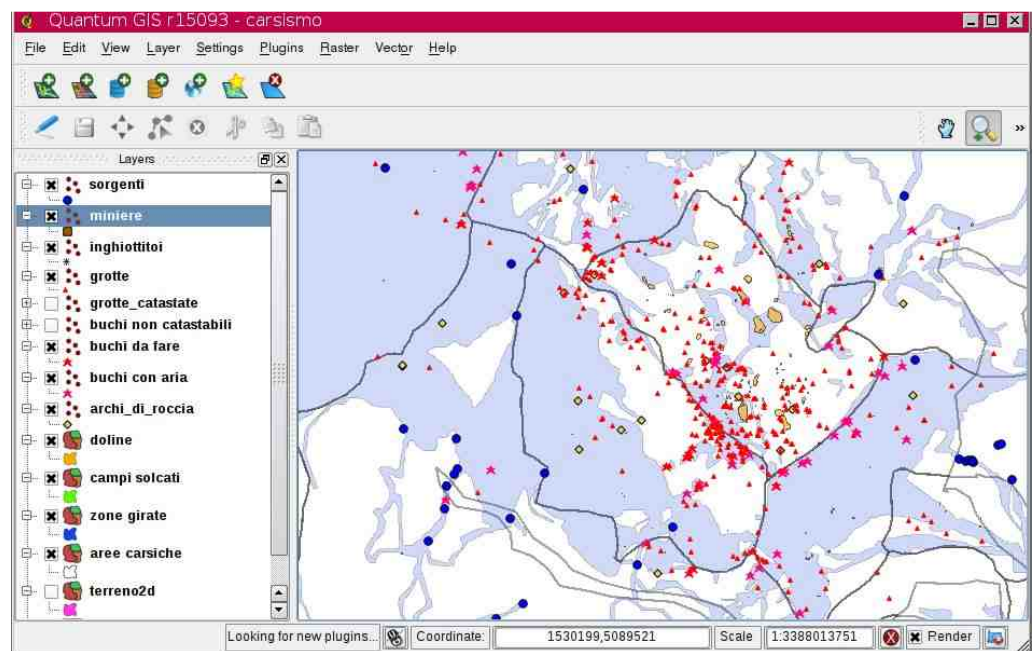


I GIS

I GIS (Geographical Information System) permettono di organizzare (cioè gestire) e visualizzare l'informazione, associando degli attributi delle entità geografiche alla loro posizione geografica. Con i GIS si possono dunque fare "elaborazioni geografiche". Per esempio cercare potenziali nuovi ingressi verificando quali punti di una grotta si avvicinano all'esterno, e dove. Oppure trovare l'areale di diffusione conosciuto di una specie troglobia, in una zona carsica.

Esempio. Il GIS del "Carsismo della Lombardia" realizzato da A. Maconi (v. figura). Contiene i seguenti oggetti:

- aree carsiche
- zone girate
- grotte
- grotte_catastate
- buchi_con_aria
- buchi_da_fare
- buchi_non_catastabili
- miniere
- inghiottitoi
- doline
- sorgenti
- archi_di_roccia
- campi_solcati
- fotografie
- fotografie_buchi_da_fare



Database del "carsismo". E' evidenziata la zona delle Grigne (A. Maconi)

La base geografica

La cartografia serve per la visualizzazione e il posizionamento sul terreno:

- carte CTR (per la Lombardia 1:10000)
- carta geologica
- ortofoto
- orografia (modelli DEM)

Attributi speleologici

Un elenco (incompleto) di attributi "speleologicamente" rilevanti:

- ingressi: posizione, tipo (orizzontale, verticale, ...)
- dati speleometrici: sviluppo, profondità
- sviluppo 3d della grotta
- dati topografici: data di esecuzione, strumenti usati, grado di accuratezza, note
- idrologia: flussi sotterranei (anche arrivi e diffluenze), laghi, sifoni
- flussi d'aria

- depositi: ghiaccio, neve, guano, (argilla ?), ...
- prosecuzioni
- scavi e disostruzioni
- campionamenti biospeleologici

Therion

Therion e' un sistema di codificazione dei rilievi ipogei in cui le informazioni vengono espresse in forma testuale. Therion ha un linguaggio molto ricco, capace di catturare la complessa semantica dei rilievi. Therion si presta dunque alla elaborazione dei rilievi al fine di produrre dati georeferenziati corredati da attributi da includere in GIS "speleologici".

Therion esporta i dati della poligonale in formato *database* (script SQL) fin dalla versione 0.2.18 (2004-02-12). Dalla versione 0.3.8 (2005-05-10) Therion esporta i modelli in formato *DXF*.

