



Ville Leinonen

**DIGIROAD-AINEISTON SOVELTUVUUS TIEOSAN
TIEOSOITTEEN LASKEMISEEN**

**DIGIROAD-AINEISTON SOVELTUVUUS TIEOSAN
TIEOSOITTEEN LASKEMISEEN**

Ville Leinonen
Opinnäytetyö
Kevät 2013
Tietotekniikan koulutusohjelma
Oulun seudun ammattikorkeakoulu

TIIVISTELMÄ

Oulun seudun ammattikorkeakoulu
Tietotekniikan koulutusohjelma

Tekijä(t): Ville Leinonen

Opinnäytetyön nimi: Digiroad-aineiston soveltuvuus tieosan tieosoitteen laskemiseen

Työn ohjaaja(t): Pekka Alaluukas

Työn valmistumislukukausi ja -vuosi: Kevät 2013 Sivumäärä: 44 + 3 liitettä

Työn tarkoituksena oli tutkia Digiroad-paikkatietoaineiston soveltuvuutta tieosan tieosoitteen laskemiseen GPS-laitteella mitatussa koordinaattipisteessä (GPS-pisteessä). Työssä selvitettiin kehitysympäristöksi valitun avoimen lähdekoodin sovellusten riittävyys. Tieosoitteen tarkkuus on riippuvainen pohja-aineistosta, joten tämä työ keskittyi Digiroad-aineiston käyttöönottoon, virheiden löytämiseen ja niiden korjaamiseen siten, että se ei aiheuttaisi aineistoon epätarkkuutta.

Työn tuloksena ilmeni, että työn alussa käyttöön otettu PostGIS-paikkatietokanta on toiminnallisuudeltaan enemmän kuin riittävä annettuun tehtävään. Digiroad-aineistossa oli noin 14 %:ssa tieosista sellaisia virheitä, jotka estivät laskemisen ja joihin ei löytynyt korjausmenetelmää. Noin 86 % aineistosta oli hyvänlaatuista; tieosoite pystyttiin laskemaan aineiston tarkkuuden rajoissa.

Johtopäätöksenä voidaan todeta, että pohja-aineiston laatu ja sen virheiden tunnistaminen ja dokumentointi on oleellisin osa järjestelmän toimintaa sen kaikissa vaiheissa. Lisäksi on oleellista ymmärtää ja tunnistaa käytettyjen projektoiden aiheuttamat virheet ja muutokset paikkatietoaineistoon. Avoimen lähdekoodin sovellusympäristö on enemmän kuin riittävä aineiston käyttämiseen. Digiroad soveltuu pohja-aineistoksi varauksin.

Asiasanat: Digiroad, paikkatietojärjestelmät, koordinaatit, tiet, satelliittipaikannus

ABSTRACT

Oulu University of Applied Sciences
Degree in Information Technology

Author(s): Ville Leinonen

Title of thesis: Digiroad-aineiston soveltuvuus tieosan tieosoitteen laskemiseen
(Applying Digiroad-material to calculate the road address)

Supervisor(s): Pekka Alaluukas

Term and year when the thesis was submitted: Spring 2013

Pages: 44 + 3 appendices

The aim of this work was to study spatial data in Digiroad release of the Finnish Transport Agency (FTA) and its use in the calculation of exact road addresses (an exact location in a public road in Finland, defined by FTA) in a GPS-measured point in time and space with open source spatial tools. The accuracy of a road address is determined by the source material, so this paper focuses on importing of Digiroad data, finding errors and fixing them, so that errors can be minimized in the data.

The open source PostGIS spatial database is more than capable to deal with the task given. Digiroad data had in approximately 14% of road parts such errors that prevented calculation of the road address. No suitable fixing method was found for these errors. Approximately 86% of the data was of good quality and the road address was calculated according to the claimed accuracy of the data.

As a result one can claim that the quality of the source material, recognizing its errors and documenting them is a crucial part of the workings of the system in all its stages. Also, it is relevant to understand and recognize the problems and errors posed by the used projections and the changes they cause in the spatial database. Open source spatial tools are too good for you; they are more than enough to deal with the tasks set for them in this paper. Digiroad data is not that good source material because it has so many errors.

The work is written in Finnish.

Keywords: Digiroad, geographic information systems, coordinates, GPS

SISÄLLYS

TIIVISTELMÄ	1
ABSTRACT	2
SISÄLLYS.....	3
LYHENNELUETTELO.....	5
1 JOHDANTO.....	9
2 PAIKKATIETO	10
2.1 Vektori	10
2.2 Rasteri	12
2.3 Referenssikoordinaattijärjestelmät.....	13
2.4 Projektiot	15
2.5 Astekoordinaattien esitystavat	15
2.6 Paikkatietokannat	16
3 DIGIROAD-AINEISTO.....	17
4 SOVELLUKSEN KEHITYSYMPÄRISTÖ	20
4.1 Tietokanta.....	20
4.2 PostGIS-tietokannan käyttämät paikkatiedon tyypit	22
5 DIGIROAD-AINEISTON KÄYTTÖÖNOTTO.....	23
5.1 Aineiston tuominen tietokantaan.....	23
5.2 Aineiston muuntaminen koordinaattijärjestelmien välillä	23
5.3 Aineiston laadun tarkistaminen ja korjaaminen	26
5.4 Geometrioiden luominen R-muodon lineaarisesti referoidusta tiedosta.....	26
5.5 Referoitujen geometroiden puhdistaminen.....	26
6 TIEOSOITTEIDEN GEOMETRIOIDEN LUOMINEN.....	28
6.1 Tieosoite ja tieosa.....	28
6.2 Tieosoite Digiroad-aineiston segmenttitaulussa	29
6.3 Tieosan osageometrioiden ominaisuudet	29
6.4 Tieosan osageometrioiden yhdistäminen tieosageometrioiksi.....	30
6.5 Yhdistettyjen tieosageometrioiden tyypit.....	32
6.6 Yhdistämisen ongelmat.....	33

7 YHDISTÄMISEN TULOKSET	36
7.1 Aineiston muuntaminen ja laatu.....	36
7.2 Tieosien geometrioiden luominen.....	36
8 TIEOSOITTEEN LASKEMINEN.....	40
9 YHTEENVETO JA POHDINTA.....	42
LÄHTEET.....	43
LIITE 1: AINEISTON KÄSITTELYYN KÄTYTETYT BASH-KOMENTOSARJAT.....	
LIITE 2: AINEISTON KÄSITTELYYN KÄYTETTYJÄ SQL-KOMENTOSARJOJA...	
LIITE 3: HAVAINNOLLISTAVIA KUVIA.....	

