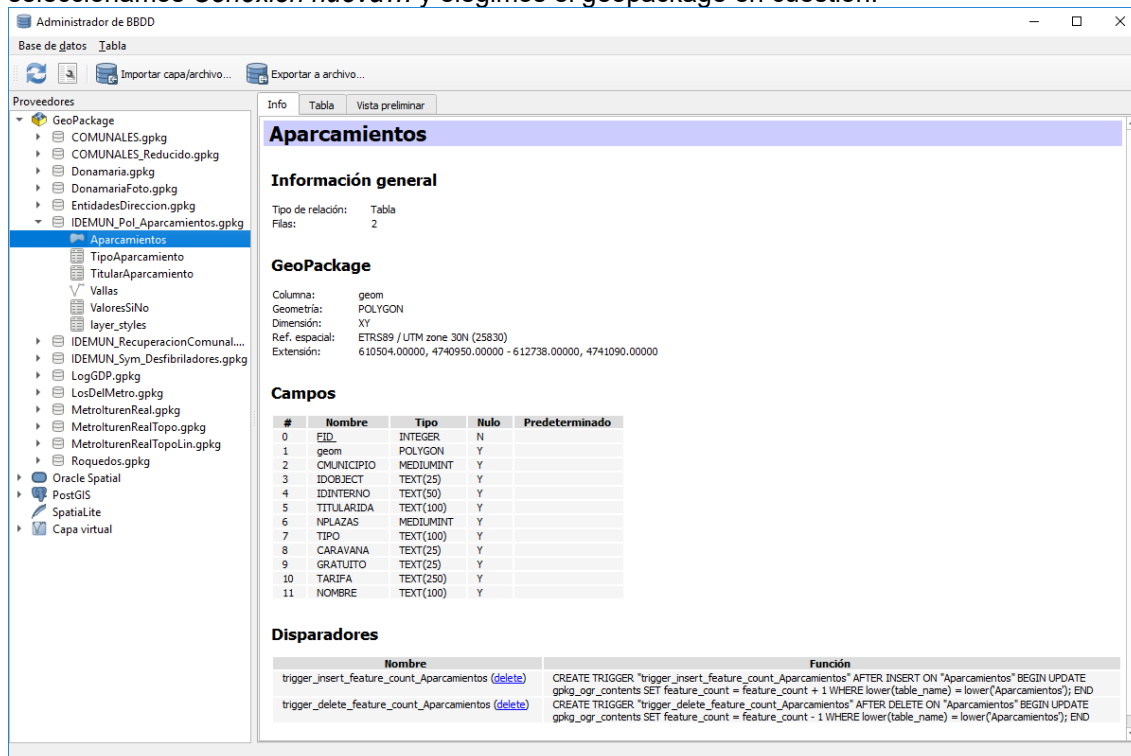
## 3.5. Cómo es mi GeoPackage

Desde el *Administrador de Bases de Datos* de QGIS podemos ver el con algo más de detalle el contenido del geopackage que hemos estado manipulando en el ejemplo anterior.


Para ello, vamos al menú *Base de datos* → *Administrador BBDD*:

