

Koko Suomen reititystä tukeva tieverkko maastotietokannasta Spatialite:lle

Jukka Rahkonen, <http://latuviitta.org>

Lisenssi [Creative Commons Attribution 3.0 Unported](https://creativecommons.org/licenses/by/3.0/)

Viimeksi muokattu 28. maaliskuuta 2014

Taustaa

Maanmittauslaitoksen maastotietokannan tiestö on laaja, avoin paikkatietoaineisto, joka käsittää yli 800 tuhatta kilometriä tiestöä. Erillisiä tieviivoja aineistossa on yli 3,2 miljoonaa kappaletta. Seuraavassa taulukossa on esitetty aineiston jakautuminen eri tieluokkiin (aineisto keväältä 2014).

Tieluokka	Tieluokan koodi	Lukumäärä kpl	Yhteispituus km
Autotie Ia	12111	6113	984
Autotie Ib	12112	20247	3190
Autotie IIa	12121	70837	10875
Autotie IIb	12122	180693	29617
Autotie IIIa	12131	474525	76409
Autotie IIIb	12132	246008	59223
Ajotie	12141	1472479	370709
Lautta	12151	437	70
Lossi	12152	77	10
Talvitie	12312	51173	30937
Kävely- ja pyörätie	12314	189213	29711
Ajopolku	12316	716840	204460
Yhteensä		3428642	816202

Maanmittauslaitos käyttää tieluokille seuraavaa jaottelua:

Tieluokka	Tieluokan selite
Autotie Ia	moottoritien kaksi- tai useampikaistainen ajorata
Autotie Ib	muun kaksiajorataisen kuin moottoritien kaksi- tai useampikaistainen ajorata tai yksiajoratainen, kaksi- tai useampikaistainen autotie, ajoradan leveys on yli 8 m
Autotie IIa	yksiajoratainen, kaksikaistainen, ajoradan leveys on 6,5-8 m
Autotie IIb	yksiajoratainen, kaksikaistainen, ajoradan leveys on 5-6,5 m

Autotie IIIa	yksiajoratainen, yksikaistainen, ajoradan leveys on 4-5 m
Autotie IIIb	yksiajoratainen, yksikaistainen, ajoradan leveys on 3-4 m
Ajotie	yksiajoratainen, yksikaistainen, ajoradan leveys on alle 3 m

Ajopolku

Ajopolku tallennetaan aina. Satunnainen, esimerkiksi puutavaran ajoon tilapäisesti käytetty traktoriura jätetään kuvaamatta. Hevosharjoitteluradat kuvataan ajopolkuina. Ajopolun vähimmäispituus on 100 m. Vähimmäispituutta lyhyempi ajopolku tallennetaan, jos se yhdistää kaksi paremman tieluokan tietä. Kapeahko ajopolun luonteinen ajoura tai käytöstä poistunut ajopolku voidaan esittää polkuna.

Kävely- ja pyörätie

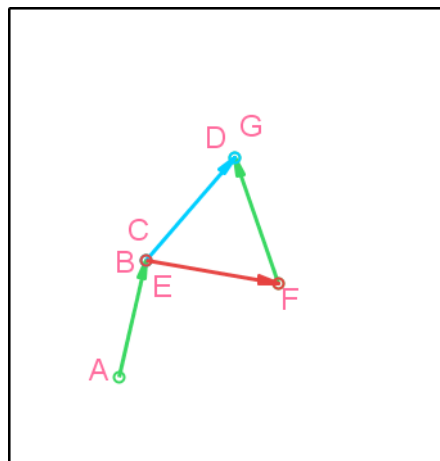
Kävely- ja pyöräteistä tallennetaan muusta liikenneverkosta erillään olevat väylät. Haja-asutusalueella tallennetaan myös kävely- ja pyörätiet, jotka oja, viheralue tms. erottaa autotiestä. Haja-asutusalueella kävely- ja pyörätien jatkuvuus pyritään säilyttämään pitkien tieosuuksien ollessa kyseessä, vaikka kävely- ja pyörätien erottaisi autotien ajoradasta paikoitellen vain tiemaalaus.

Aineiston tarkka kuvaus on saatavilla Maanmittauslaitoksen sivustolta

http://www.maanmittauslaitos.fi/sites/default/files/Maastotietokohteet_2013.pdf

Maastotietokannan tiestön valmistelu reititystä varten

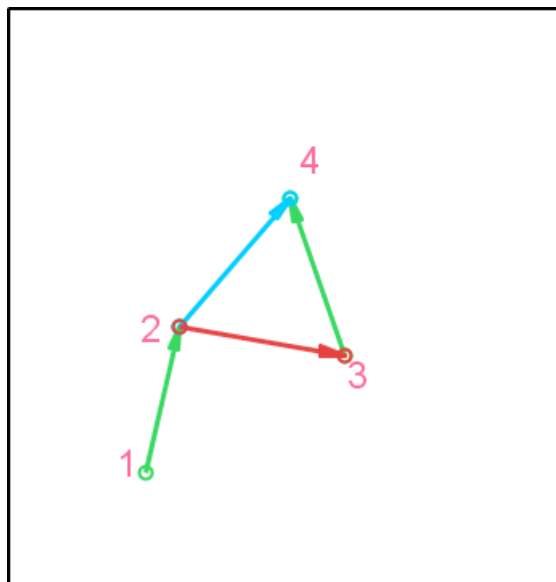
Reitityksen tarkoitus on antaa vastaus kysymykseen ”Kuinka pääsen lyhintä reittiä pisteestä A pisteeseen B”. Reitityksen kannalta maastotietokannan luovutukseen käytettävissä tiedostomuodoissa on se, että niissä jokainen viiva esiintyy omana erillisenä yksikkönään. Esimerkiksi alla olevassa kuvassa on viiva, joka alkaa pisteestä A ja loppuu pisteeseen B, sekä toinen viiva, joka alkaa pisteestä C ja loppuu pisteeseen D. Tähän aineistoon ei kuitenkaan sisälly tietoa siitä, että pisteet B ja C ovat samat pisteet, eivätkä reititysohjelmat siitä syystä ymmärrä, että pisteestä A on mahdollista päästä pidemmällekin kuin pisteeseen B, esimerkiksi pisteeseen D. Aineisto ei siis ole valmis reititykseen ilman pientä jatkojalostusta.



Kuva 1. Irrallisista viivoista koostuva paikkatietoaineisto.