LYHENNELUETTELO

3Dm	Kaksiulotteinen m-arvolla; ks. M-arvo.
3Dz	Kolmiulotteinen; kaksiulotteinen, jossa mukana Z-arvo.
4D	Kolmiulotteinen m-arvolla, ks. 3Dm.
ArcGIS	Esri-yrityksen toimittama paikkatietojärjestelmä.
Datumi	Referenssikoordinaattijärjestelmä (8, sivu 3D-koordinaatistot, alisivu Määritelmiä; 5, s. 2).
EPSG	European Petroleum Survey Group, lopettanut toimintansa vuonna 2005, korvaava järjestö OGP (10, hakusana EPSG).
ESRI shapefile	Paikkatiedon tallennusmuoto, jonka on kehittänyt Esri.
ETRS89	European Terrestrial Reference System 1989. 3D-koordinaattijärjestelmä, joka on kiinnitetty Euraasian mannerlaatan yhtenäiseen osaan (8, sivu 3D-koordinaatistot, alisivu EUREF-FIN; 8, sivu Tasokoordinaatistot, alisivu ETRS-TM35FIN).
ETRS-TM35FIN	ETRS89-koordinaattijärjestelmän kanssa käytettävä karttaprojektio ja tasokoordinaatisto. Suomen maastokartoissa vuodesta 2005 käytetty tasokoordinaatisto, joka poikkeaa UTM-projektioista (ks. UTM) siten, että Suomi kuuluu kokonaisuudessaan yhteen 13 asteen levyiseen projektiokaistaan. Kaista sijoittuu itäisten pituusasteiden 19 ja 32 välille ja sen keskimeridiaani on 27 astetta itäistä pituutta. Kyseessä on siten UTM-projektion kaista 35, jota on jatkettu länteen viisi astetta ja itään kaksi (8, sivu Karttaprojektiot).
Euref-FIN	ETRS89-järjestelmän realisaatio Suomessa (8, sivu 3D-koordinaatistot, alisivu EUREF-FIN).
EWKB	Extended WKB. PostGIS-järjestelmän laajennus. WKB-esitystapaan. Ks. WKB.

EWKT	Extended WKT. PostGIS-järjestelmän lääjennus WKT-esitystapaan. Lisää tuen upotetulle SRID:lle sekä 3Dm, 3Dz ja 4D koordinaateille. Ks. WKT.
GML	Geography Markup Language. Paikkatiedon esitysformaatti (1).
GPS	Global Positioning System, globaali paikannusjärjestelmä, joka toimii satelliittien avulla (10, hakusana GPS).
GRS80	VGS84-järjestelmän käyttämä korkeusjärjestelmä.
ISO	International Organisation for Standardization, Kansainvälinen standardointijärjestö.
KKJ	Kartastokoordinaattijärjestelmä; suomalainen koordinaattijärjestelmä, joka oli käytössä vuodesta 1970 vuoteen 2005. Se perustuu perustuu Hayfordin vertausellipsoidiin, joka tunnetaan myös nimellä International 1924. Karttakoordinaatit ovat tämän ellipsoidipinnan koordinaatteja (8, sivu Tasokoordinaatistot, alasivu KKJ).
KKJ3	Kartastokoordinaattijärjestelmä; myös KKJ zone 3; ks. KKJ. Zone 3 ilmaisee, että koordinaatiston keskimeridiaanina käytetään astetta 27 (8, sivu Tasokoordinaatistot, alasivu KKJ).
KML	Keyhole Markup Language. Paikkatiedon esitysformaatti (2).
LINestring	WKT:n nimitys viivaa esittävälle geometriatyypille; WKT:n käyttämä nimitys kirjoitetaan normaalista englanninkielen termistä eroavasti kokonaan isolla.
M-arvo	Vektorien yhteydessä käytetty (3Dm, 4D) pisteellä (viivavektorien yhteydessä alku- tai loppupiste) oleva arvo, joka voi olla mitattu arvo, esimerkiksi kaivon syvyys, tai merkityn pisteen etäisyys alkupisteestä, tai melkein mitä tahansa arvon määrittäjästä riippuen.

MapInfo	Eräs paikkatietojärjestelmä.
MULTILINESTRING	WKT:n nimitys monta viivaa esittävälle geometriatyypille; ks. LINESTRING.
MULTIPOINT	WKT:n nimitys montaa pistettä esittävälle geometriatyypille; ks. POINT.
MULTIPOLYGON	WKT:n nimitys montaa POLYGONia esittävälle geometriatyypille, ks. POLYGON.
MySQL	Eräs tietokantaohjelmisto.
OGC	Open Geospatial Consortium, Inc on kansainvälisen teollisuuden yhteenliittymä joka julkaisee OpenGIS® standardeja tukeakseen yhteensopivia ratkaisuja verkon paikkatietosovelluksissa.
OGP	Gas Producers Surveying and Positioning Committee; komitea, joka ylläpitää ESPG SRID -tietokantaa.
OpenJump	Avoimen lähdekoodin paikkatietojärjestelmä, kirjoitettu Javalla.
POINT	WKT:n nimitys pistettä edustavalle geometriatyypille.
POLYGON	WKT:n nimitys kaksiulotteista monikulmiota edustavalle geometriatyypille.
PostgreSQL	Eräs tietokantaohjelmisto.
PostGIS	Laajennus PostgreSQL-tietokantaan, joka lisää tuen geometriaobjekteille ja paikkatietorelaatioille.
PROJ.4	Karttaprojektiokirjasto, joka oli aluperin Gerald Evenden, nykyisin USGS:n ylläpitämä.
QuantumGis	Eräs avoimen lähdekoodin paikkatietojärjestelmä.
SQL-MM part 3	ISO-standardi paikkatiedon esitystavallle (4).
SRID	Spatial Reference Identification. Koordinaattijärjestelmän referenssitunniste. Standardin on määritellyt ESPG ja sen omistaa ja ylläpitäjä on vuoden 2005 jälkeen ollut Gas Producers (OGP) Surveying and Positioning -komitea (http://www.epsg.org/) (10, hakusana SRID).

SVG	Simple Vector Graphics. Vektoritiedon esitysformaatti, jota käytetään esittämään myös paikkatietoa.
Udig	Avoimen lähdekoodin paikkatietojärjestelmä.
USGS	U.S. Geological Survey.
UTM	Universal Transverse Mercator, Gauss-Krüger-projektion kaltainen oikeakulmainen karttaprojektio.
WGS84	World Geodetic System 1984. GPS-järjestelmän käyttämä horisontaalinen ja vertikaalinen koordinaattijärjestelmä ja siihen liittyvä geoidimalli (9, hakusana WGS84).
WKB	Well Known Binary -format. Binääripohjainen standardi paikkatiedon esittämiseen tietokoneen luettavassa muodossa. Määritelty OpenGIS-standardissa; vrt. EWKB.
WKT	Well Known Text -format. Ihmisen luettava paikkatiedon esitysformaatti. Määritelty OpenGIS-standardissa; vrt. EWKT.
Xbase	Tiedostoformaatti, jonka yleisest käytetty päätte on .dbf.
YKJ	Yhtenäiskoordinaatisto; yksi KKJ:n suorakulmaisista tasokoordinaatistoista; vastaa KKJ zone 3 (9, hakusana YKJ).

1 JOHDANTO

Työn tarkoituksena on tutkia Digiroad-paikkatietoaineiston soveltuvuutta tieosan tieosoitteen laskemiseen GPS-laitteella mitatussa koordinaattipisteessä (GPS-pisteessä). Työssä selvitetään kehitysympäristöksi valitun avoimen lähdekoodin sovellusten (PostGIS) riittävyys. Tieosoitteen tarkkuus on riippuvainen pohja-aineistosta, joten tämä työ keskittyy Digiroad-aineiston käyttöönottoon, virheiden löytämiseen ja niiden korjaamiseen siten, että se ei aiheuta aineistoon epätarkkuutta.