Ma e' con al versione 0.5.0 del 2007-02-02 che Therion diventa *geografico*. Vengono introdotte diverse caratteristiche *geografiche*:

- specifica del sistema di coordinate, attraverso il comando *cs*, sia per i dati che per l'output, e gestione delle conversioni (libreria proj4)
- generazione di mappe 2D e modelli 3D in formato shapefile (shapelib)
- generazione di mappe e modelli KML
- calcolo automatico della declinazione magnetica (geomag)
- attributi custom

Infine e' arrivata l'esportazione di informazioni sotto forma di liste (tabelle). Dal 2007-05-25 (pre-release della versione 0.5.1, 2007-08-29) c'e' la *continuation-list*. Dal 2008-01-30 la *survey-list* e dal 2008-04-16 la *cave-list*.

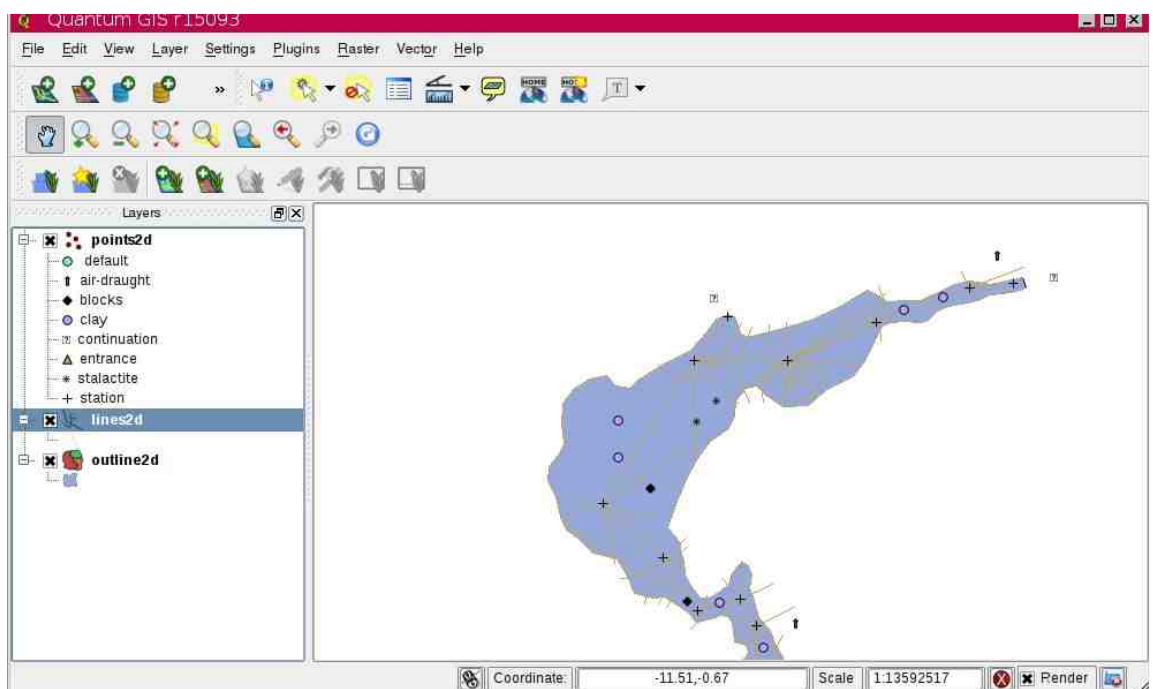
Export di shapefile 2D

Therion puo' generare le mappe in formato shapefile, che quindi possono esser caricate in un GIS,

```
export map -proj plan -o cave.shp
```

Gli shapefile generati comprendono

- outline2d (polygonz)
- lines2d (arcz), con attributi *type* e *subtype*. Inoltre, per le linee di tipo label, l'attributo *text*
- points2d (pointz), con attributi *type* e *subtype*. Inoltre, per punti di tipo station, gli attributi *stname* e *stsurvey*, mentre per i punti di tipo label o remark, l'attributo *text*
- areas2d (pologonz) con attributi *type* e *subtype*.



Esempio di pianta in formato shapefile, visualizzata in QGIS

Export di shapefile 3D

Il rilievo puo' esser esportato come modello shapefile (esri),

```
export model -format esri -o cave.esri
```

L'output consiste in tre shapefiles:

- stations3d, con attributi: *id, uid, name, survey, surface, fixed, entrance, continua, comment (*)*
- shots3d, con attributi: *from, to, survey, surface, duplicate, approxim.*
- walls3d, con attributi: *id, name, survey,*