Jotta reititys kävisi mahdolliseksi, on analysoitava kaikkien viivojen alkua- ja loppupisteet ja tutkittava, onko niiden kautta kulkuyhteys jollekin muulle viivalle. Kulkuyhteys on olemassa, jos päätepisteiden koordinaatit ovat samat. Analyysissä sulautetaan samassa paikassa olevat pisteet niin, että jälkelle jää minimimäärä pisteitä, ja näille pisteille annetaan uudet yksilölliset tunnukset. Esimerkiksi kuvassa 1 olevat pisteet B, C ja E sulautetaan yhteen, ja jäljelle jäävälle solmupisteelle annetaan tunnus "2". Pisteestä 1 pääsee nyt pisteeseen 4 kulkemalla pisteen 2 kautta. Kuvasta näkee myös, että pisteestä 1 voi mennä pisteeseen 4 paitsi lyhintä reittiä **1-2-4**, niin myös reittiä **1-2-3-4** pitkin. Tämä on hyvä tieto, sillä toisinaan lyhimmän reitin käyttö ei ole mahdollista. Jos esimerkiksi väli 2-4 on yksisuuntainen ja kulku suunnassa 4 → 2 on kielletty, niin lyhin sallittu reitti menosuuntaan on **1-2-4**, mutta paluumatkalla onkin käytettävä pidempää reittiä **4-3-2-1**. Jos väli 2-4 ei olekaan yksisuuntainen, vaan se on pyörätie, niin silloin lyhin sallittu reitti jalankulkijalla on 1-2-4, mutta autolle 1-2-3-4.

Pituuden ja yksisuuntaisuuden lisäksi edistyneet reititysmenetelmät voivat ottaa huomioon suurimman sallitun ajonopeuden, kääntymiskiellot, tien päällysteen laadun, teiden jyrkkyyden ym. tietoja.



Esivalmistelut: maastotietokannan tiestön lataaminen ja vienti Spatialite-tietokantaan

Tarvittavat ohjelmistot:

- Riittävän uusi GDAL. Windowsille suositellaan uusinta kehitysversiota GIS-internals:sta <http://www.gisinternals.com/sdk/>
- Spatialite-gui versio 1.7.1 tai uudempi <http://www.gaia-gis.it/gaia-sins/>
- Spatialite_network 4.1.1 tai uudempi <http://www.gaia-gis.it/gaia-sins/>
- Spatialite 4.1.1 komentoriviohjelma <http://www.gaia-gis.it/gaia-sins/>

Reititystulosten näyttämiseen kartalla voidaan käyttää OpenJUMP-ohjelmaa

- OpenJUMP Plus kehitysversio http://sourceforge.net/projects/jump-pilot/files/OpenJUMP_snapshots/
- OpenJUMP:in päivittämiseksi uusin SQLite-ajuri, versio 3.8.2-SNAPSHOT <https://bitbucket.org/xerial/sqlite-jdbc/downloads>
- SQLite-tietokannan Spatialite-laajennus, versio 4.2 <http://www.gaia-gis.it/gaia-sins/spatialite-4.2.0-test-win-x86.7z> (Windows 32-bittinen) tai <http://www.gaia-gis.it/gaia-sins/spatialite-4.2.0-test-win-amd64.7z> (Windows 64-bittinen)

OpenJUMP:in ja Spatialiten käyttö saattaa olla mahdollista myös Linuxilla ja Mac:llä.

Aineiston lataus

Maastotietokannan tiestö on ladattavissa Maanmittauslaitoksen aineistopalvelusta <https://tiedostopalvelu.maanmittauslaitos.fi/tp/kartta>

Onneksi aineiston voi hakea hyvin paljon vähemmällä vaivalla Kapsi ry:n ylläpitämästä palvelusta käyttämällä rsync-ohjelmistoa. Palvelun osoite on <http://kartat.kapsi.fi/>. Rsync-ohjelman asentamista ja käyttöä ei käsitellä tässä ohjeessa, mutta se selviää aikaisemmasta ohjeesta http://latuviitta.org/documents/MML-taustakarttasarja_WMS-palveluksi.pdf

Aineistoon kuuluu 115 erillistä zip-tiedostoa, joiden yhteenlaskettu koko on 548 megatavua.

Aineiston vieminen Spatialite-tietokantaan

Kun kaikki 115 zip-tiedostoa on saatu ladatuksi, niin niistä muodostetaan Spatialite-tietokanta GDAL-ohjelmalla ”ogr2ogr”. Zip-tiedostoja ei tarvitse purkaa, koska GDAL pystyy lukemaan tiedot suoraan arkiston sisältä. Ensimmäisellä ogr2ogr-komennolla luodaan Spatialite-tietokanta. Seuraavilla komennoilla lisätään jo luotuun tietokantatauluun lisää tietoja toisista zip-tiedostoista. Tästä syystä ensimmäinen tarvittava ogr2ogr-komento on siis hieman erilainen kuin 114 seuraavaa, joissa ainoa muutettava osa kulloinkin käsiteltävän zip-tiedoston nimi.

```
ogr -f sqlite -dsco spatialite=yes gr2mtk_tos.sqlite -dim 2
/vsizip/e:\mtk_tos\etrs89\gml\K32.zip tieviiva -lco spatial_index=no -gt 65536

ogr2ogr -f sqlite -append mtk_tos.sqlite -dim 2
/vsizip/e:\mtk_tos\etrs89\gml\K34.zip tieviiva -gt 65536

ogr2ogr -f sqlite -append mtk_tos.sqlite -dim 2
/vsizip/e:\mtk_tos\etrs89\gml\K42.zip tieviiva -gt 65536

--tämä komento toistuu jokaiselle zip-tiedostolle--

ogr2ogr mtk_tos.sqlite -sql "SELECT CreateSpatialIndex('tieviiva','geometry')"
```

Huom. Latauksen nopeuttamiseksi aineistot kannattaa ladata tietokantaan ensin ilman spatiaali-indeksiä, ja luoda indeksi vasta kaikkien tieviivojen lataamisen jälkeen.

Alku- ja loppupisteiden sulautus ja tietojen lisääminen tiestöaineistoon

Jotta maastotietokannan tiestö saadaan jalostetuksi kuvan 1. mukaisesta alkutilanteesta kuvan 2. mukaiseen lopputilanteeseen, niin on tehtävä seuraavat toimenpiteet:

1. Valitaan kaikkien tieviivojen alkupisteet ja tallennetaan ne tilapäiseen paikkaan.
2. Valitaan kaikkien tieviivojen loppupisteet ja tallennetaan ne tilapäiseen paikkaan.
3. Etsitään tilapäisestä paikasta kaikki sellaiset pisteet, joilla on täsmälleen samat koordinaatit ja sulautetaan ne yhteen, niin että jäljelle jää vain yksi.
4. Numeroidaan tilapäiseen paikkaan jäljelle jääneet solmupisteet niin, että ne saavat yksilölliset tunnukset.
5. Seuraavien vaiheiden nopeuttamiseksi luodaan solmupistetaululle spatiaali-indeksi.
6. Lisätään tieaineistoon kaksi uutta ominaisuustietoa: viivan alkava solmupiste ja viivan päättävä solmupiste.
7. Kopioidaan viivat, joiden alku- ja loppupiste on sama, uuteen tauluun.
8. Kopioinnin jälkeen poistetaan nämä viivat alkuperäisestä tieaineistosta.
9. Kirjoitetaan uusiin kenttiin tilapäisestä taulusta löytyvät yksilölliset tunnisteet. Tämä tapahtuu valitsemalla viiva kerrallaan viivan alkupiste ja etsimällä tilapäisestä taulusta rivi, jolla on samat koordinaatit. Tältä riviltä löytyvä tunniste on viivan alkupisteen yksilöivä tunnus. Sama haku tehdään aineiston kaikkien viivojen alku- ja loppupisteille. Kohdassa 5 tehty spatiaali-indeksi on tässä vaiheessa välttämätön, sillä muussa tapauksessa tietokannassa pitäisi verrata jokaista alku- ja loppupistettä jokaiseen sulautettuun solmupisteeseen ja kokeilla, löytyykö samoja koordinaatteja. Tiestöaineistolla vertailujen määräksi tulisi peräti **6,8 miljoonaa x 3,0 miljoonaa=20,4 biljoonaa**. Spatiaali-indeksin ansiosta vertailujen lukumääräksi jää lähelle alku- ja loppupisteiden yhteenlaskettua lukumäärää eli noin 7 miljoonaa.