Digiroad on paikkatietokokoelma, joka sisältää Suomen tiestön keskiviivageometriat ja ominaisuustiedot. Opinnäytetyön tarkoituksena on muuntaa aineisto käytettäväksi avoimen lähdekoodin paikkatietojärjestelmässä (PostGIS) ja tutkia aineiston soveltuvuutta tieosoitteen laskemiseen GPS-pisteen avulla. PostGIS-paikkatietokantaa käytetään tietokantana ja aineiston analysointijärjestelmänä.

Työn toinen luku käsittelee paikkatiedon perusteet. Kolmas luku esittelee Tiehallinnon Digiroad-aineiston tekniset tiedot ja viidennessä luvussa esitellään kehitysympäristö eli PostGIS. Viides luku kuvaa Digiroad-aineiston käyttöönottoa, eli aineiston muuntamista PostGIS-paikkatietojärjestelmään ja siihen liittyviä ongelmia. Kuudennessa luvussa käsitellään tieosoitetta ja siihen liittyviä ongelmia. Seitsemännessä luvussa esitetään ratkaisu tavoitteeseen, eli tieosoitteen laskemiseen GPS-pisteellä. Tulokset on esitetty kahdeksannessa luvussa ja niitä pohditaan syvemmin viimeisessä luvussa.

2 PAIKKATIETO

Paikkatieto on sijainti- ja ominaisuustiedon looginen tietokokonaisuus. Paikkatiedon kuvaamiseen tietojärjestelmissä käytetään yleensä vektori- ja rasterimateriaalia. Digiroad-aineistossa paikkatiedot saadaan yhdistämällä vektorigeometriat ja -ominaisuudet, jotka sijaitsevat erillisissä tiedostoissa.

2.1 Vektori

Paikkatietojärjestelmissä paikkatietoa kuvataan vektorimuodossa pisteinä (POINT), viivoina (LINESTRING, MULTILINESTRING) ja alueina (POLYGON). Tutkielman luettavuuden vuoksi kaikki paikkatietoarvot ilmaistaan EWKT- tai WKT-muodossa. Piste on yksinkertainen, nollaulotteinen sijainti referenssikoordinaatistossa, jota käytetään kuvaamaan esimerkiksi kaivoa. Viiva on yksiulotteinen kahden tai useamman pisteen väliä kuvaava vektori. Viivageometrioita voidaan käyttää pituuden laskemiseen. Alue on kaksiulotteinen geometria, joka soveltuu kuvaamaan alueita ja niitä peittäviä ominaisuuksia. (9, hakusana Vektori.)

Pistevektorin esimerkki on annettu taulukossa 1. SRID (Spatial Reference ID) tarkoittaa koordinaattijärjestelmän referenssiä. SRID 4326 tarkoittaa, että kyseessä on WGS84-koordinaatistossa oleva piste. POINT määrittää tässä kahdella arvolla pisteen X- ja Y-koordinaatit; arvot erotetaan toisistaan väleillä. EWKT-formaatissa useampiulotteisten (esim. 4D) muotojen ero 2D-muotoon verrattuna on vain välilyönnillä erotettujen arvojen määrä. (12, kohta 4.1.2.)

Esimerkkiluettelossa (taulukko 1) on annettu myös viivavektorin esimerkki 3D-muodossa kolmen pisteen välillä. SRID 2393 viittaa siihen, että annettu vektori kuuluu KKJ3-koordinaatistoon. LINESTRING määrittää viivavektorin kolmen pisteen X-, Y- ja Z-koordinaattien avulla. Koordinaatit erotetaan ilmaisussa toisistaan väleillä. Pisteet on erotettu toisistaan pilkuilla. EWKT-formaatti määrittää kolmannen pisteen olevan M-

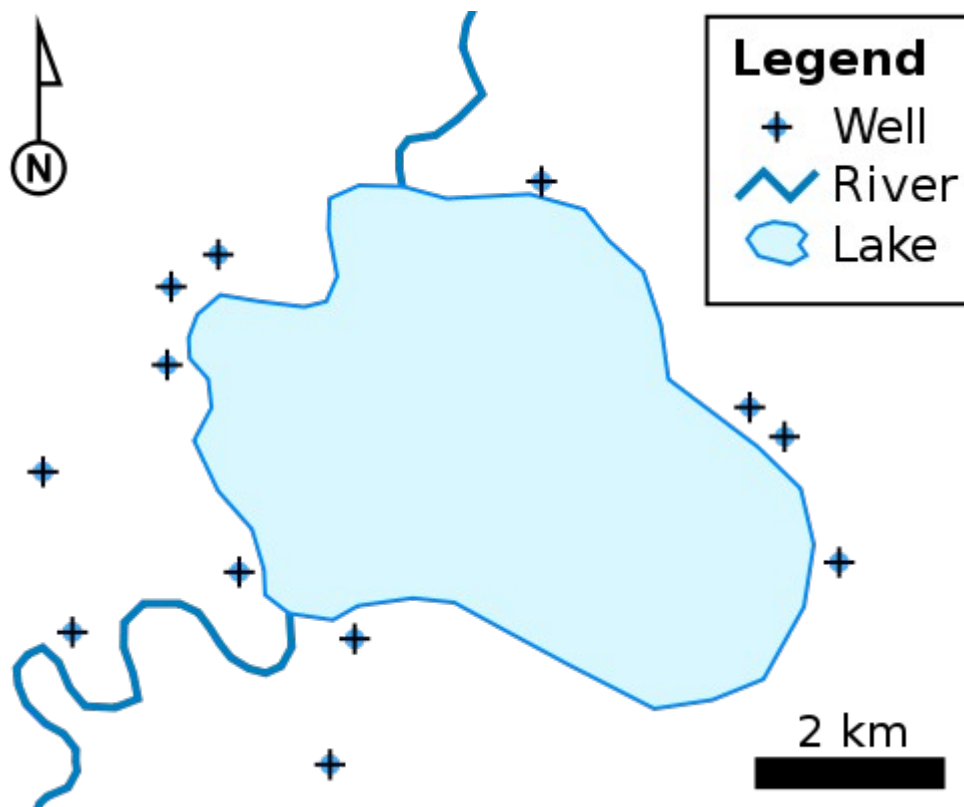
arvo, mikäli tyyppimääritelmässä on M lopussa, esimerkiksi LINESTRINGM ja POINTM. 4D-muodossa viimeinen arvo on aina M-arvo. (12, kohta 4.1.2.)

TAULUKKO 1: Esimerkkejä EWKT-formaatin ilmaisutavoista.

Pistevektori	SRID=4326;POINT(65.452 42.424242)
Viivavektori	SRID=2393;LINESTRING(3536583.9722 565 7145997.72312178 -49.3143008517072 , 3536633.95942334 7145974.13437963 -49.1545709301536 , 3536733.91562356 7145925.29559281 -49.2851296706899)
Polygoni	POLYGON(((0 0 0,4 0 0,4 4 0,0 4 0,0 0 0), (1 1 0,2 1 0,2 2 0,1 2 0,1 1 0)))

Vektorien koordinaattien esittämiseen käytetään 2D-, 3Dz-, 3Dm- ja 4D-muotoja. Eri standardit määrittelevät WKT- ja WKB-ilmaisunsa eri tavalla. OpenGIS-standardi vaatii vain tuen 2D-muodolle, kun taas SQL/MM-standardi määrittelee ja vaatii lisäksi myös 3D-, 3Dz-, 3Dm- ja 4D-muodot. PostGis-järjestelmän tukee lisäksi laajennettua EWKT-muotoa, joka vastaa WKT:tä, mutta lisää SRID:n (eli koordinaattijärjestelmäreferenssin) WKT:n esitykseen. Sama koskee soveltaen WKB-esitystä. (10, hakusana WKT; 3, lviili; 4; 12, kohta 4.1.3.) Vektori ei siis ilmaise suoraa, vaan tietyn suuntaista viivaa. Kuvassa 1 on esimerkkinä jokea ilmaiseva viivavektori.