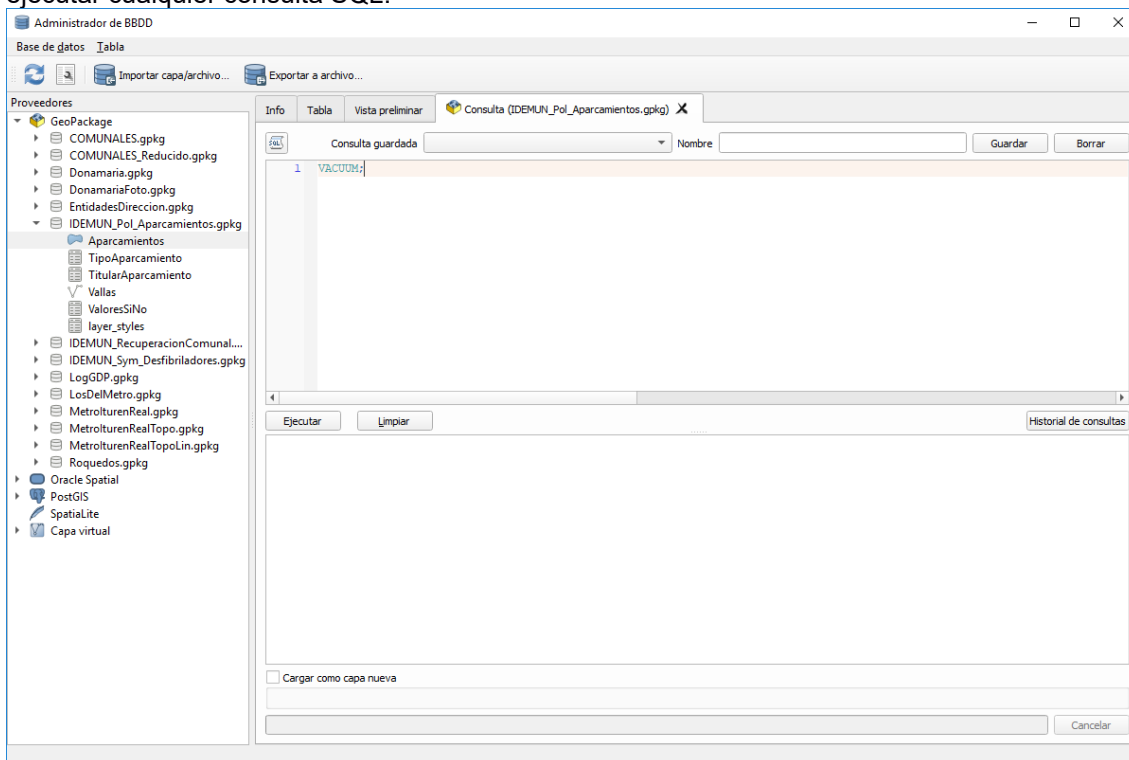


Si no vemos nuestro geopackage, con botón derecho sobre el proveedor GeoPackage, seleccionamos *Conexión nueva...* y elegimos el geopackage en cuestión:



Y vemos las capas de nuestro geopackage, para cada capa, pulsando las pestañas de la parte superior, veremos la información general de la capa, los datos de la misma y, si tiene geometría, una vista preliminar.

Si pulsamos el botón  activaremos una nueva pestaña, *Consulta SQL*, donde podremos ejecutar cualquier consulta SQL:



Una muy útil, es ejecutar *VACUUM*; que elimina el espacio no utilizado del geopackage y reduce su tamaño.



## 4 SOFTWARE NO GEOGRÁFICO PARA ACCEDER A UN GEOPACKAGE

En ocasiones puede que queramos realizar operaciones no espaciales sobre los datos subyacentes de nuestro geopackage. Por ejemplo, actualizaciones masivas de un atributo. Aunque desde QGIS podemos hacer más operaciones de este tipo de las que pensamos, como con el *Administrador de BBDD*. Puede que estemos acostumbrados a abrir directamente el .dbf de un shapefile con Libre Office y “tocarlo” para actualizar datos... aunque no es una práctica muy recomendable, por peligrosa. Tampoco es recomendable hacerlo con un geopackage.

Cuidado, si usamos estas herramientas veremos que aparecen más tablas de las que vemos con el Administrador de BBDD, eso es porque son tablas que se manejan desde un GIS, no es aconsejable modificarlas desde otro tipo de herramientas, ya que podemos dejar el geopackage inutilizable.

Si a pesar de todo lo anterior seguimos con la misma idea, ¿cómo lo hago ahora con un geopackage? Aquí tenemos dos opciones:

### DB Browser for SQLite

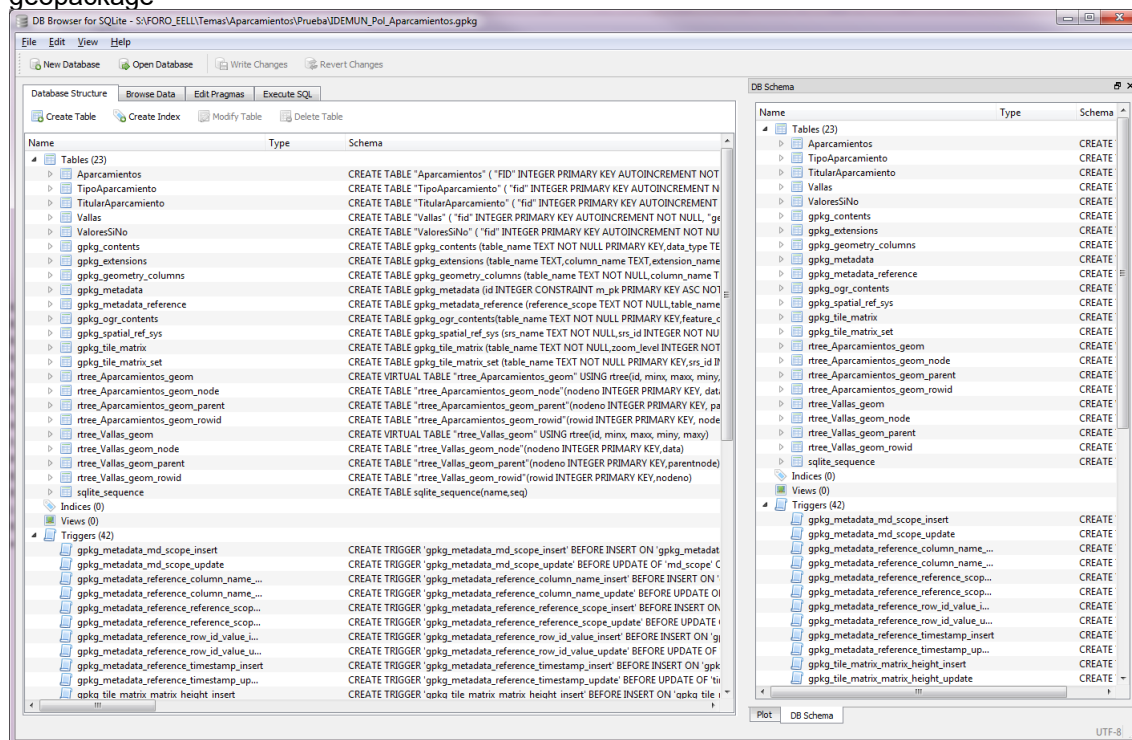
DB Browser for SQLite es una consola para SQLite que permite ver las tablas que existen, su estructura, sus datos y permite realizar cualquier operación del estándar SQL. Para que reconozca los geopackages hay que añadir la extensión espacial como se explica en <https://github.com/sqlitebrowser/sqlitebrowser/wiki/SpatialLite-on-Windows>.

Para instalarlo (en Windows):

Hay que descargar (de <https://sqlitebrowser.org/dl/>) y ejecutar el instalable adecuado para nuestro equipo.

Hay que descargar el archivo [mod\\_spatialite-4.3.0a-win-amd64.7z](#) (o el archivo [mod\\_spatialite-4.3.0a-win-x86.7z](#) para 32 bits) y descomprimirlo en la carpeta donde haya quedado instalado DB Browser for SQLite (normalmente será C:\Program Files\DB Browser for SQLite).

Después, solo hay que ejecutar el icono que habrá quedado en nuestro escritorio... y abrir el geopackage