Käytännössä vaiheet 1-4 voidaan tehdä yhdellä SQL-komennolla. Vaiheissa 7. ja 8. toteutetaan teknisistä syistä tarvittava siivous. Tiestöaineistoon sisältyy nimittäin jonkin verran silmukoita, ja alku- ja loppupisteiden sulauttamisen jälkeen näillä silmukoilla on sama alku- ja lähtöpiste. Tilanne vastaa todellisuutta, koska silmukkaa pitkin pääsee ainoastaan pisteestä A pisteeseen A, mutta Spatialiten reititysohjelman mielestä tämä tilanne on virhe. Tästä syystä silmukkamaiset tiet kopiodaan ensin omaan tietokantatauluunsa, minkä jälkeen ne poistetaan alkuperäisestä tieaineistosta.

Nämä kaikki vaiheet voidaan suorittaa automaattisesti peräkkäin kirjoittamalla tarvittavat SQL-komennot komentotiedostoon. Komentotiedosto voidaan suorittaa Spatialite-gui -ohjelmasta, mutta parempi vaihtoehto on kuitenkin suorittaa se Spatialite-komentoikkunasta. Komentoikkunaan nimittäin tulostuu hyödyllistä tietoa mahdollisista virheistä, kun taas Spatialite-gui ilmoittaa virhetilanteesta vain suorituksen epäonnistuneen.

Liitteessä 1 on valmis komentojono joka suorittaa kaikki vaiheet 1-9. Komentocono tallennetaan tekstitiedostoksi esimerkiksi nimellä ”mtk_reititys.sql”, jonka jälkeen sen käyttäminen Spatialiten komentoriviltä tapahtuu seuraavalla tavalla:

Vaihe 1. Käynnistetään komentoriviohjelma ja avataan tietokanta ”mtk_tos.sqlite”. Tässä tapauksessa ohjelma ”spatialite.exe” on hakemistossa ”c:\ohjelmat\spatialite” ja tietokanta sekä komentotiedosto ovat työhakemistossa ”c:\mtk_tos”.

```
C:\mtk_tos>c:\ohjelmat\spatialite\spatialite mtk_tos.sqlite
```

Komento listaa näytölle tietoja ohjelmistosta ja avaa komentorivin, jolla voidaan antaa SQLite-

tietokannan ymmärtämiä komentoja.

Vaihe 2. Annetaan Spatialite-komentoikkunassa käsky lukea tekstitiedosto, johon SQL-komennot on tallennettu, ja suorittaa ne. Käsky annetaan muodossa: **”.read tiedoston.nimi tiedoston_koodisto”**. Windowsin suomalaisilla asetuksilla koodistona on ”cp1252”. Alla oleva esimerkki näyttää, miltä tässä tilanteessa näkyy tietokoneen ruudulla.

```
C:\mtk_tos>c:\ohjelmat\spatialite\spatialite mtk_tos.sqlite
Spatialite version ..: 4.1.0      Supported Extensions:
- 'VirtualShape'      [direct Shapefile access]
- 'VirtualDbf'       [direct DBF access]
- 'VirtualXL'        [direct XLS access]
- 'VirtualText'      [direct CSV/TXT access]
- 'VirtualNetwork'   [Dijkstra shortest path]
- 'RTree'            [Spatial Index - R*Tree]
- 'MbrCache'         [Spatial Index - MBR cache]
- 'VirtualSpatialIndex' [R*Tree metahandler]
- 'VirtualXPath'     [XML Path Language - XPath]
- 'VirtualFDO'       [FDO-OGR interoperability]
- 'Spatialite'       [Spatial SQL - OGC]

PROJ.4 version .....: Rel. 4.8.0, 6 March 2012
GEOS version .....: 3.4.0dev-CAPI-1.8.0 r3800
LWGEOM version .....: 2.0.3
SQLite version .....: 3.7.17
Enter ".help" for instructions
SQLite version 3.7.17 2013-05-20 00:56:22
Enter ".help" for instructions
Enter SQL statements terminated with a ";"

spatialite> .read mtk_reititys.sql cp1252
```

Valmiiseen komentotiedostoon on kirjoitettu muutamia SQL.-komentoja, jotka tulostavat ruudulle kellonaikoja ja väliaikatietoja komentotiedoston etenemisestä. Mallitulostus eräältä keväiseltä sunnuntailta näyttää tältä:

```
2014-03-09 09:55:41 alku
2014-03-09 09:59:58 nodes tehty
2014-03-09 09:59:58 lisatty node_from ja node_to
1
2014-03-09 10:00:01 geometriakentta tehty
1
2014-03-09 10:05:23 spatiaali-indeksi tehty
2014-03-09 10:31:43 node_from taytetty
2014-03-09 10:59:09 node_to taytetty
2014-03-09 10:59:39 silmukat kopioitu
2014-03-09 11:01:09 silmukat poistettu
2014-03-09 11:01:09 valmis
```

Spatialiten reititystaulun luominen spatialite_network-ohjelmalla

Maastotietokannan tiestö on nyt muokatta sellaiseen kuntoon, että siitä voidaan rakentaa Spatialiten avulla reititykseen soveltuva verkosto eli graafi. Myös tämä vaihe on mahdollista tehdä Spatialite-gui-ohjelmalla, mutta tässäkin tapauksessa se on parempi tehdä komentorivityökalulla ”spatialite_network”, joka tulostaa enemmän tietoa mahdollisista ongelmista. Tai sitten voi ensin kokeilla onnistuuko verkoston luominen helpolla Spatialite-gui-menetelmällä ja käyttää komentorivityökalua vasta, jos helppo menetelmä ei toimi.