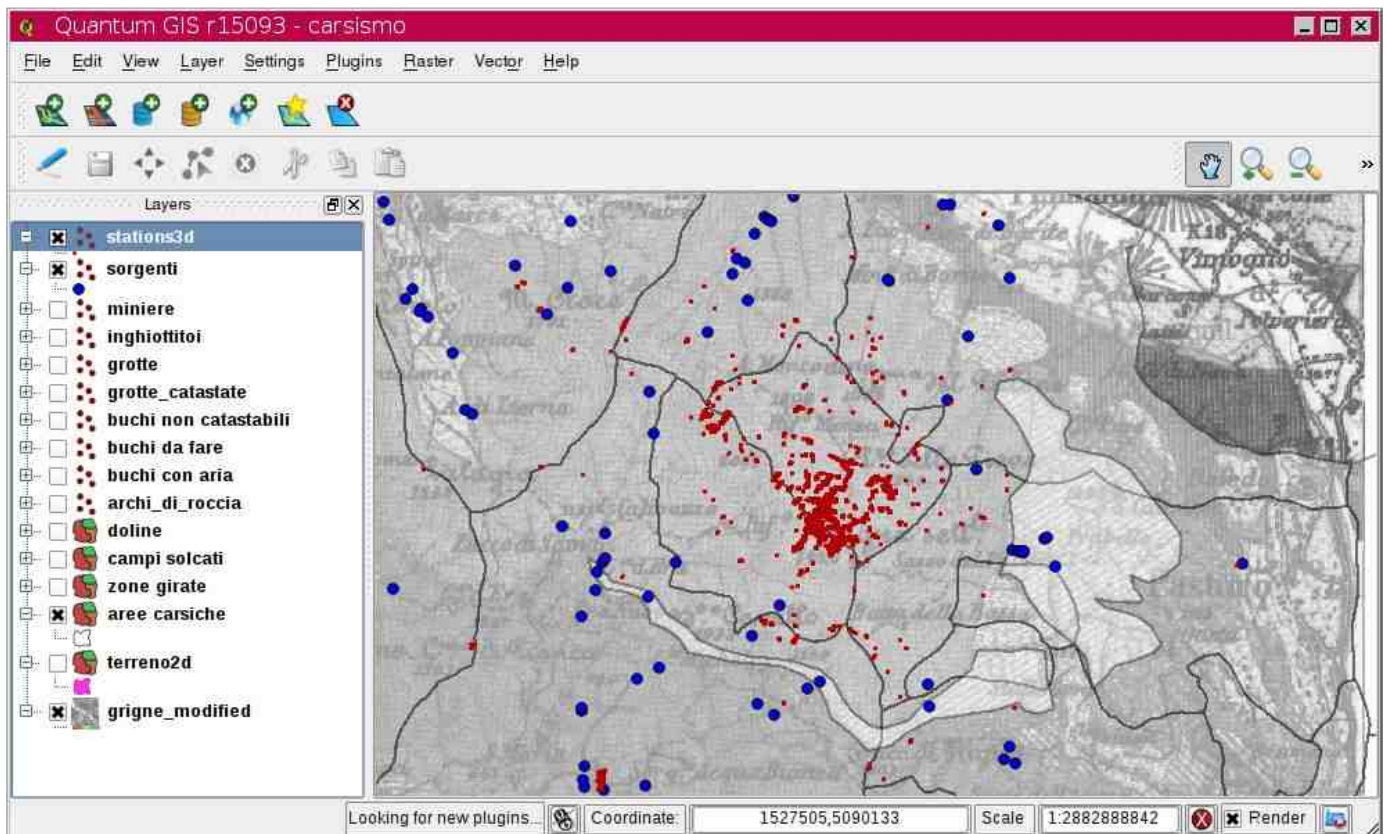
Note.

L'attributo *comment* di stations3d e' la stringa di commento che si mette sul comando *station* nel blocco centerline.

Questo comando puo' avere anche uno o piu' attributi custom; la sintassi e'

```
station <commento> <flag> attr <nome> <valore>
```

L'immagine mostra il risultato per le grotte della Grigna importato nel GIS del carsismo usando Qgis. E' visualizzata la carta geologica, i contorni delle aree carsiche, le sorgenti (cerchi blue) e i capisaldi. Sono stati utilizzati i dati di poligonale convertiti in Therion dal progetto in Compass. Purtroppo i dati in Compass non hanno molti attributi.



I dati delle poligonali delle grotte della Grigna inclusi nel GIS del "carsismo"

Export di liste

Un altro modo per cui Therion puo' esser utile per un GIS sono le liste. Therion produce tabelle, liste di attributi che possono essere inclusi in un database spaziale, ed interfacciati al GIS. I dati esportati da Therion sono automaticamente aggiornati nel GIS dopo l'elaborazione di Therion. In tal modo si tengono i dati delle grotte in un unico posto, cioe' nei rilievi.

Ci sono tre tipi di liste

- cave-list
- continuation-list
- survey-list

La prima, cave-list, include, oltre alle posizioni degli ingressi,

- profondita'
- sviluppo (esplorato, approssimativo, duplicato, superficiale)
- numero dei tiri
- numero delle stazioni

Le seconda, continuation-list, e' la lista dei punti aperti, le possibili prosecuzioni.