2D-esitystavassa on X- ja Y-pisteet, 3Dz on kolmiulotteinen piste X-,Y- ja Z-koordinaateilla, 3Dm on X- ja Y- koordinaatit sekä M-pisteen omaava pistevektori, missä M on tarkoittaa M-arvoa. 4D esitystapa sisältää X-, Y-, Z- ja M-arvot. Lisäksi on olemassa muita formaatteja kuten esimerkiksi SVG, KML ja GML (1; 2).

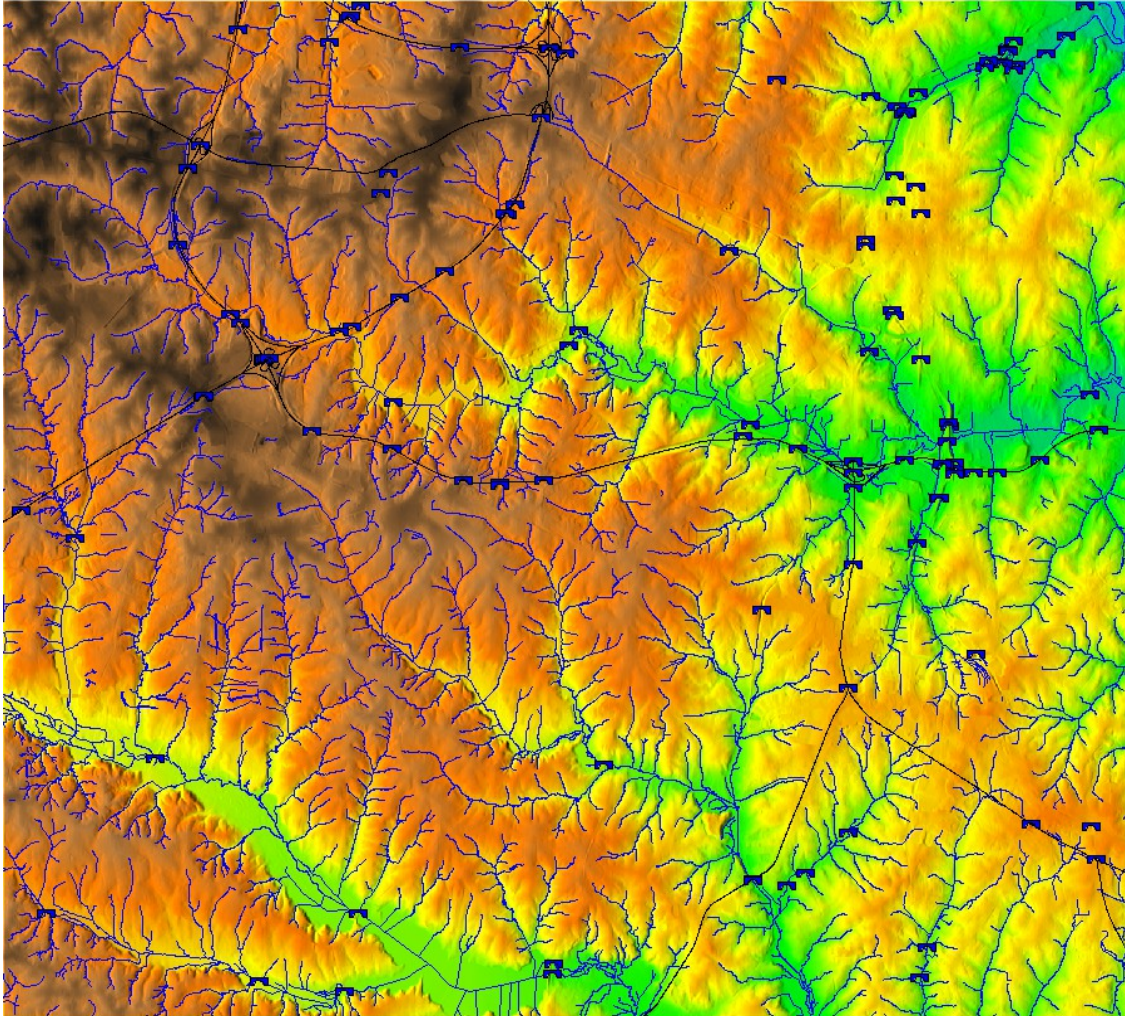


KUVA 1: Esimerkkikartta, jossa on kuvattu viivavektorilla joki. Lähde: (9.).

M-arvo vektoriaineiston geometrioissa on ns. ”measure”-arvo. M-arvo voi tarkoittaa mitä vain arvoa, jonka sen määrittäjä on sille antanut. Se voi kuvata esimerkiksi pistegeometriassa kaivon syvyyttä tai pisteen korkeutta meren pinnasta. Linearisesti referoiduissa aineistoissa M-arvo on referenssiluku, jonka perusteella viiva voidaan jakaa osiin. M-arvolla ei ole tarkkaa universaalia määritelmää, vaan sen merkitys vaihtelee aineistoissa. (7, s. 9–10.)

2.2 Rasteri

Rasteridata on käytännössä digitaalinen kuva, jossa yksi pikseli vastaa sen peittämän alueen ominaisuustietoa. Yksittäinen pikseli voi kuvata esimerkiksi sen peittämän alueen lämpötilaa tai korkeutta. Jälkimmäisestä on esimerkki alla kuvassa 2.



KUVA 2: Esimerkki rasterikartasta. Kartassa on kuvattu pikseleiden väriarvolla sen korkeutta. Kartassa on lisäksi viivavektoreilla ilmaistuja jokia. Lähde: <http://www.grassbook.org/gallery/ch3getstarted/images.html>.

2.3 Referenssikoordinaattijärjestelmät

Datumi tarkoittaa referenssipistettä. Se tarkoittaa myös referenssipisteiden kokoelmaa. Referenssikoordinaattijärjestelmissä koordinaatisto rakennetaan relatiiviseksi yksittäisten referenssipisteiden suhteen. Horisontaalisia referenssipisteitä käytetään kuvaamaan maan pinnalla olevaa pistettä pituus- ja leveysasteina tai muussa koordinaattijärjestelmässä. Vertikaaliset referenssipisteet kuvaavat korkeutta tai syvyyttä. (5, s. 2.)