Komento sekä onnistuneesta suorituksesta syntyvä tulostus kuvaruudulla näyttävät tältä:

```
C:\mtk_tos>c:\ohjelmat\spatialite\spatialite_network -d mtk_tos.sqlite
-T tieviiva -f node_from -t node_to -g geometry -n nimi_suomi
-o tieviiva_net_data

SQLite version: 3.7.17
Spatialite version: 4.1.0
Step I - checking for table and columns existence

spatialite-network
=====
Spatialite db: mtk_tos.sqlite
validating table: tieviiva

columns layout
=====
FromNode: node_from
  ToNode: node_to
    Cost: GLength(geometry)
    Name: nimi_suomi
Geometry: geometry

assuming arcs to be BIDIRECTIONAL

NETWORK-DATA table creation required: 'tieviiva_net_data'
Overwrite not allowed if table already exists
=====

Step II - checking value types consistency
Step III - checking topologic consistency
Step IV - final evaluation

Statistics
=====
# Arcs : 6857284
# Nodes: 3037794
Node max incoming arcs: 8
Node max outcoming arcs: 8
# Nodes cardinality=1: 1028240 [terminal nodes]
# Nodes cardinality=2: 381765 [meaningless, pass-through]
=====

OK: network passed validation
    you can apply this configuration to build a valid VirtualNetwork
OK: validation passed

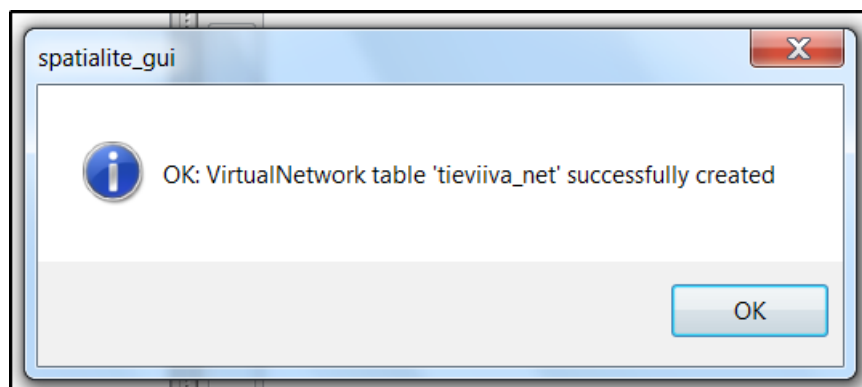
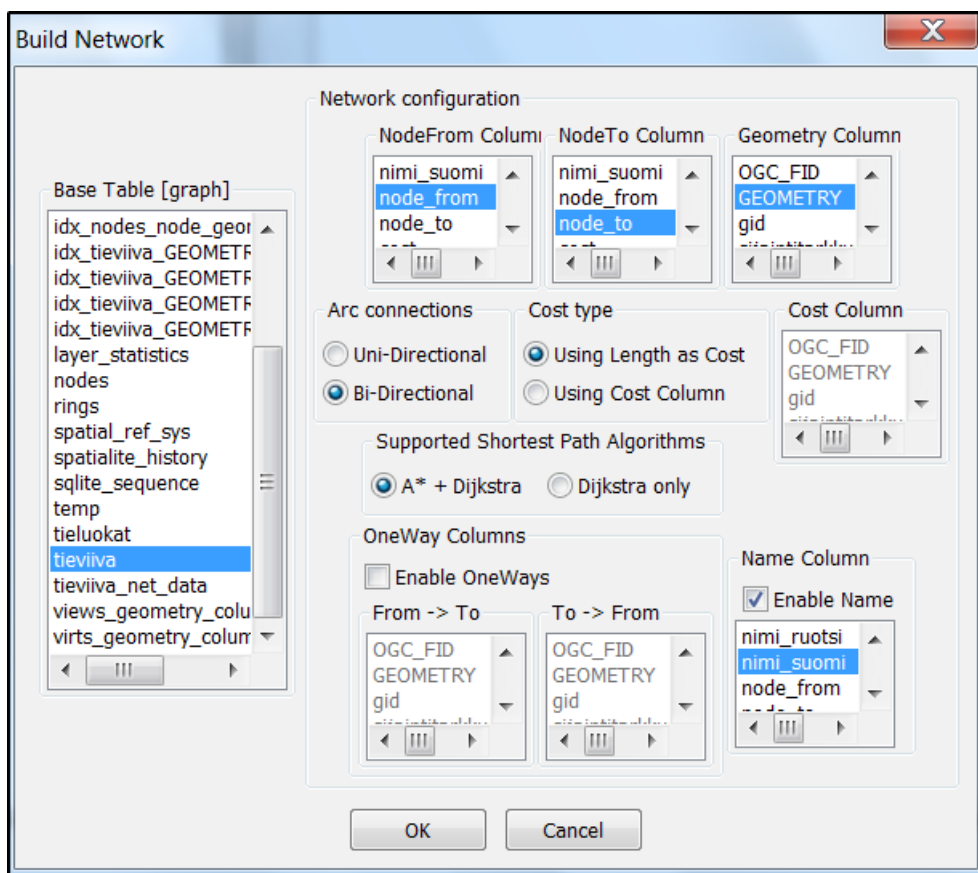
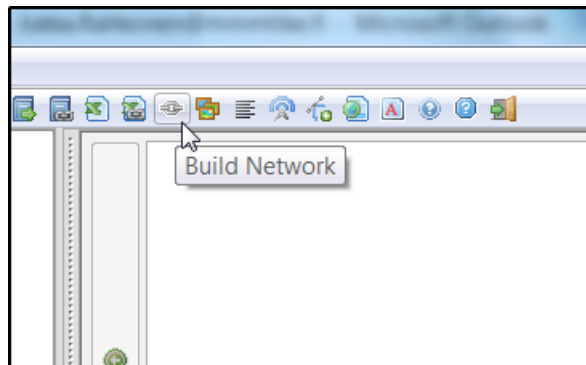
OK: NETWORK-DATA table 'tieviiva_net_data' successfully created
OK: table 'tieviiva_net_data' successfully created
```

Nyt ollaan jo melkein valmiita. Ihan viimeinen työvaihe on luoda SQL-komennolla virtuaalitaulu, jonka avulla reititystä lopulta käytetään. SQL-komento voidaan antaa Spatialite-gui-ohjelman SQL-ikkunasta tai Spatialite-komentoikkunasta.

```
CREATE VIRTUAL TABLE "tieviiva_net" USING VirtualNetwork("tieviiva_net_data");
```

Spatialiten reititystaulun luominen spatialite_gui-ohjelmalla

Spatialite-gui-ohjelmassa on työkalu, jonka avulla reititysverkon tekeminen on hyvin yksinkertaista. Seuraavat kolme kuvaa puhukoot puolestaan.