La terza e' la lista delle singole sessioni di rilievo. Essa include

- titolo
- lunghezza
- profondita'
- esplorato non rilevato
- approssimativo
- lunghezza tiri doppi
- numero dei tiri
- numero dei capisaldi
- attributi custom

Gli attributi "custom", utilizzati nella esportazione delle liste, possono esser definiti nel blocco centerline e nei disegni. In quest'ultimo caso la sintassi e' `-attr <nome> <valore>`. Gli attributi custom usati per questo export sono quelli dei seguenti elementi

- survey --> survey-list
- centerline
- scrap
- point
- line
- area
- map
- surface

The screenshot shows the spatialite-gui application window. The title bar reads "spatialite-gui [a GUI tool for SQLite/Spatialite]". The interface includes a file explorer on the left showing a tree view of a database with tables like "cave", "geom", and "geom_cols_ref". The main window displays a SQL query: `select * from cave where Distance(geom, GeomFromText('POINT(1529802 5089076)')) < 70`. Below the query, a table of results is shown with columns: PK_UID, Title, Length, and De. The table contains four rows of data.

PK_UID	Title	Length	De
1	Topino e le Giostre Lo Lc 5242	2571.000000	430.00
2	Probabile Probabile Abisso Lo Lc 5282	6.000000	5.0000
3	Pozzo Terroso Lo Lc 5320	5.000000	5.0000
4	Fessura del Crinoide	13.000000	10.0000

At the bottom of the window, it says "Current SQLite DB: /home/projects/compass2therion/grigna.sqlite".

La cave-list caricata nel database Spatialite

Esempio d'uso.

Il primo passo e' creare un database spaziale:

```
$ spatialite db.sqlite  
spatialite> .read init.sql cp1252
```

Alternativamente, con l'interfaccia grafica spatialite-gui, si crea un nuovo database, e questo e' gia' inizializzato per essere un database spaziale.

I file delle liste in formato testo prodotti da Therion possono venire importati, ma vengono convertiti in tabelle senza attributi geometrici, cioè senza alcuna colonna che ne descriva la geometria. Perciò la lista delle grotte *cave-list* ha sì le coordinate X e Y ma non ha la geometria. Bisogna aggiungere una colonna "geometry" di tipo "POINT", associarla al sistema di riferimento appropriato (inserendo un record nella tabella "geometry_columns"), e riempirla con gli opportuni valori.

Un secondo problema, non indifferente, è che i nomi delle grotte, con sviluppo e profondità compaiono su una riga e le coordinate degli ingressi sulle righe successive. Per questi motivi ho trasformato il file della *cave-list* in uno script SQL, che

- crea la tabella, con anche la colonna per la geometria
- inserisce il record corrispondente nella tabella "geometry_columns"
- inserisce i dati delle grotte.

Con questo basta eseguire il file sql (si può fare anche con *spatialite-gui*),

```
spatialite> .read cave.sql ASCII
```

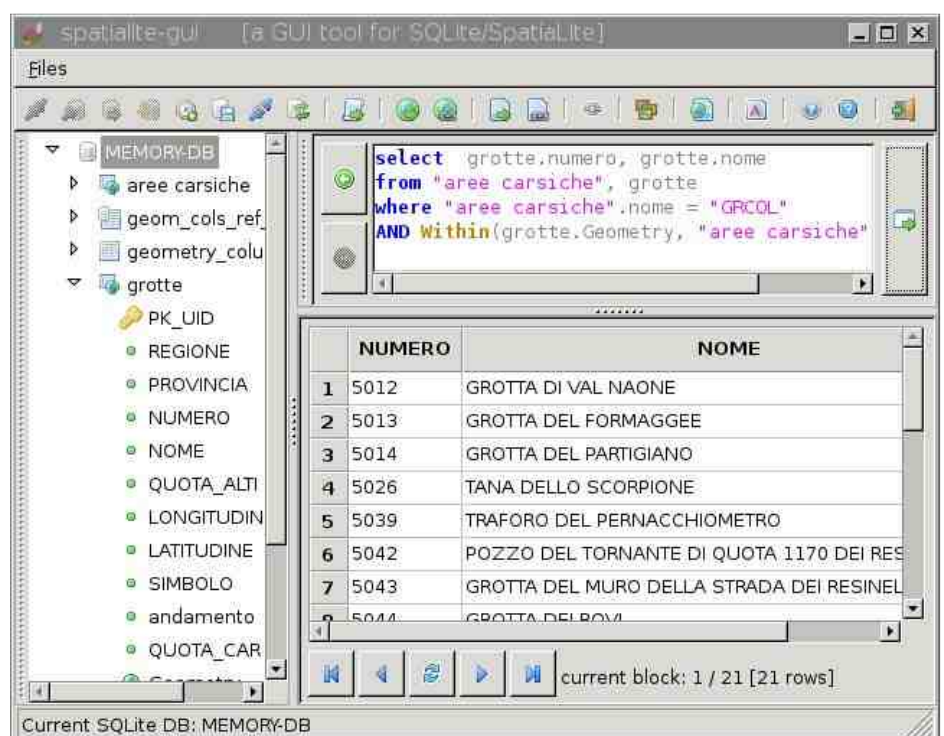
La *survey-list* generata da Therion ha un altro problema: una indentazione di due spazi davanti ai nomi delle grotte. Perciò *Spatialite* anziché usare il nome "Lo Lc 1501" usa " Lo Lc 1501". Sembra una cosa da poco, ma poi non si riesce a metterla in relazione con la tabella della *cave-list*. Bisogna rimuovere gli spazi iniziali, al fine di fare ricerche incrociate sulle due tabelle sulla base dei nomi delle grotte.