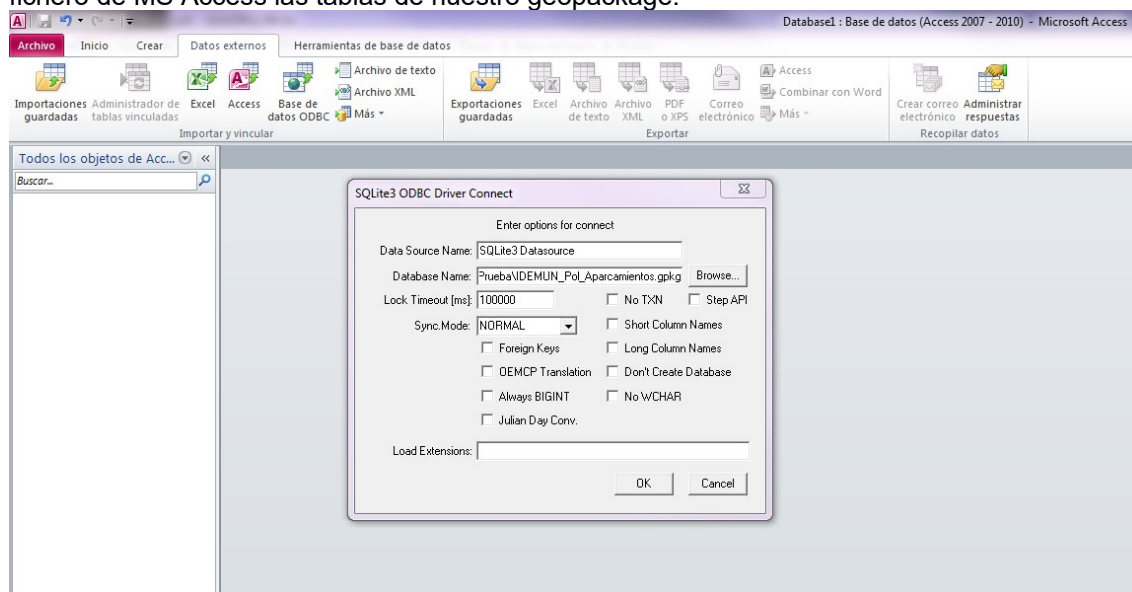


# Microsoft Access

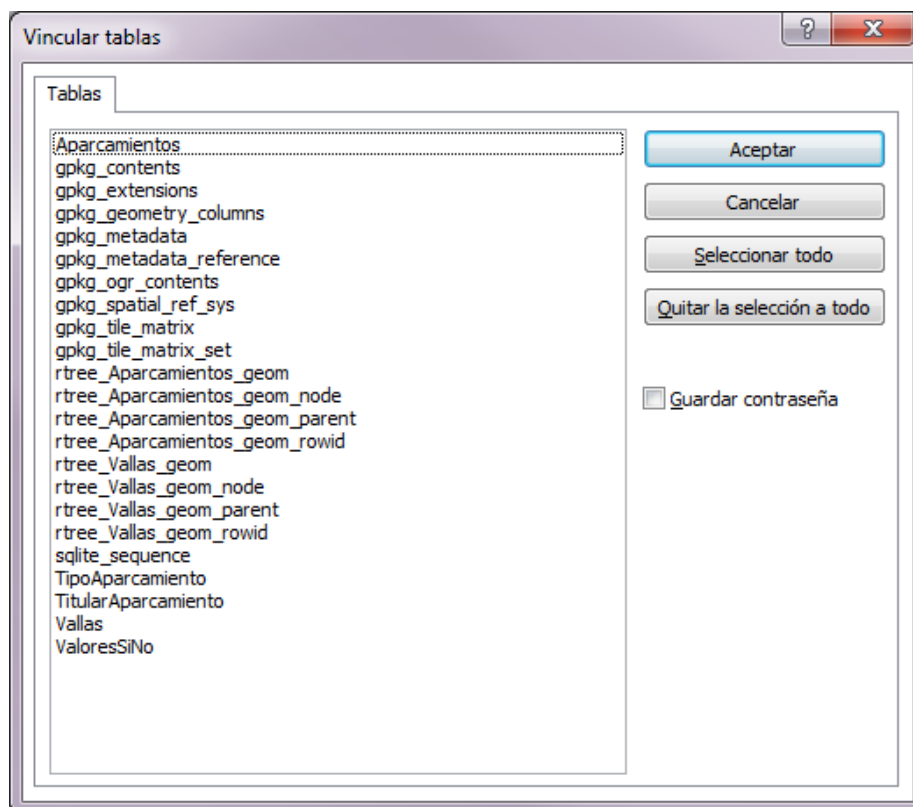
También es posible acceder a las tablas vía [ODBC](#).

Para ello, tenemos que descargarnos el [SQLite ODBC Driver](#). Se recomienda instalar los dos, el de 32 y el de 64 bits: [sqliteodbc.exe](#) y [sqliteodbc\\_w64.exe](#).