Horisontaalisia koordinaattijärjestelmiä ovat esimerkiksi WGS84 ja Euref-FIN. WGS84 on GPS-paikannusjärjestelmän käyttämä horisontaalinen koordinaattijärjestelmä. Se on suunniteltu siten, että se toimii riittävällä, noin muutaman metrin tarkkuudella koko maapallon alueella. Euref-FIN on määritelty siten, että sen piste vastaa metrin tarkkuudella WGS84-järjestelmän pistettä Suomen alueella. Digiroad-aineisto käyttää Euref-FIN-järjestelmää, joka vastaa käytännön tarkoituksissa WGS84-järjestelmää. Ero WGS84- ja Euref-FIN-järjestelmän välillä on se, että Euref-FIN on sidottu ETRS89-koordinaatistoon; ETRS89 on relatiivinen suhteessa ITRS:n geodeettiseen datumiin, eli se liikkuu Euraasian laatan liikkeen suhteessa. WGS84-koordinaatisto, joka on geosentrinen, ei liiku laatan mukana. (5, s. 5.)

Vertikaalisia koordinaattijärjestelmiä ovat esimerkiksi GRS80, WGS84, N60 ja N2000. Kahdessa ensin mainitussa järjestelmässä referenssipiste on suhteessa etäisyyteen maapallon laskennallisesta keskipisteestä. ”N2000-järjestelmä ei ole aiempien kansallisten korkeusjärjestelmien tavoin sidottu Helsingin keskimääräiseen merenpintaan vaan se perustuu NAP:n määrittämään korkeustasoon.” Digiroad-aineisto käyttää vertikaalisena koordinaattijärjestelmänä N60-järjestelmää, joka on sidottu Helsingin keskivedenpintaan. (6, s. 1, 7.) Liitteessä 3 on havainnollistettu erinäisten vertikaalisten koordinaattijärjestelmien keskinäisiä suhteita ja merkitseviä eroja. Suomen Geodeettinen laitos tarjoaa tarvittavat muunnosaineistot, mikäli on tarvetta siirtyä N60-, N2000- ja GRP80-järjestelmien välillä. GRS80 ja WGS84 vastaavat toisiaan käytännössä toisiaan, koska niiden välinen ero ei ole merkitsevä (n. 105 mikrometriä). (10, hakusana WGS.)

2.4 Projektiot

Maapallon esittäminen kaksiulotteisessa tasokoordinaatistossa on mahdotonta, koska maapallo on kolmiulotteinen, navoiltaan litistynyt ellipsoidi. Sen vuoksi on käytettävä karttaprojektiota. Projektion valinta riippuu siitä, mikä mitattava ominaisuus on tärkein kartassa, oli se sitten tila, muoto, suunta tai etäisyys. Aineiston tulee olla järkevissä projektiossa käyttötarkoituksen kannalta. Koordinaattijärjestelmän valinta määrittää käytetyn projektion.

Tässä tapauksessa valintana oli KKJ3 joka säilyttää etäisyydet mahdollisimman hyvin, sekä käyttää mittayksikkönä metrejä. Koko Suomen aineiston käsittelemiseen tällaisia ovat YKJ eli KKJ Zone 3 ja TM35FIN, joista Julkisen hallinnon suositus 154 asettaa etusijalle TM35FIN-projektion, koska KKJ on vanhentunut. KKJ-järjestelmä on poistumassa käytöstä, koska sen referenssipisteiden suhteet ovat muuttuneet sen käyttöönoton jälkeen maankuoren liikkeiden takia. Uusi suositus TM35FIN käyttää keskimeridiaanina 27 astetta. Etäisyyden ollessa keskimeridiaaniin alle 300 km on mittavirhe 100 metrin matkalla 7 cm. Pinta-alan muutos 100 m x 100 m alueella on 14 neliometriä. (5, s. 9.)

2.5 Astekoordinaattien esitystavat

Astekoordinaateilla on useampi esitystapa. Yleisin esitystapa GPS-laitteissa on WGS84-standardin mukainen dd mm ss,ssss. Tässä d tarkoittaa asteita, m minuutteja ja s tarkoittaa sekunteja, desimaalit pilkulla erotettuna. Kaksi muuta tapaa ovat dd mm,mmmm eli asteina ja minuutteina desimaaleineen, sekä esitys kokonaan desimaaliasteissa, jolloin muoto on dd,dddd. PostGIS-järjestelmä käyttää desimaaliasteita leveys/pituus-järjestyksessä, kun koordinaattijärjestelmänä on järjestelmä, joka käyttää yksikkönä asteita. (10, hakusana Aste.)

2.6 Paikkatietokannat

Paikkatietokannat ovat tietokantoja, jotka on optimoitu käsittelemään ja tallentamaan paikkatietoa eli geometrioita ja niihin liittyviä ominaisuuksia. Paikkatietokannat kykenevät tunnistamaan kahden geometrian välisiä suhteita, kuten esimerkiksi sen, onko pistegeometria oikealla vai vasemmalla puolella viivageometriaa suhteessa digitointisuuntaan tai onko piste POLYGON:in sisällä vai ulkona. Vaikka tavallinen tietokanta voi tallentaa koordinaattipisteitä, niissä ei ole toiminnallisuutta vertailla paikkatietojen keskinäisiä, geometrisiä suhteita. (9, hakusana Spatial database.)

Vapaan lähdekoodin paikkatietokantoja ovat mm. PostgreSQL-tietokantaohjelma PostGIS-laajennuksella ja MySQL-tietokantaohjelmistot. PostgreSQL:n PostGIS-laajennus tukee OpenGIS "Simple Features Specification for SQL" -määritelmää. PostGIS-laajennus lisää PostgreSQL-tietokantaan tuen geometriaobjekteille (pisteet, viivat, alueet) ja tarvittavat funktiot ja operaattorit, joilla työskennellään paikkatiedon kanssa. Esimerkiksi ST_length3d(geom)-funktio laskee viivageometrian pituuden ottaen huomioon kaikki kolme ulottuvuutta sekä koordinaattijärjestelmän käyttämän ellipsoidin. ST_Intersects(geom,geom)-funktio kertoo, mikäli kaksi geometriaa leikkaavat toisensa.

3 DIGIROAD-AINEISTO

Digiroad on Suomen Tiehallinnon julkaisema aineisto, joka sisältää Suomen kaikkien teiden keskiviivageometriat ja ominaisuustiedot. Digiroad-aineisto on saatavilla kolmessa toimitusmuodossa, jotka ovat joko Euref-FIN tai YKJ-koordinaatistossa jaoteltuna maakunnittain tai kunnittain. Digiroad on suunnattu yrityksille ja viranomaisille.

Digiroad R -toimitusmuoto toimitetaan ESRI shapefile -muodossa, joko Euref-FIN- tai YKJ-koordinaatistossa. R-toimitusmuoto sisältää "referenssiketjuille dynaamisesti segmentoituja ominaisuustietoja". *Digiroad XML R* -toimitusmuoto toimitetaan XML-tiedostoina, mutta sisältää samat tiedot kuin R-muoto ja samat aineiston käyttöönottomenetelmät toimivat. *Digiroad K* -toimitusmuoto toimitetaan ESRI shapefile -muodossa, joko Euref-FIN tai YKJ-koordinaatistossa. Siinä liikenne-elementit on katkottu ominaisuustiedoiltaan yhtenäisiin osiin. Digiroad-aineiston kuvaus on julkaistu palvelun tarjoajan sivulla. (7, s. 12–14.)