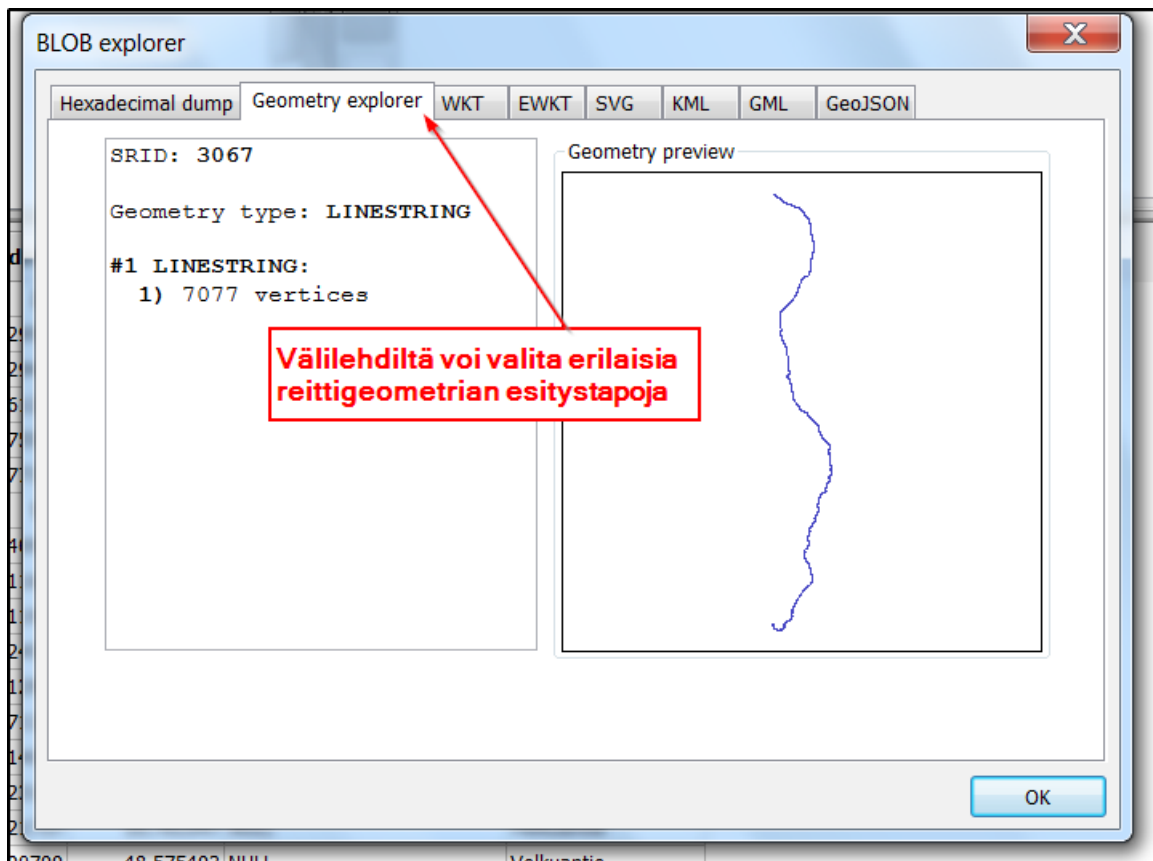
Reitityksen käyttö Spatialite-gui-ohjelmalla

Reititystä tukeva maastotietokannan tiestö on nyt valmis. Tarjolla on 816 tuhatta kilometriä tiestöä moottoriteistä metsäautoteihin asti. Kaikki sekä ilmaista että vapaata, kunhan muistaa mainita Maanmittauslaitoksen lisenssin ehtojen mukaan tiestöaineiston alkuperän. Ei käyttömaksuja, ei tarvetta nettiyhteydelle, ei piilotettua pakkoa jakaa omia paikkatietojaan kenellekään. Kuulostaa niin hyvältä, että olisi kiva, jos tätä reititysominaisuutta pystyisi vielä jotenkin käyttämäänkin. Aloitetaan tutustuminen jälleen Spatialite-gui-ohjelmalla. Ohjelmalla ei oikeastaan pysty tekemään mitään sellaista, joka auttaisi suoraan reitin valitsemisessa, mutta se havainnollistaa hyvin, mitä tapahtuu pellin alla. Reitityksessä on pohjimmiltaan kyse vain siitä, että koneelle annetaan lähtöpiste ja määränpää, ja kone palauttaa omasta mielestään parhaan reitin näiden pisteiden välille.

The screenshot shows the Spatialite-gui application interface. The top part displays a file explorer with the database file 'E:\mtk_tos\mtk_tos.sqlite' and a tree view of the database structure. The main window shows a SQL query: `select * from tieviiva_net where nodefrom=101 and nodeto=201;`. Below the query, a table of results is displayed. The first row is highlighted, showing a cost of 169062.562427. Red annotations explain that the first row's cost is 169 km and that the Geometry column contains the route's geometry. A mouse cursor is shown clicking on the Geometry column header to view more details.

	Algorithm	ArcRowid	NodeFrom	NodeTo	Cost	Geometry	Name
1	Dijkstra	NULL	101	201	169062.562427	BLOB sz=113280 GEOMETRY	NULL
2	Dijkstra	86697	101	1629595	88.924910	NULL	Pohjanpääntie
3	Dijkstra	86692	1629595	3029432	343.960408	NULL	Pohjanpääntie
4	Dijkstra	86701	3029432	561282	35.731613	NULL	Pohjanpääntie
5	Dijkstra	86693	561282	2075599	215.871805	NULL	Velkuantie
6	Dijkstra	86704	2075599	2477318	215.477214	NULL	Velkuantie
7	Dijkstra	86706	2477318	541	37.472890	NULL	Velkuantie
8	Dijkstra	261282	541	2240928	24.334880	NULL	Velkuantie
9	Dijkstra	261287	2240928	2511623	33.120390	NULL	Velkuantie
10	Dijkstra	261284	2511623	1111913	75.295816	NULL	Velkuantie
11	Dijkstra	261283	1111913	1224574	121.638048	NULL	Velkuantie
12	Dijkstra	261279	1224574	2112996	461.502208	NULL	Velkuantie
13	Dijkstra	261294	2112996	771388	103.969137	NULL	Velkuantie
14	Dijkstra	261314	771388	114077	68.356630	NULL	Velkuantie

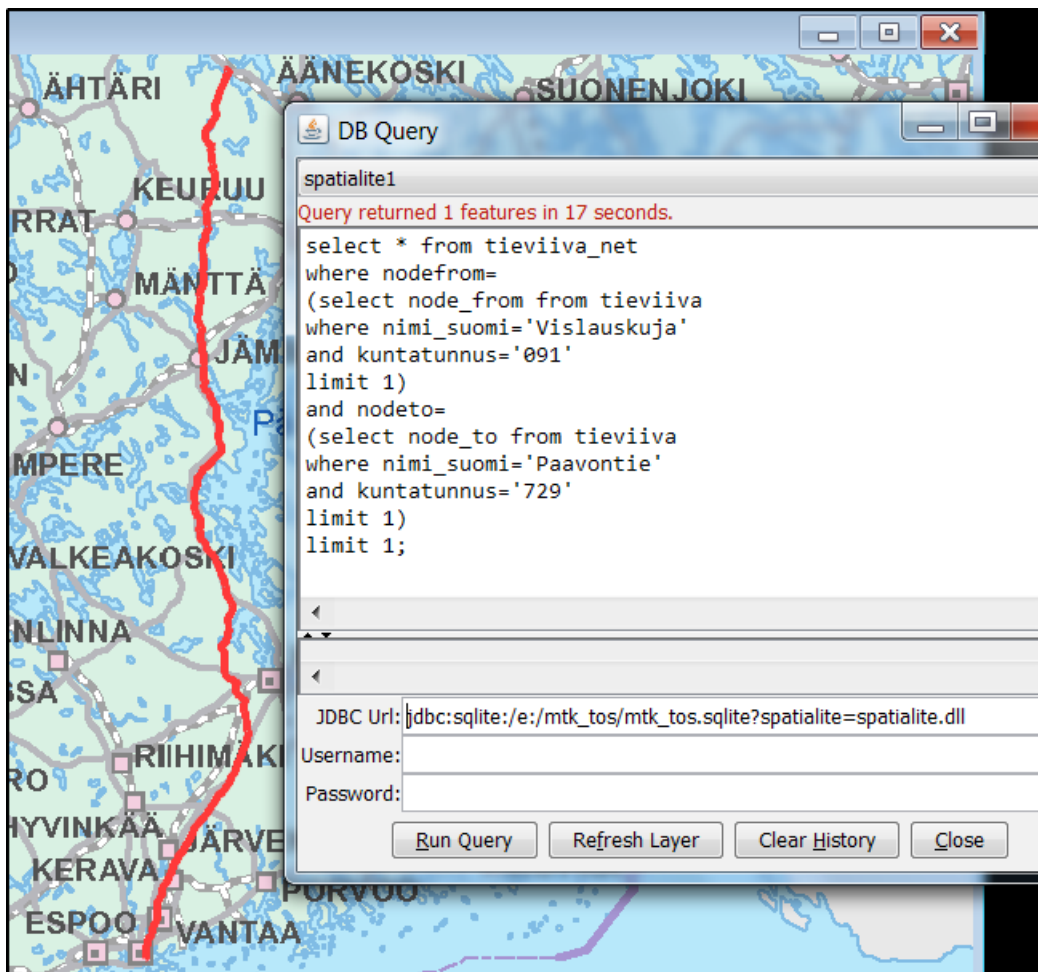
Reittigeometriaa voidaan tarkastella BLOB Explorer -työkalulla. KML-välilehdeltä reitti on mahdollista kopioida ja tallentaa tiedostoon, jonka jälkeen se voidaan näyttää esimerkiksi Google Earth:ssa, mutta muuta suoraa hyötykäyttöä tämän ohjelman kautta on vaikea kuvitella. Mielikuvitusta käyttämällä voi kuitenkin kuvitella, mitä joku ohjelmoija voisi tämän reititysmoottorin ympärille rakentaa.