De esta forma (*Datos externos* → *Base de datos ODBC*) podremos importar o anexar a un fichero de MS Access las tablas de nuestro geopackage:

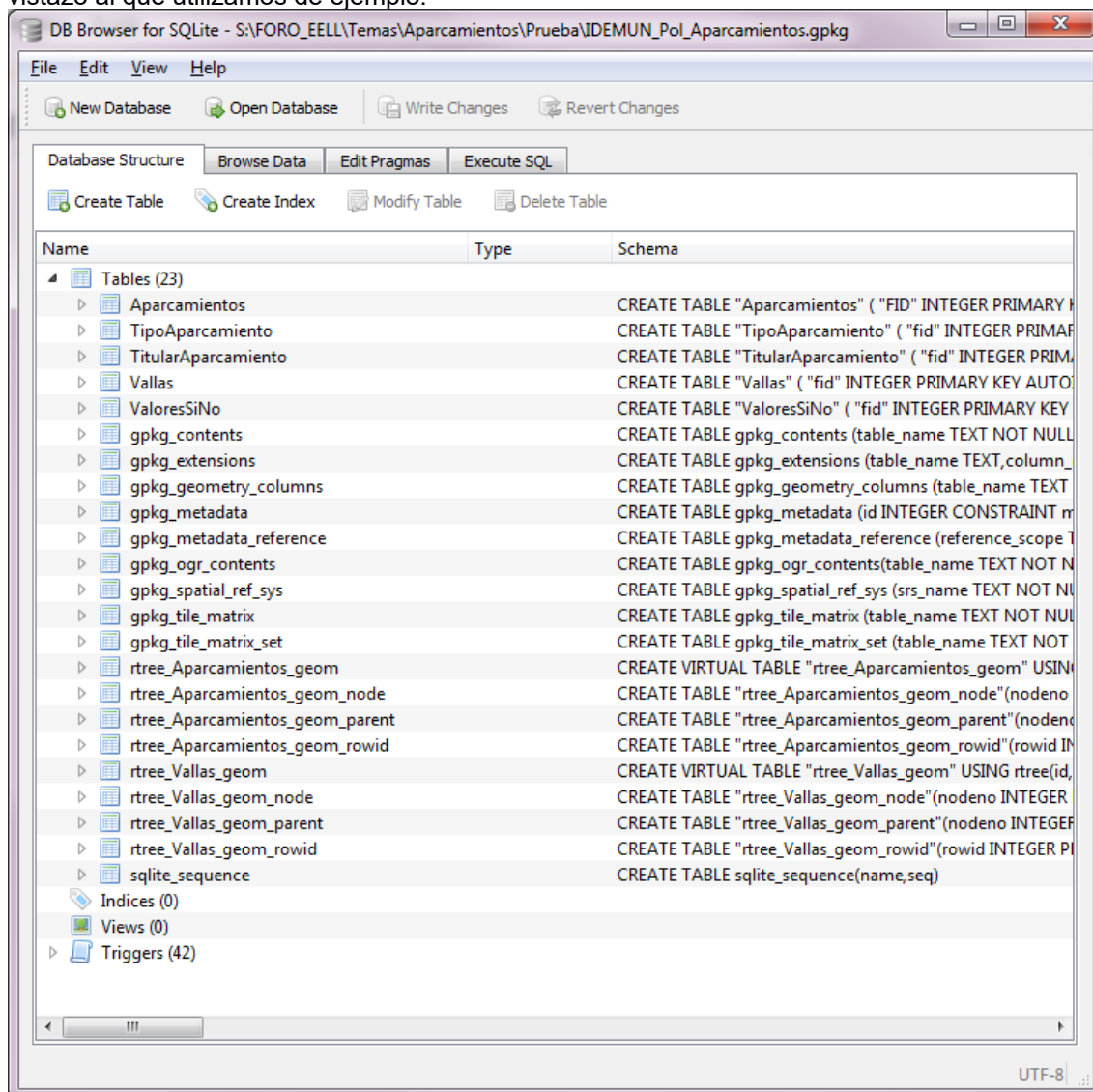


Y veremos todas las tablas que tenemos en nuestro geopackage:



## 5 CÓMO ES POR DENTRO UN GEOPACKAGE

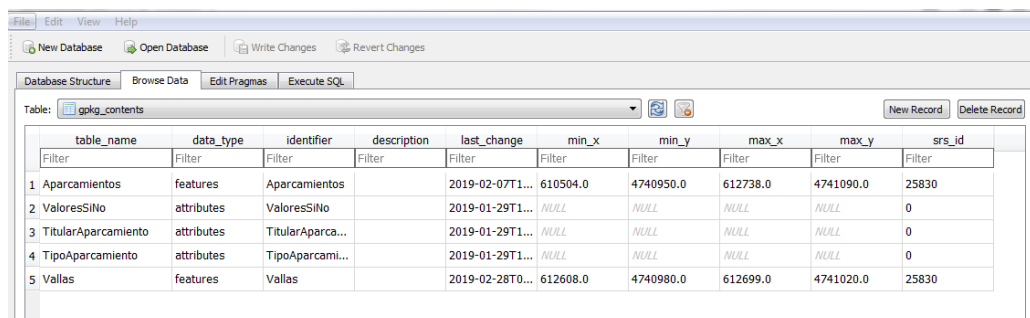
En el capítulo anterior hemos visto cómo acceder al contenido de un geopackage. Echemos un vistazo al que utilizamos de ejemplo:



**Tablas de usuario:** son las que utilizamos para almacenar nuestros datos o nuestras listas controladas (Aparcamientos, TipoAparcamiento, TitularAparcamiento, Vallas, ValoresSiNo).

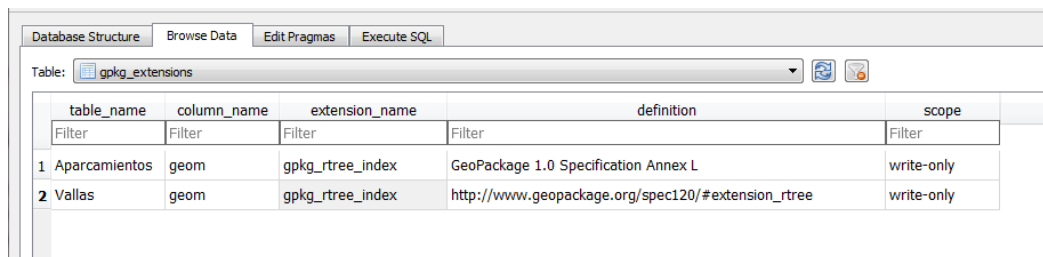
El resto de tablas es mejor no tocarlas, o si es necesario, hacerlo con sumo cuidado.

- **gpkg\_contents:** por cada una de nuestras tablas de usuario, existirá una entrada en esta tabla donde se indica, entre otras cosas, el tipo de datos, el bounding box (coordenadas que delimitan la entidad geográfica: min\_x, min\_y, max\_x, max\_y) o el sistema de referencia de coordenadas (en notación [EPSG](#)).



	table_name	data_type	identifier	description	last_change	min_x	min_y	max_x	max_y	srs_id
1	Aparcamientos	features	Aparcamientos		2019-02-07T1...	610504.0	4740950.0	612738.0	4741090.0	25830
2	ValoresSiNo	attributes	ValoresSiNo		2019-01-29T1...	NULL	NULL	NULL	NULL	0
3	TitularAparcamiento	attributes	TitularAparca...		2019-01-29T1...	NULL	NULL	NULL	NULL	0
4	TipoAparcamiento	attributes	TipoAparcami...		2019-01-29T1...	NULL	NULL	NULL	NULL	0
5	Vallas	features	Vallas		2019-02-28T0...	612608.0	4740980.0	612699.0	4741020.0	25830