Aineisto toimitetaan DVD-levyllä, joka sisältää 31 ZIP-pakattua tiedostoa. ZIP-tiedostot sisältävä ESRI shapefile -kokonaisuuksia. Aineiston toimitus on jaoteltu lääneittäin. ESRI shapefile viittaa tiedostokokoelmaan, jossa on vähintään kolme tiedostoa: SHP-tiedosto sisältää itse geometrian (geometriat), PRJ-tiedosto sisältää käytetyn koordinaattijärjestelmän sekä projektion ja DBF-tiedosto sisältää SHP-tiedoston sisältämien geometrioiden ominaisuustiedot. DBF-tiedosto on xBase-formaatissa. Tiedostot käyttävät merkistöä ISO-8859-1.

Aineisto on jaettu tiedostoihin alla olevan mukaisesti (7):

1. Digiroad_KETJU-tiedostot sisältävät referenssiketjujen geometriat sekä ominaisuustiedot: Digiroad_KETJU.dbf-tiedosto sisältää ominaisuustiedot ja Digiroad_KETJU.shp sisältää geometriatiedot. Digiroad_KETJU.prj-tiedosto sisältää

edellämainittujen tiedostojen käyttämän koordinaattijärjestelmän. Työn toteutuksessa näiden tiedostojen sisältämiin tietoihin viitataan tietokannassa ketju-nimisenä tauluna.

2. Digiroad_LIIKENNE_ELEMENTTI-tiedostot sisältävät liikenne-elementtitaulun geometrian ja sen ominaisuustiedot. Ominaisuustietoja ovat mm. elementin liikennevirran suunta, osoitenumerot ja väylätyypit (yleinen tie, kevyen liikenteen väylä jne.). Työn toteutuksessa näiden tiedostojen sisältämiin tietoihin viitataan tietokannassa liikenne_elementti-nimisenä tauluna.

3. Digiroad_NIMI.dbf-tiedosto (vain yksi tiedosto) sisältää nimi-ominaisuudet, esimerkiksi tien tai sillan nimen. Tiedostoon viitataan muista ominaisuustauluista. Työn toteutuksessa näiden tiedostojen sisältämiin tietoihin viitataan tietokannassa nimi-nimisenä tauluna.

4. Digiroad_PALVELU-tiedostot sisältävät tiedot tiestöön liittyvistä palveluista, kuten esimerkiksi pysäköintialueista. Palvelu ilmaistaan antamalla pistesegmenttinä kohta, josta palveluun mennään ajoneuvolla sisään. Työn toteutuksessa näiden tiedostojen sisältämiin tietoihin viitataan tietokannassa palvelu-nimisenä tauluna.

5. Digiroad_SEGMENTTI.dbf-tiedosto (R-muodoissa; kolme tiedostoa K-toimitusmuodosta riippuen) sisältää segmenttitaulun. Segmentit kuvaavat teiden eri osien (engl. segment) ominaisuustietoja, kuten esimerkiksi tienumeroja, tieosia tai voimassa olevia nopeusrajoituksia. Digiroad R -toimitusmuodossa segmenttitaulu sisältää pelkän viitteen ketju-taulussa olevaan geometriaan, josta itse geometriat luodaan lineaarisen referoinnin avulla. (Lineaarista referoinnista, ks. 7, s. 9–10.) Digiroad K-toimitusmuodossa segmenttitaulu ei sisällä viitettä, vaan tiedot geometriasta sekä niihin yhdistettävät ominaisuudet. Työn toteutuksessa näiden tiedostojen sisältämiin tietoihin viitataan tietokannassa segmentti-nimisenä tauluna.

Tieosoiteaineisto sisältää noin 170 Mt geometrioita niiden luomisen jälkeen, yhteensä noin 83 000 rivillä. Tieosoitetietoja eli uniikkeja tieosan ja tienosanumeron yhdistelmiä

on tietokannassa 18 162 kappaletta. Koko aineisto vie PostgreSQL-kannassa n. 12 Gt tilaa.

4 SOVELLUKSEN KEHITYSYMPÄRISTÖ

Opinnäytetyön tarkoituksena oli tieosoitteen laskeminen Digiroad-aineiston avulla GPS-pisteessä. Tätä varten Digiroad-aineisto otettiin käyttöön kehitysympäristössä, ja kehitettiin tarvittavat komentosarjat, joilla annettu tehtävä pystyttiin toteuttamaan.

Käytetyssä kehityskoneessa oli AMD Opteron 165 -moniydinprosessori, 2 Gt DDR-400 RAM-muistia ja SATA-2-kiintolevy. Tieosoitteiden laskemiseen tarvittavan paikkatietokannan kehitysympäristönä toimi alun perin Ubuntu 7.10 "Gutsy"-Linux-jakelu, joka ajan kuluessa vaihtui 9.10 "Karmic"-versioon. Ubuntu valittiin, koska tavoitteena oli nimenomaisesti avoimen lähdekoodin työkalujen riittävyden tarkastelu, ja Ubuntu on avoimen lähdekoodin käyttöjärjestelmäjakeluista suosituin.

Ohjelmistona oli käytössä PostgreSQL-tietokantaohjelmisto PostGIS-laajennuksella. PostgreSQL-tietokannasta käytettiin versioita 8.1, 8.2, 8.3 ja 8.4. ja PostGIS-laajennuksesta versioita 1.1.0–1.5.1. Paikkatiedon visualisoitiin ja muokkaamiseen käytettiin QuantumGIS-, Udig- ja OpenJUMP-ohjelmia. Tietokannan hallinnointiin käytettiin PgAdmin3-ohjelmaa.

Digiroad R -aineisto vaatii vähintään PostGIS:n version 1.1.0, koska lineaariseen referointiin tarvittavia funktioita ei ole vanhemmissa versioissa. Digiroad K -aineiston kanssa voidaan käyttää aikaisempiakin versioita. Oleellisista käytetyistä ohjelmista on tarjolla versiot myös muille käyttöjärjestelmille tai vastaava vapaan lähdekoodin ohjelma.

4.1 Tietokanta

Paikkatietokantajärjestelmäksi valittiin PostgreSQL PostGIS-laajennuksella. PostGIS on tällä hetkellä standardijärjestelmä avoimeen lähdekoodiin perustuvissa paikkatietokannoissa, koska se implementoi OpenGIS "Simple Features Specification for SQL" 1.2. -määritelmän mukaisen tuen paikkatiedolle. Tämän ansiosta tietokanta on

yhteensopiva yleisimpien paikkatiedon esitystapojen kanssa. Näitä ovat mm. WKT (Well Known Text), KML (Keyhole Markup Language), SVG (Simple Vector Graphics) ja GML (Geography Markup Language).

PostGIS-laajennus lisää joukon paikkatiedon vertailuun ja muokkaamiseen tarvittavia funktioita. Esimerkiksi paikkatiedon vertailuun on tarkoitettu `st_intersect(geom,geom)`, joka tutkii, risteävätkö geometriat, sekä `st_overlaps(geom,geom)`, jota käytetään geometrioiden päällekkyyksien tutkimiseen. Esimerkkinä paikkatiedon muokkaamiseen käytettävistä funktioista on hyvä mainita `st_transform(geom, srid)`, joka muuntaa geometrian toiseen horisontaalisen koordinaattijärjestelmään, ja `st_linemerge()`, joka pyrkii yhdistämään joukon viivageometrioita yhdeksi jatkuvaksi LINESTRING-geometriaksi.