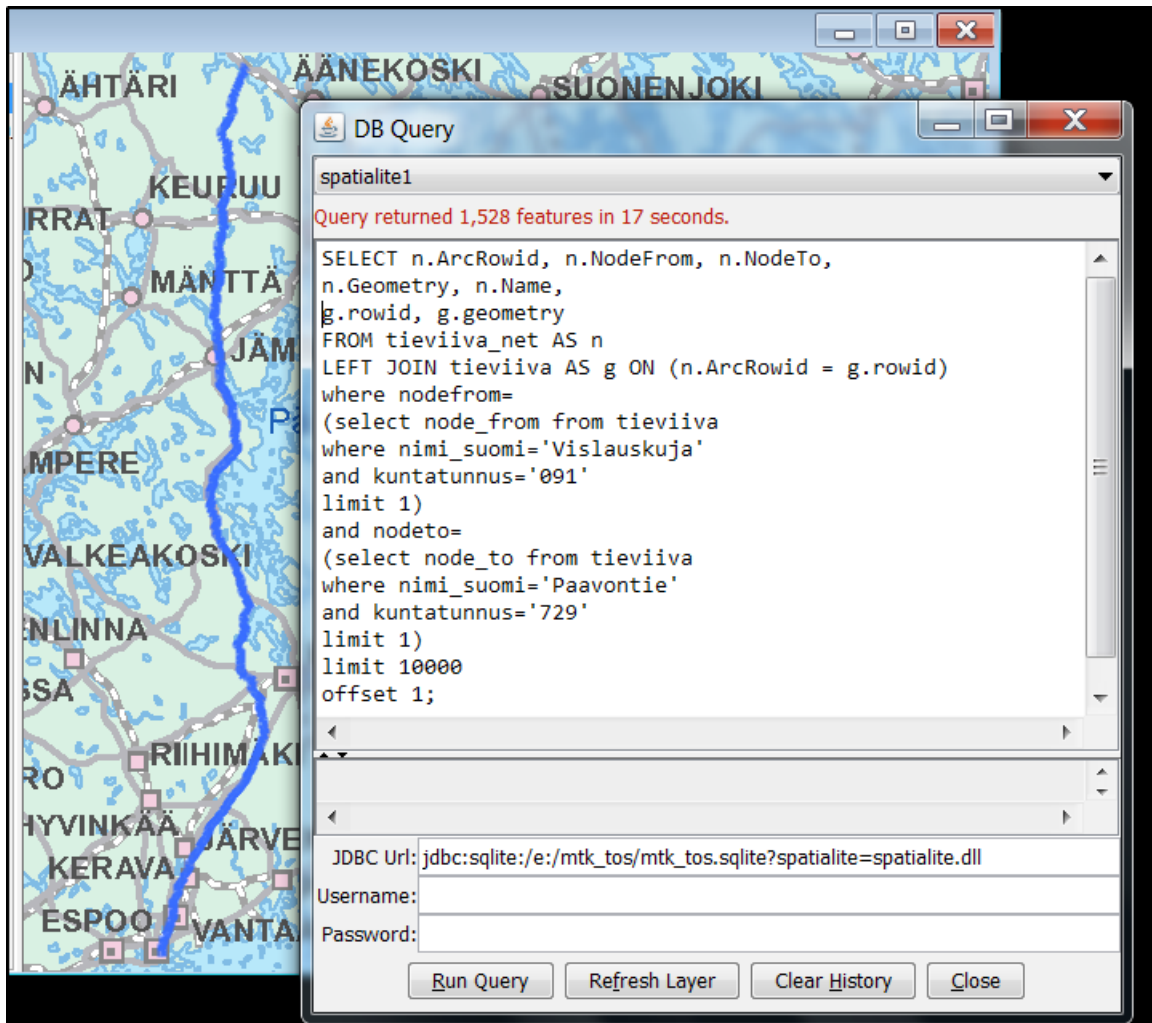


Reitityskyselyn lähtö- ja loppupisteet on annettava pisteen tunnisteiden avulla, mitä kukaan käyttäjä ei tietenkään käytännössä voi tehdä. Maastotietokannan tiestössä on kuitenkin onneksi hienoja ominaisuustietoja, joiden avulla kyselyiden teko helpottuu kovasti. Aineistossa on nimittäin mukana sekä teiden nimet että teiden sijaintikunnat. Katsotaanpa, kuinka voidaan etsiä reitti Helsingin Vislauskujalta Saarijärven Paavontielle. Käyttäjän on tiedettävä, että Helsingin kuntakoodi on '091' ja Saarijärven '729' mutta muuten kyselyn tekeminen on varsin helppoa. Unohdetaan suurimmat tarkkuusvaatimukset ja oikaistaan sen verran, että valitaan lähtöpisteeksi ensimmäisen tietokannasta löytyvät Vislauskuja-nimisen tienpätkän alkupiste ja määränpääksi vastaavasti ensimmäisen Saarijärveltä löytyvän Paavontien pätkän loppupiste. Vislauskujan pätkiä on muuten tietokannassa 9 ja Paavontien pätkiä 30 kappaletta.

```
select * from tieviiva_net
where nodefrom=
(select node_from from tieviiva
where nimi_suomi='Vislauskuja'
and kuntatunnus='091'
limit 1)
and nodeto=
(select node_to from tieviiva
where nimi_suomi='Paavontie'
and kuntatunnus='729'
limit 1);
```

Reitityksen käyttö OpenJUMP-ohjelmalla





tomize Tools Raster Plugins Sextante Window Help

200 km

Attributes: Project 1:SELECT n.ArcRowid, n.NodeFrom, n.NodeTo, n.Geometry, n.Name, g....