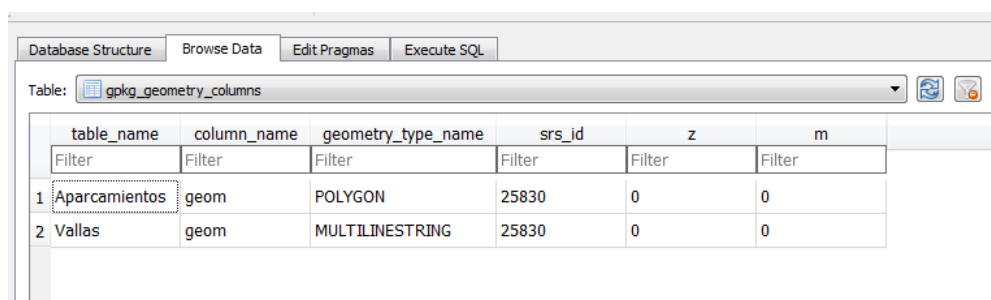
- **gpkg\_extensions**: se utiliza para saber si a todo el GeoPackage, a alguna tabla o a alguna columna le aplica alguna extensión. Las extensiones permiten añadir funcionalidades o tipos de datos o de geometrías a un GeoPackage.



table_name	column_name	extension_name	definition	scope
1 Aparcamientos	geom	gpkg_rtree_index	GeoPackage 1.0 Specification Annex L	write-only
2 Vallas	geom	gpkg_rtree_index	http://www.geopackage.org/spec120/#extension_rtree	write-only

En nuestro ejemplo, nos dice que las columnas de geometría de nuestras tablas de *Aparcamientos* y *Vallas*, se indexarán usando [árboles R](#).

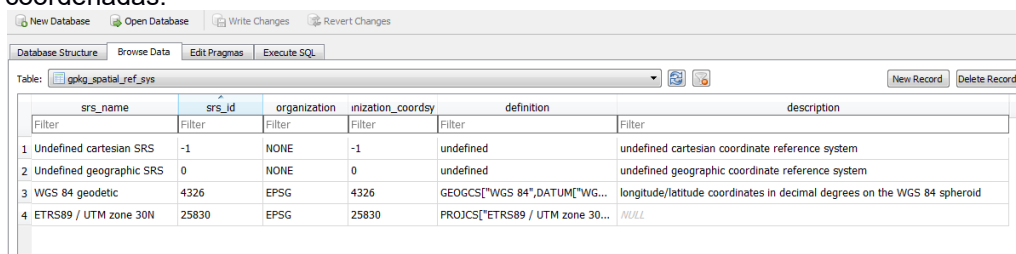
- **gpkg\_geometry\_columns**: si existen columnas de geometría, constarán en esta tabla, indicando tipo de geometría y sistema de referencia de coordenadas:



table_name	column_name	geometry_type_name	srs_id	z	m
1 Aparcamientos	geom	POLYGON	25830	0	0
2 Vallas	geom	MULTILINESTRING	25830	0	0

En nuestro ejemplo vemos que, tanto *Aparcamientos* como *Vallas*, tienen sendas columnas de geometría, de nombre geom.

- **gpkg\_metadata** y **gpkg\_metadata\_reference**: se utilizan para los metadatos del Geopackage o parte de él.
- **gpkg\_spatial\_ref\_sys**: contiene las definiciones de los sistemas de referencia de coordenadas.



srs_name	srs_id	organization	inization_coordsy	definition	description
1 Undefined cartesian SRS	-1	NONE	-1	undefined	undefined cartesian coordinate reference system
2 Undefined geographic SRS	0	NONE	0	undefined	undefined geographic coordinate reference system
3 WGS 84 geodetic	4326	EPSG	4326	GEOGCS["WGS 84",DATUM["WG...	longitude/latitude coordinates in decimal degrees on the WGS 84 spheroid
4 ETRS89 / UTM zone 30N	25830	EPSG	25830	PROJCS["ETRS89 / UTM zone 30...	<i>NULL</i>

- **gpkg\_tile\_matrix** y **gpkg\_tile\_matrix\_set**: se utiliza para los datos raster.
- **layer\_styles**: Se utiliza para guardar la simbología de nuestras capas QGIS.

## 6 REFERENCIAS

- [Switch from Shapefile](#). Jachym Cepicky.
- [Di no al Shapefile y sí al GeoPackage](#). Aurelio Morales.
- [GeoPackage para novatos: ventajas y uso en ArcGIS, QGIS, GeoServer y Leaflet](#). Aurelio Morales.
- [GeoPackage como alternativa a GML y SHP](#). Francisco Javier López-Pellicer.