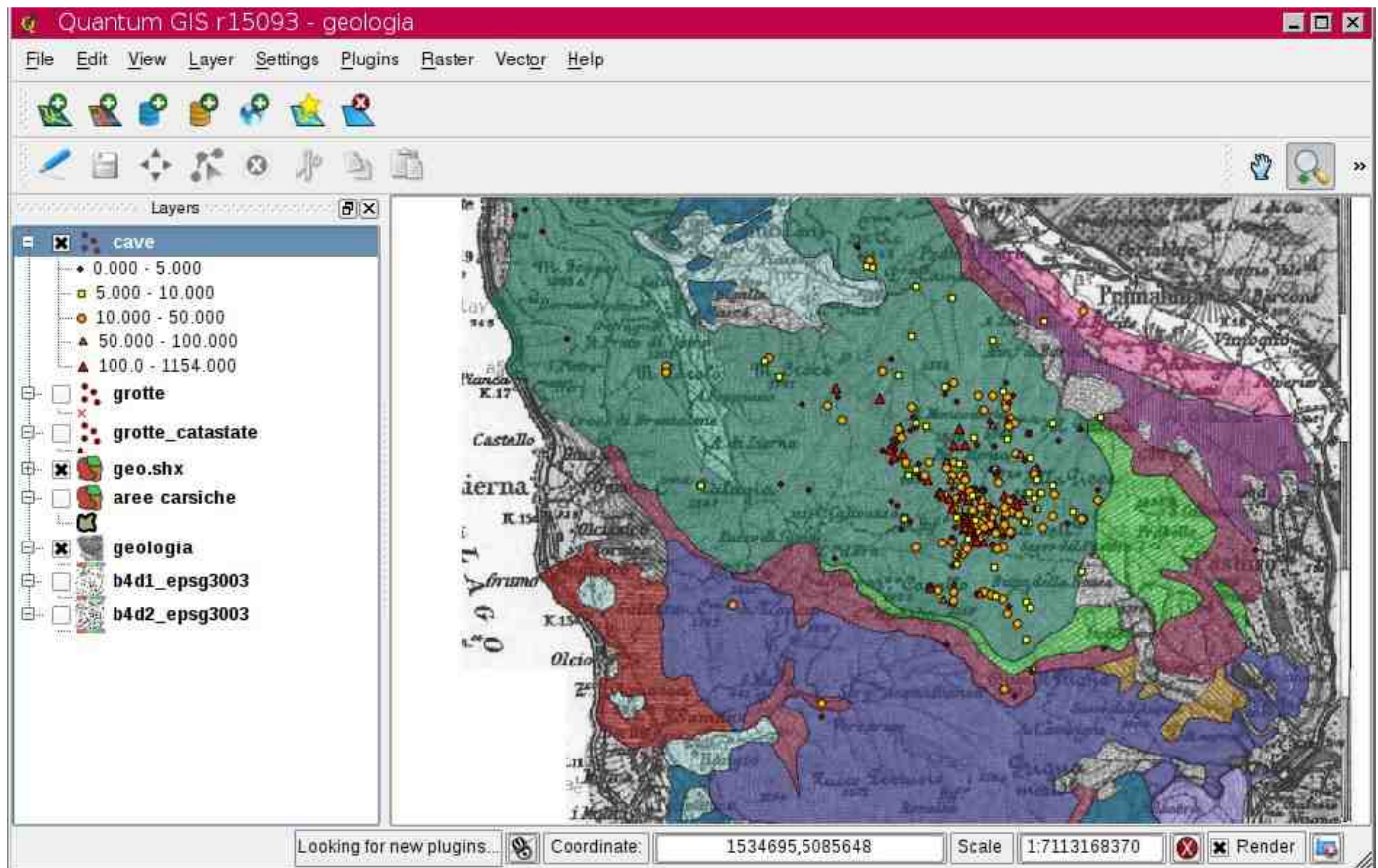
Con le informazioni di georeferenziazione e quelle dei survey si possono fare ricerche incrociate, per esempio costruire una tabella georeferenziata delle grotte con più di 100 tiri di rilievo,

```
CREATE TABLE survey100( PK_UID INTEGER NOT NULL AUTOINCREMENT, Title VARCHAR(80) NOT NULL, Shots INTEGER, geom POINT );  
INSERT INTO survey100(Title, Shots, geom) SELECT cave.title, survey.shots,  
AsText(cave.geom) FROM cave,survey WHERE cave.title = survey.title AND survey.shot >  
100;
```