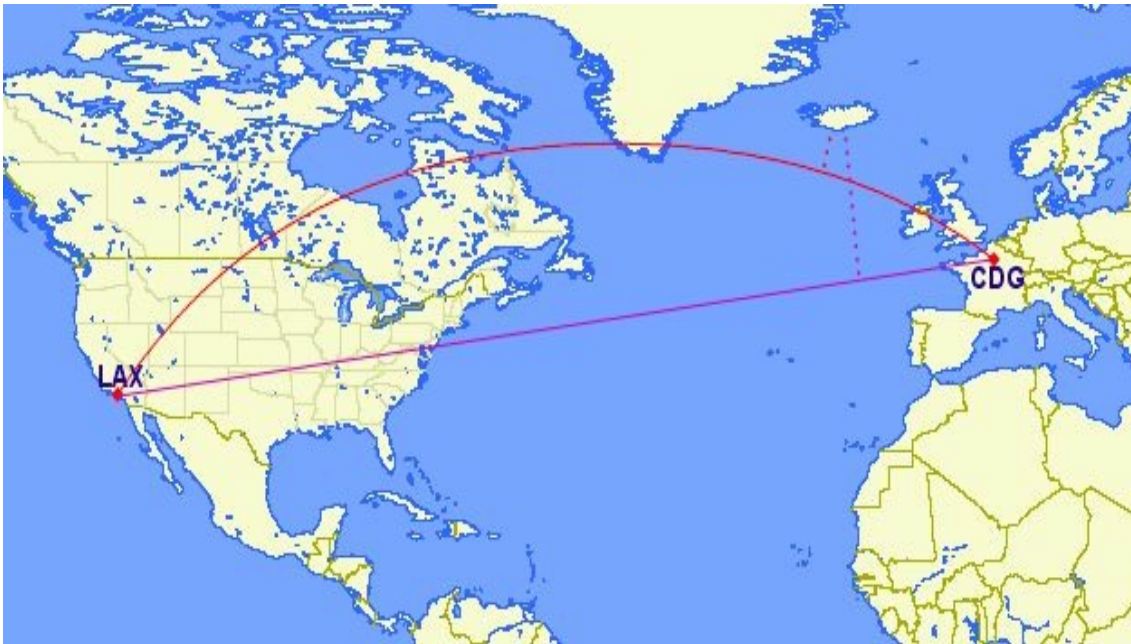
PostGIS-paikkatietokannassa geometriat esitetään pisteinä (POINT), multipisteinä (MULTIPOINT), viivoina (LINESTRING), multiviivoina (MULTILINESTRING), alueina (POLYGON) ja multialueina (MULTIPOLYGON) tai geometriajoukkoina (GEOMETRYCOLLECTION). Jatkossa käytetään englanninkielistä, WKT:n nimityksiin perustuvaa terminologiaa seuraamisen helpottamiseksi.

Käytännössä POINT esittää yksittäistä pistettä, LINESTRING kahden tai useamman pisteen välillä jatkuvana kulkevaa viivaa, ja POLYGON aluetta. MULTI-prefiksi muuttaa geometriatyypin monen vastaavan tyyppin joukoksi. GEOMETRYCOLLECTION on joukko, joka sisältää useaa eri tyyppiä. MULTI-tyyppijä käytetään tilanteissa, jolloin voidaan tiivistää useita geometrioita koskevia ominaisuustietoja yhdelle riville. (12, kohta 4.1.2.)

PostGIS-järjestelmässä on vektorigeometrioille tietotyypit geometry ja geography. Ero tyyppien välillä on se, että geography-tietotyyppi tukee vain WGS84-koordinaattijärjestelmää (SRID 4329) ja -projektiota, kun taas geometry-tietotyyppi voi olla missä tahansa projektiossa. Tämän johdosta kaikki geography-tyypin laskennat tehdään pallomaisessa koordinaatistossa, kun taas geometry-tyyppiset tehdään

kartesiolaisessa koordinaatistossa. Tätä on havainnollistettu kuvassa 3. PostGIS-tietokantaa varten on kehitteillä myös rasterituki. (14.)

4.2 PostGIS-tietokannan käyttämät paikkatiedon tyypit



KUVA 3: Geometry- ja geography-laskennan erot mitattaessa suorinta tietä Pariisin ja Los Angelesin lentokenttien välillä. Kaareva viiva edustaa geography-tyyppiä, kun taas suora viiva on geometry-tyyppinen, tasossa laskettu suorin tie. (14)

5 DIGIROAD-AINEISTON KÄYTTÖÖNOTTO

Luvussa 5 käydään läpi Digiroad-aineiston tuonti paikkatietokantaan, muokkaaminen ja käyttöönotto tietokannan laskemisessa.

5.1 Aineiston tuominen tietokantaan

ESRI shapefile- ja xBase-formaatissa oleville aineistoille on olemassa avoimen lähdekoodin muunnosohjelmat useisiin avoimiin tiedostoformaatteihin. Työssä käytettiin shp2pgsql-ohjelmaa muuntamaan geometrioita sisältävät SHP-tiedostot ja ominaisuudet sisältävät DBF-tiedostot. Liitteessä 1 esiteltyt komentosarjat 2 ja 6 muunsivat tiedostojen sisällön käyttäen shp2pgsql-ohjelmaa PostgreSQL-tietokantaohjelmiston käyttämän SQL-kielen INSERT-murteen lausekkeiksi.

Aineiston muuntamiseen on myös muita vaihtoehtoja, joista osa on nopeampia. Esimerkiksi DBF-tiedostojen muuntamiseen on tarjolla Linux-järjestelmissä dbfdump-ohjelma, joka tuottaa CSV-formaatissa olevan tiedoston. CSV-tiedostojen tuominen on 2–10 kertaa nopeampaa kuin edellä mainittujen shp2pgsql-ohjelman tuottamien tiedostojen tuonti tietokantaan.

Jotta Digiroad-aineisto voidaan tuoda oikein luotuun PostGIS-tietokantaan (ks. liite 1, tiedosto 4), tulee määrittää SRID Spatial_Ref_System-taulusta, jonka tulee vastata aineiston mukana tulleessa PRJ-tiedostossa olevaa tietoa. Käytössä olleessa PostGIS-versiossa SRID on 4258 (ETRS89). Tietojen tuominen tietokantaan on esitetty liitteen 1 komentosarjoissa 5 ja 7.