SELECT n.ArcRowid, n.NodeFrom, n.NodeTo, n.Geometry, n.Name, g....

Kaikki reitin osuudet on valittu omiksi viivokseen ja niille voidaan poimia tietokannasta halutut ominaisuudet.

Valitun rivin sijainti kartalla

FID	ArcRowid	NodeFrom	NodeTo	Geometry	Name
2777	1629737	28	10		
2778	1629738	28	10		
2779	1629744	10	24		
2780	1629740	24	29		
2781	1629742	29	17		
2782	1629739	17	6		
2783	1629735	1967115	2948760		Kukkarontie
2784	1629732	2948760	2220706		Kukkarontie
2785	1629712	33869	1892271		Kukkarontie
2786	1629705	2953273	1610956		Kukkarontie
2787	1629712	33869	2953273		Kukkarontie
2788	1629705	2953273	1610956		Kukkarontie
2789	1629709	1610956	400611		Kukkarontie
2790	1629711	400611	2900944		Kukkarontie
2791	1629708	2900944	2480467		Kukkarontie
2792	1629656	2480467	2024788		Kukkarontie
2793	1629663	2024788	10144		Kukkarontie
2794	1759691	10144	2576745		Kukkarontie
2795	1759674	2576745	2048330		Kukkarontie
2796	1759676	2048330	2694102		Kelantie
2797	1759668	2694102	1817895		Kelantie
2798	1759665	1817895	142837		Kelantie
2799	1759658	142837	1312687		Kelantie
2800	1759653	1312687	877366		Kelantie
2801	1738644	877366	3000064		Kelantie
2802	1738635	3000064	2939650		Kelantie
2803	1738614	2939650	1422472		Kelantie
2804	1738629	1422472	2233879		Kelantie
2805	1738632	2233879	429495		Kelantie
2806	1738621	429495	573401		Kelantie
2807	1738624	573401	1846798		Kelantie



Reititys autoteitä pitkin

Edellä luotu reititysverkko on erittäin kattava ja mielenkiintoinen, mutta ei kovin käyttökelpoinen, koska se ohjaa autot pyöriteille ja kävelijät moottoriteille, jos sellaisia lyhimmälle reitille sattuu. Mutta kuinka reititys saataisiin käyttämään vain tiettyjä valittuja tieluokkia? Tähän on olemassa muutama eri vaihtoehto.

Vaihtoehto 1: Poista kielletyt luokat aineistosta

Suoraviivainen ratkaisu olisi poistaa esimerkiksi autoja varten tehtävää reititystä varten kaikki ei-autoiltavat luokat 12141, 12312, 12313, 12314 ja 12316 kokonaan aineistosta. Tämä toimii varmasti ja lisäksi tietokannan koko pienenee. Ratkaisu voisi olla hyvä pelkästään autokäyttöön tarkoitettulle navigaattorille. Huonona puolena on se, että mahdollisuus jalankulkureititykseen katoaa. Tauluista voitaisiin myös kopioida tietyt tieluokat omiin tauluihinsa, mutta se puolestaan suurentaa tietokannan kokoa.

Vaihtoehto 2: Annetaan suuri kustannusarvo tieluokille, joita ei haluta käyttää

Reitin valintaa voidaan ohjailia myös kustannussarakkeen avulla. Spatialiten reitityksessä on oletuksena käyttää kustannuksena tien pituutta, mutta sitä voidaan myös pyytää lukemaan kunkin tieosuuden kustannus tätä varten tehdystä tietokantataulun kustannussarekkeesta. Ei-toivotulle tieluokalle voidaan antaa hyvin suuri kustannus, jolloin reititys suosii luonnollisesti muita vaihtoehtoja. Kustannussarekkeeseen voitaisiin ensiksi tallentaa kaikkien teiden pituudet, ja sitten päivittää ei-toivottujen luokkien kustannukseksi esimerkiksi 99999 kilometriä.

Tämä vaihtoehto ei kuitenkaan toimi käytännössä Spatialitellä ja maastotietokannan tiestöllä.

Korkean kustannuksen tiet ovat ilmeisesti edelleen reitin laskennassa mukana, ja niiden takia laskenta käy joissain tapauksissa tavattoman hitaaksi. Spatialite ei myöskään tue PostGIS-tietokannan pgRouting-reititysohjelman käyttämää tapaa, jossa tien käyttö reititykseen voidaan estää antamalla sen kustannukseksi negatiivinen luku.

