



SpatiaLite, the Shapefile of the future?

Pirmin Kalberer
Sourcepole AG, Switzerland
www.sourcepole.com





About Sourcepole

- › **GIS-Knoppix: first GIS live-CD**
- › **QGIS**
 - › Core development & PSC membership
 - › QGIS Mapserver
- › **OGR / GDAL**
 - › Interlis driver
 - › schema support for PostGIS driver
- › **Ruby on Rails**
 - › MapLayers plugin
 - › Mapfish server plugin



Spatialite introduction

- › Spatial extension to SQLite embedded database
- › Like PostGIS extends PostgreSQL
- › Single file storage, SQL support
- › Spatial data types & functions
- › Licenses: MPL, GPL , LGPL



History

- › **Alessandro Furieri, Italy**
- › **Initial development for management of timetables for train and buses**
- › **Tuscany Region Government (Transportation Dept.)**
- › **Version 1.0: 21.3.2008**
- › **Current Version 2.3.1 (2.4.0RC3a)**

› **Embedded database**

- › File based, no server process
- › C programming language
- › Access via library (<275KiB)
- › SQL API
- › Dynamic type system

› **Well-proven and widely used**

- › Version 1.0: 17.8.2000
- › Mozilla, PHP, Symbian, Apple, Adobe, Skype, ...

› **Good SQL-92 compatibility**

- › Joins, views, triggers
- › Transactions



SQLite Locking

- › Write ahead logging (WAL) introduced in 3.7
- › Before that write access locked database also for reading
- › Activation: `PRAGMA journal_mode=WAL`
- › Reading and writing can proceed concurrently
- › Spatialite 2.4.0RC3 uses SQLite 3.7.0



Spatialite data types

- › **OGC Simple Feature types (WKT, WKB)**
 - › Point, MultiPoint
 - › LineString, MultiLineString
 - › Polygon, MultiPolygon
 - › XY, XYZ, XYM, XYZM
- › **R*Tree index tables or BBOX In-Memory-Cache**
- › **Networks (Routing)**
- › **Raster data**



Spatialite Features

- › **GEOS library**
- › **Geometry access**
 - › ST_GeomFromText, ...
 - › ST_AsText, ST_Area, ST_IsValid, ...
- › **Queries and operations**
 - › ST_Intersects, ST_Contains, ST_Touches, ...
 - › ST_Union, ST_Intersection, ST_Buffer,
 - › ST_Generalize, SimplifyPreserveTopology
- › **OGC Metadata**
 - › geometry_columns, spatial_ref_sys tables



Spatial SQL examples

```
SELECT t2.Name, t2.Peoples,  
        ST_Distance(t1.geom, t2.geom) AS Distance  
FROM Towns AS t1, Towns AS t2  
WHERE t1.Name = 'Firenze' AND  
        ST_Distance(t1.geom, t2.geom) < 10000;
```

```
SELECT Regions.Name, COUNT(*)  
FROM Towns, Regions  
WHERE ST_Within(Towns.geom, Regions.geom)  
AND Towns.ROWID IN  
(SELECT pkid FROM idx_Towns_geom WHERE  
  xmin > MbrMinX(Regions.geom) AND  
  xmax < MbrMaxX(Regions.geom) AND  
  ymin > MbrMinY(Regions.geom) AND  
  ymax < MbrMaxY(Regions.geom))  
GROUP BY Regions.Name;
```



More features

- › Built-in Shapefile import- and export
- › Coordinate reprojection via PROJ.4 library
- › Charset conversion with GNU libiconv
- › Use of external Shapefiles (+DBF) and CSV/TxtTab files as virtual tables
- › Raster data library
- › Library for networks and routing



Spatialite Tools

- › **Command line:**
 - › spatialite, sqlite
- › **GUI:**
 - › spatialite-gui
 - › spatialite-gis
- › **OSS GIS Tools:**
 - › QGIS
 - › OGR/GDAL (UMN Mapserver, ...)
 - › GeoTools (GeoServer, ...)
 - › FDO (MapGuide Open Source, FME, ...)



Import Shapefiles

The top part of the image shows a file explorer window with a directory of shapefiles and a SQLite database. A blue arrow points from the shapefiles to the SQLite database, and another blue arrow points from the SQLite database to the spatialite-gui application.

The file explorer shows a directory named 'hydr' containing the following files:

Name	Größe
buildings.dbf	1.3 MiB
buildings.prj	144 B
buildings.shp	3.3 MiB
buildings.shx	145.5 KiB
natural.dbf	1.2 MiB
natural.prj	144 B
natural.shp	15.6 MiB
natural.shx	132.8 KiB
places.dbf	1.5 MiB
places.shp	503.4 KiB
places.shx	143.9 KiB
points.dbf	2.5 MiB
points.prj	144 B
points.shp	930.7 KiB
points.shx	
railways.dbf	
railways.prj	
railways.shp	
railways.shx	
roads.dbf	
roads.prj	
roads.shp	
roads.shx	
waterways.dbf	
waterways.prj	
waterways.shp	
waterways.shx	

The SQLite database window shows a single file named 'hydr.sqlite' with a size of 132.7 MiB.

The spatialite-gui application window shows a SQL query and its results:

```
SELECT ROWID, "PK_UID", "osm_id", "name", "type", "population", "Geometry"
FROM "places"
ORDER BY ROWID
```

	ROWID	PK_UID	osm_id	name	type	population	
	1	1	588370	Merzenstein	village	144	B
	2	2	17328659	Wien	city	1626440	B
	3	3	17328660	Knoppen	village	205	B
	4	4	17328662	Knoppen-Melzen	hamlet	0	B
	5	5	17328664	Hallstatt	village	388	B
	6	6	17328667	Bad Aussee	village	1132	B
	7	7	17330426	Purkersdorf	village	8364	B

Current SQLite DB: /mnt/wal/projects/referate/fossgis10/hydr.sqlite



Shapefile comparison

- › **Less files (multiple layers, indices, ...)**
- › **Eliminate column name limits (>10!)**
- › **64-bit IEEE Floating point type**
- › **Embedded Unicode support**
- › **Integrated metadata (Projection, etc.)**
- › **Library with SQL API (Insert, Join, etc.)**



GUI Demos

- › **spatialite-gui**
- › **spatialite-gis**
 - › Routing
 - › Rasterdata
- › **QGIS**



Links

- › <http://www.gaia-gis.it/spatialite/>
- › <http://en.wikipedia.org/wiki/Shapefile>
- › http://gis.hsr.ch/wiki/HSR_Texas_Geo_Database_Benchmark
- › **OSGEO Live DVD!**



Thank you! - Questions?



Pirmin Kalberer
pka@sourcepole.ch