Il database *Spatialite* può essere importato in *QGIS*: "Layer -> Add Spatialite layer", premere il bottone "New", selezionare il file del database, e premere il bottone "Add".

La cave-list caricata in *Spatialite* assieme al file "aree carsiche" del database del "carsismo". La figura mostra il risultato di una ricerca geometrica per le grotte in una data area carsica, utilizzando la funzione geometrica "Within".





I GIS facilitano la visualizzazione dei dati. La tabella del database SpatialLite e' importata in QGIS e sono visualizzate le grotte della Grigna catalogate per profondita' sullo sfondo della carta geologica. Le grotte sono rappresentate con colori e simboli diversi, a seconda della profondita'.

Export di database

Therion esporta i dati dei rilievi anche in formato "database", cioe' come script SQL. Questo script, caricato in un database genera diverse tabelle, tra cui la tabella dei capisaldi, "station", che contiene le coordinate X,Y,Z dei capisaldi nel sistema di riferimento specificato, e la tabella "shot" con i dati dei tiri dei rilievi.

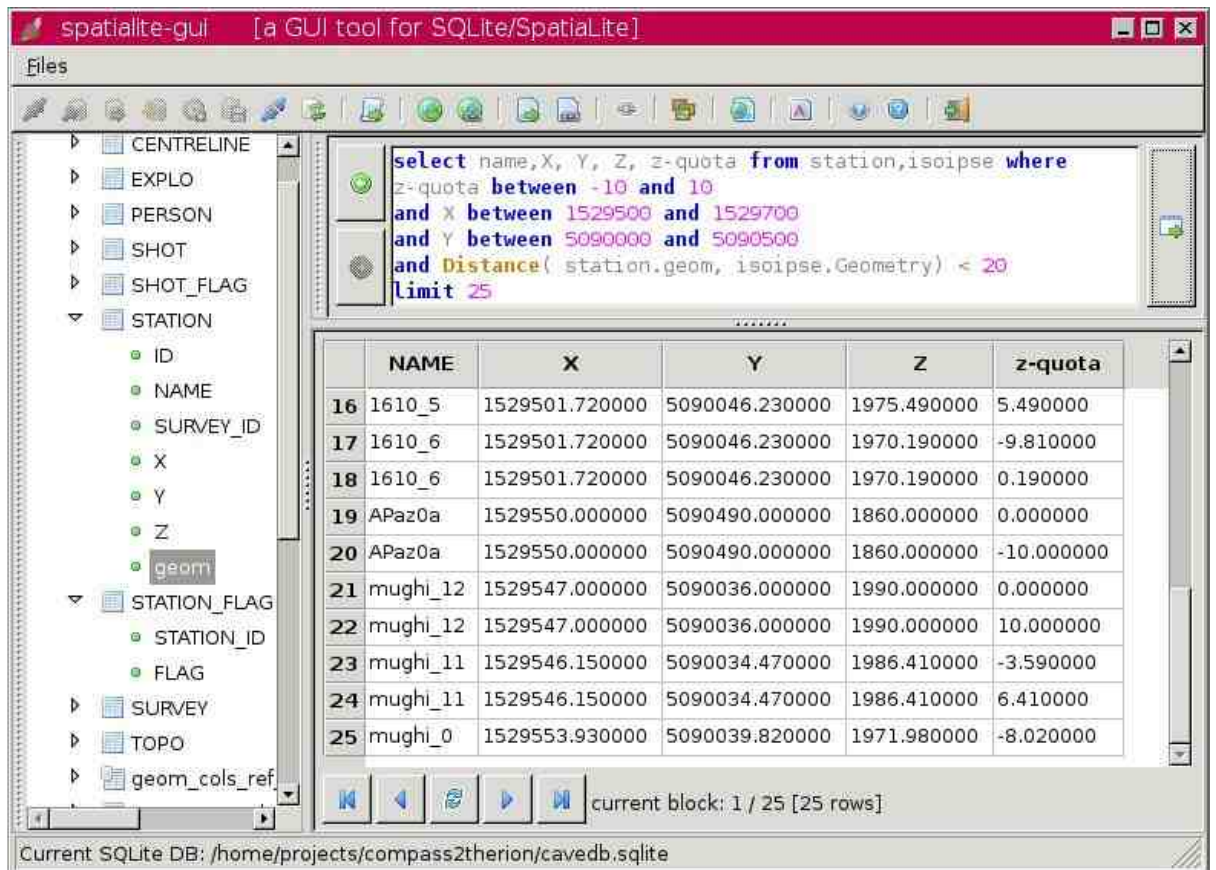
Non e' un database geografico, ma si puo' rendere tale aggiungendo una colonna per la geometria alla tabella "station", per specificare la posizione geografica dei capisaldi, e riempiendola con gli opportuni valori. Dopodiche' il database puo' essere usato in un GIS.