Vaihtoehto 3: Annetaan teille kielletty ajosuunta molempiin suuntiin

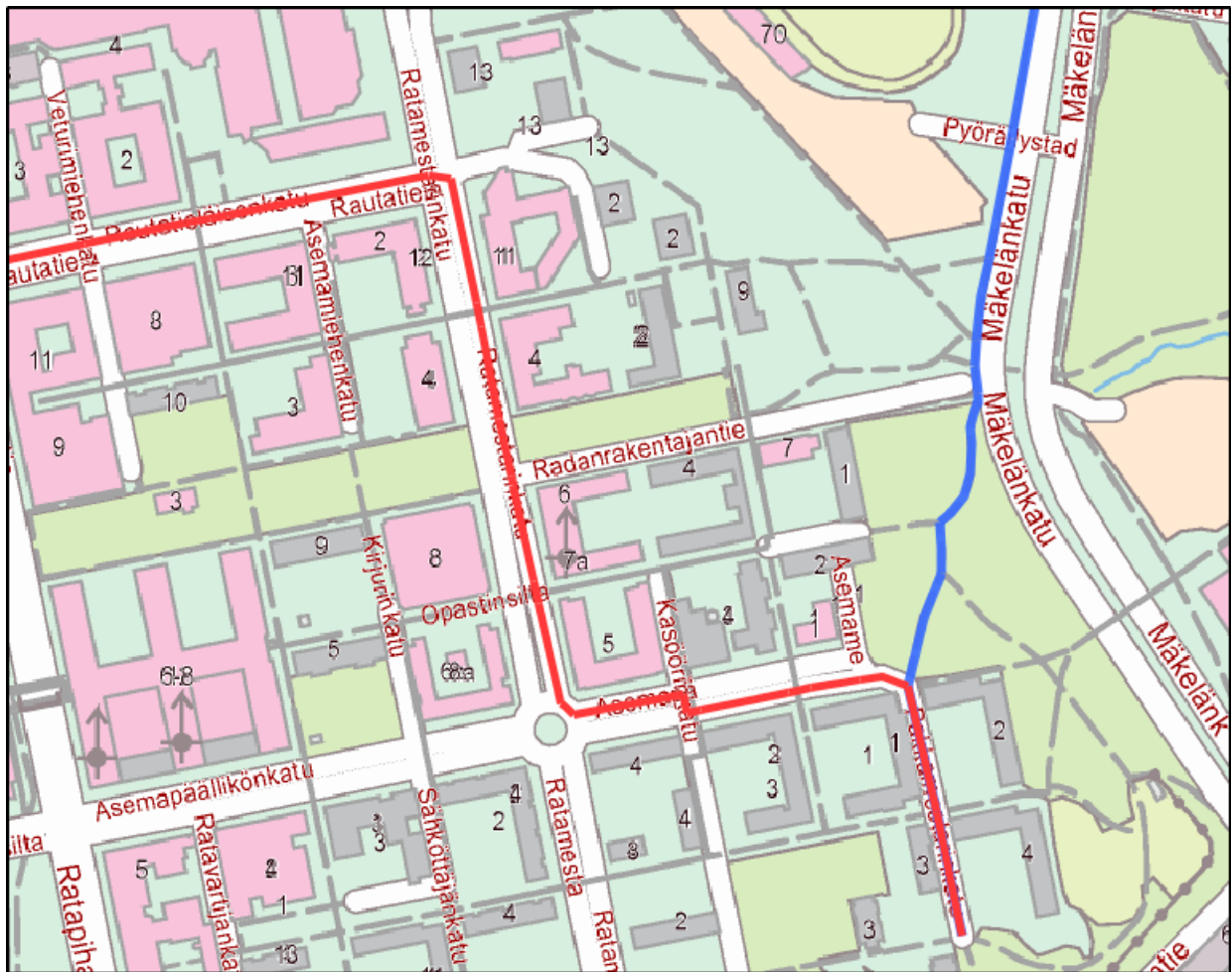
Spatialite osaa ottaa reitityksessä huomioon teiden yksisuuntaisuuden. Tätä ominaisuutta luovasti soveltamalla voidaan toteuttaa samasta tieverkkodatasta sekä reititys, joka käyttää kaikkia mahdollisia tieluokkia, että vain tietyt valitut tieluokat hyväksyvä reititys. Luova soveltaminen tarkoittaa sitä, että laitetaan teille kielletyn ajosuunnan merkki molempiin päihin. Tempu tuntuu niin yksinkertaiselta, että sen on pakko toimia, joten kokeillaan.

```
alter table tieviiva add column eteen;
alter table tieviiva add column taakse;
update tieviiva set eteen=1;
update tieviiva set eteen=0
where kohdeluokka in (12141,12312,12313,12314,12316);
update tieviiva set taakse=1;
update tieviiva set taakse=0
where kohdeluokka in (12141,12312,12313,12314,12316);
```

```
c:\ohjelmat\spatialite\spatialite_network -d mtk_tos.sqlite -T tieviiva
-f node_from -t node_to -g geometry -n nimi_suomi
--bidirectional --oneway-tofrom eteen --oneway-fromto taakse
-o tieviiva_auto_data
...
WARNING: arc forbidden in both directions; ROWID=3476041 From=475440 To=1791373
WARNING: arc forbidden in both directions; ROWID=3476043 From=1791373 To=2369127
WARNING: arc forbidden in both directions; ROWID=3476044 From=1290766 To=1511566
Step IV - final evaluation

Statistics
=====
# Arcs : 1997874
# Nodes: 3037794
Node max incoming arcs: 7
Node max outgoing arcs: 7
# Nodes cardinality=1: 86517 [terminal nodes]
# Nodes cardinality=2: 667173 [meaningless, pass-through]
=====

CREATE VIRTUAL TABLE "tieviiva_auto_net" USING
VirtualNetwork("tieviiva_auto_data")
```



Kaiken kestää?

Tietokannan teko:

Zip-tiedostoja on yhteensä 115 kpl. Aikaa kuluu

1. GML-tiedostojen vienti Spatialite-tietokantaan
09.08 → 09.17 = 9 min
2. Spatiaali-indeksin teko
09.23 → 09.29 = 6 min

Yhteensä 15 minuuttia

Vertailun vuoksi mittaustulos GML-tiedostojen viemisestä tietokantaan ogr2ogr-ohjelman oletusasetuksilla, jolloin spatiaali-indeksi luodaan heti, kun ensimmäinen GML-tiedosto on viety tauluun. Tämä johtaa siihen, että kun seuraavia GML-tiedostoja lisätään tauluun, niin samalla myös päivitetään spatiaali-indeksiä, mikä on huomattavasti hitaampaa.

10.32 -> 11.43

71 minuuttia

Kaikkia teitä käyttävät reitityksen luonti:

nodes 4:17 min

spatiaali-indeksi 5:22 min

node_from	26:20 min
node_to	27:26 min
silmukoiden siivous	1:00 min
yhteensä	64:25 min

Lisäksi on ajettava spatialite_network-komento, joka vie aikaa 4 min 20 sek.

Vaihtoehtoisen reitityksen teko:

Aika toistaiseksi mittaamatta, eteen- ja taakse-sarakkeiden päivitykseen menee yhteensä ainakin tunti, ehkä enemmänkin.

Liite 1: SQL-komentotiedosto

```
-- Tähän voi kirjoittaa kommentteja, kunhan kirjoittaa
-- rivin alkuun kaksi tavuviivaa
select datetime('now')||' alku';
CREATE TABLE nodes AS
SELECT ST_StartPoint(geometry) AS node_geometry
FROM tieviiva
UNION
SELECT ST_EndPoint(geometry) AS node_geometry
FROM tieviiva;
-- UNION hoitaa sen, että identtisistä koordinaateista
-- tallennetaan vain yksi piste tauluun
select datetime('now')||' nodes tehty';
alter table tieviiva add column node_from;
alter table tieviiva add column node_to;
select datetime('now')||' lisätty node_from ja node_to';
select RecoverGeometryColumn('nodes','node_geometry',3067,'POINT');
select datetime('now')||' geometriakenttä tehty';
-- kerrottiin, että node_geometry-kentässä on pisteitä,
-- joiden projektio on EPSG:3067
select CreateSpatialIndex('nodes','node_geometry');
select datetime('now')||' spatiaali-indeksi tehty';
-- tehtiin spatiaali-indeksi ja kerrottiin se
update tieviiva set node_from =
(select rowid from "nodes"
where "nodes"."node_geometry"=StartPoint(tieviiva.geometry) and
"nodes"."rowid" in
(select rowid from spatialindex where f_table_name='nodes'
and search_frame=StartPoint(tieviiva.geometry)));
select datetime('now')||' node_from täytetty';
-- Tässä kirjoitettiin tieviivatauluun alkupisteiden yksilöivät
-- tunnisteet ja käytettiin hyväksi Spatialiten spatiaali-indeksiä
update tieviiva set node_to =
(select rowid from "nodes"
where "nodes"."node_geometry"=EndPoint(tieviiva.geometry) and
"nodes"."rowid" in
(select rowid from spatialindex where f_table_name='nodes'
and search_frame=EndPoint(tieviiva.geometry)));
select datetime('now')||' node_to täytetty';
-- Tässä kirjoitettiin tieviivatauluun alkupisteiden yksilöivät
-- tunnisteet ja käytettiin hyväksi Spatialiten spatiaali-indeksiä
create table rings as
select * from tieviiva
where node_from=node_to;
select datetime('now')||' silmukat kopioitu';
-- Kopioitiin silmukkamaiset viivat, ja nyt ne tuhotaan
```

```
delete from tieviiva where node_from=node_to;
select datetime('now')||' silmukat poistettu';
select datetime('now')||' valmis';
```