Esempio

In questo esempio si suppone di lavorare con SpatialLite usando un nuovo database gia' predisposto con le tabelle per gestire gli attributi geometrici. Dapprima viene aggiunta la colonna della geometria alla tabella "station", e si inserisce il record corrispondente nella tabella "geometry_columns". Quindi si esegue un comando SELECT che scrive i comandi per inserire i valori appropriati nella colonna della geometria. Questi comandi vengono scritti su un file che viene poi caricato per eseguirli.

```
.read cave.db ASCII
ALTER TABLE station ADD (geom POINT);
INSERT INTO geometry_columns VALUES ('station', 'geom', 'POINT', 2, 3003, 0);
.separator ' '
.output "/tmp/geo.sql"
select "UPDATE TABLE station SET geom=GeomFromText('POINT(", X, Y, ")') WHERE ID
=", id, ";" FROM station;
.read "/tmp/geo.sql"
```

A questo punto il database puo' esser caricato in un GIS (o in un database spaziale) assieme al DEM per fare della analisi 3D dei dati.



Il database generato da Therion e lo shapefile delle isoipse per cercare i punti vicini alla superficie dei rilievi, usando le funzioni geometriche di Spatialite: in questo caso la distanza (Distance) fra due elementi. Il risultato verra' poi usato per creare la tabella dei capisaldi vicini alla superficie.

Esempio di analisi 3D

Con il database esportato da Therion e il DEM delle isoipse caricati in Spatialite si possono trovare i capisaldi piu' vicini alla superficie di una data distanza (10 m, nell'esempio). Questi vengono salvati in una tabella con attributi (nome, X, Y, Z e la geometria).

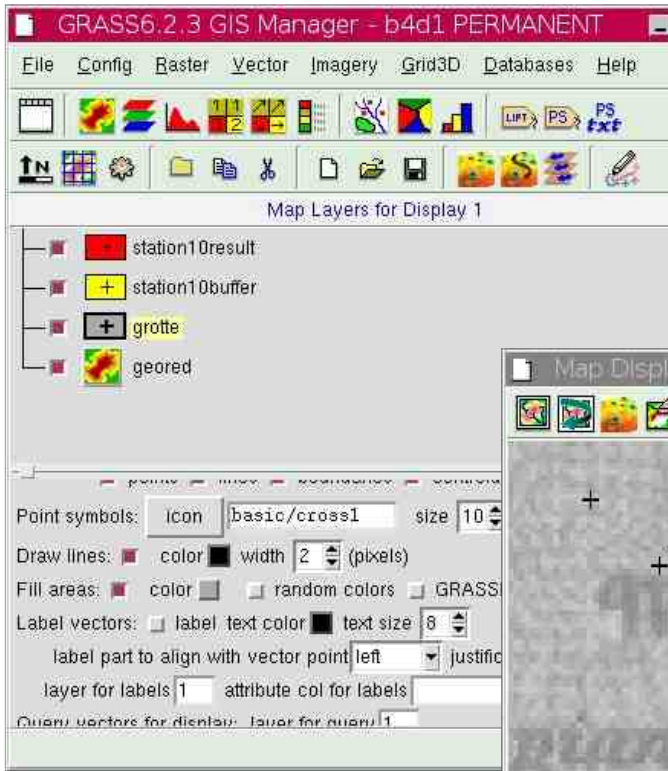
QGis si puo' connettere al database ed importare una tabella con geometria come layer vettoriale e convertirla in shapefile.

Lo shapefile caricato in Grass entro il database del "carsismo", viene elaborato. Dapprima si genera la mappa vettoriale buffer (20 m) intorno ai capisaldi. A questa si sottrae il buffer vettoriale (40 m) intorno agli ingressi delle grotte. Il risultato e' la mappa vettoriale delle zone con potenziali nuovi ingressi. (Nella figura viene presentata la zona del M.te Croce, Grigna, e non ce ne sono).

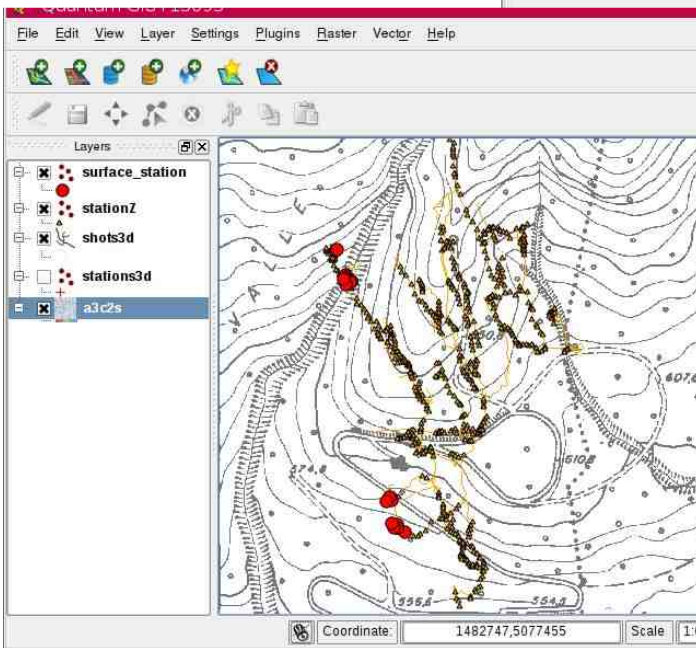
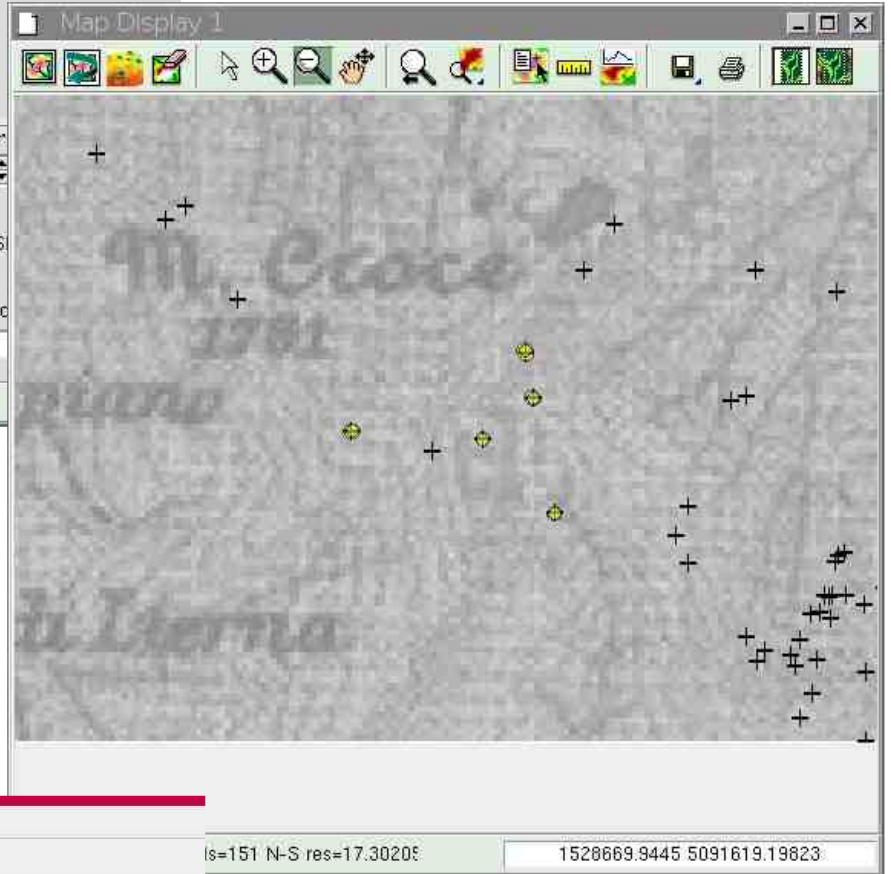
```

v.buffer input=station10 output=station10buffer type=point buffer=20
v.buffer input=grotte ouput=grottebuffer type=point buffer=40
v.overlay ainput=station10buffer atype=area binput=grottebugger btype=area
ouput=result operator=not

```

I buffer delle zone con un caposaldo a meno di 20 m (in giallo) e quello dove non c'è alcun ingresso (in rosso: nell'esempio questo buffer risulta vuoto). Le croci indicano le posizioni degli ingressi.



La maggior parte di queste elaborazioni possono essere eseguite efficientemente nel database e poi visualizzate attraverso il GIS. La figura a sinistra mostra i capisaldi a meno di 10 m dalla superficie per la grotta LoVa 2465 (Campo dei Fiori, Varese). In rosso i punti dove la grotta si avvicina alla superficie, compreso l'ingresso.

Links

Therion <http://therion.speleo.sk>

QGIS <http://www.qgis.org>

Spatialite <http://www.gaia-gis.it/spatialite>

Grass <http://grass.fbk.eu>