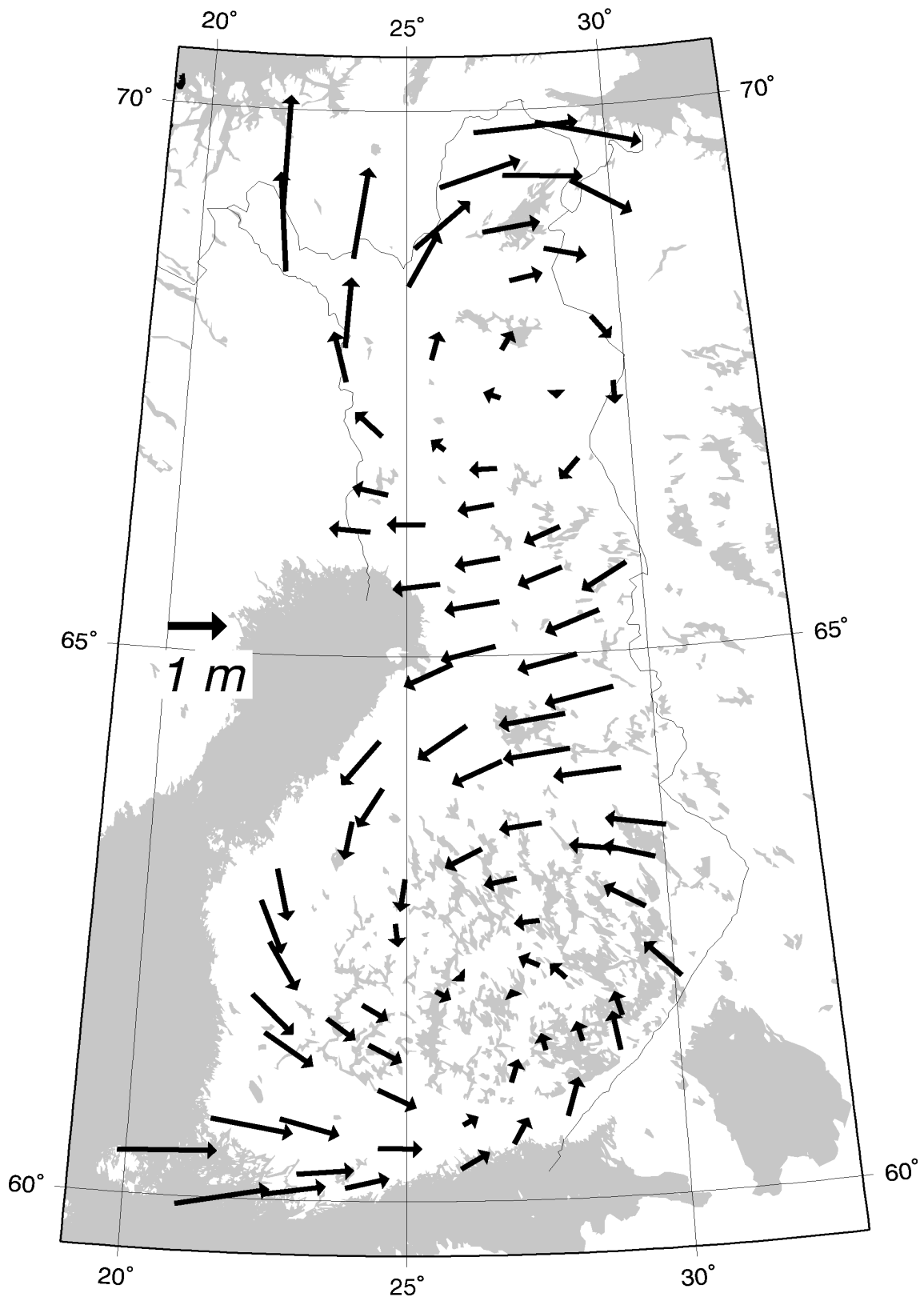
5.2 Aineiston muuntaminen koordinaattijärjestelmien välillä

SRID 4258 -aineisto käyttää mittayksikkönä desimaalimetria. PostGIS käyttää SRID:n määrittelemää mittayksikköä kaikissa laskutoimituksissaan. Koska harvempi ihminen tuntee olonsa kotoisaksi asteitten kanssa, mutta osaa tulkita metriset etäisyydet,

muutettiin tiedot metrejä käyttävään koordinaattijärjestelmään. Tämä voidaan perustella sillä, että tieosoitteessa käytetään metrejä, onhan se muotoa tienumero, tieosanumero ja etäisyys tieosan alusta metreinä. Aineiston etukäteisellä muuntamisella vältetään ajonaikaiselta geometrioiden muunnokselta asteista metriseen järjestelmään. Metrejä käytettäviä järjestelmiä ovat Suomessa KKJ3 eli YKJ (SRID 2393) ja TM35FIN (SRID 3067). Työssä katsottiin parhaaksi käyttää KKJ:tä (ks. liite 1, kommentisarja 8).

Kaikki paikkageometrioita käyttävät taulut on syytä muuntaa samaan koordinaattijärjestelmään, koska järjestelmässä ei voi tehdä vertailua koordinaattijärjestelmissä olevien geometrioiden välillä. PostGIS tarjoaa muunnokseen `st_transform(geom, srid)`-funktion, joka käyttää PROJ.4-kirjastoa. Kirjaston muunnosmenetelmä aiheuttaa kuitenkin 1–5 metrin virheen (ks. kuva 4), koska kirjasto ei käytä Maanmittauslaitoksen mittaamaa kolmioverkkoa affiiniseen muunnokseen. Affinisessa muunnoksessa siirrytään koordinaattijärjestelmästä toiseen niin, että muutos on eri koordinaattiakselien suhteen.

Käytettävän koordinaattijärjestelmän valinnassa tulee ottaa huomioon järjestelmän kattama alue, eri projektioiden aiheuttamat vääristymät esimerkiksi etäisyyteen tai pisteiden väliseen kulmaan ja aineiston käyttötarkoitus. Opinnäytetyössä on käytetty tekohetkellä yleistä KKJ zone 3 -projektiota, kun on tarvittu metrinen mitta etäisyyksille, vaikka muunnos WGS84:stä KKJ3:een ei ole tarkka. KKJ3:een päädyttiin, koska vertailuaineisto oli samassa projektiossa. Maanmittauslaitos suosittelee käytettäväksi tietopalvelusovelluksissa ETRS-TM35FIN-karttaprojektiota ja -tasokoordinaatistoa.



KUVA 4: Euref-FIN- ja KKJ-järjestelmien välillä tapahtuvan muunnoksen jäännösvirheet (13.).

5.3 Aineiston laadun tarkistaminen ja korjaaminen

Aineistosta on tarpeellista poistaa kaksinkertaiset oid_tunnus-tiedot segmenttitaulusta sekä ketjutaulusta. Lisäksi on suositeltavaa tarkistaa aineiston geometriat käyttämällä PostGIS-järjestelmän tarjoamaa `st_isValid(the_geom)`-funktioita, jolla voidaan tarkistaa, että geometria on sääntöjen mukainen.

5.4 Geometrioiden luominen R-muodon lineaarisesti referoidusta tiedosta

Digiroad R -muodossa segmenttitaulun tiedot eivät sisällä geometrioita ja täten suoraan käytettävää paikkatietoa, vaan pelkästään ominaisuustietoja. Mikäli halutaan etsiä tietoa paikkatiedon avulla segmenttitaulun sisältämästä aineistosta, on se ensin muunnettava geometrioiksi. Segmenttiaineisto muutetaan liitteen 1 komentosarjan 11 avulla geometrioiksi käyttämällä alku- ja loppupistearvoja ja segmentti-aulussa olevaa `ketju_oid`-viitettä ketju-aulussa olevaan geometriaan (ks. myös liite 2, tiedosto 2).

Geometrian viitteen avulla käytetään PostGIS:n `st_between_measurements(geom,alkupiste,loppupiste)`-funktioita. Alku- ja loppupiste eivät ole mitattuja pituuksia tai arvoja, vaan ne ovat m-arvoja. Digiroad-aineiston tapauksessa ne ovat referenssikohtaisia suhdelukuja referenssiketjulla. M-arvolla voidaan myös esittää mitattua arvoa esimerkiksi pistegeometrialla, joka esittää kaivoa (kaivon syvyys tai veden määrä). Digiroad-aineiston spesifikaatio on tarkka mainitessaan, että m-arvot eivät ole mitattua metrisiä arvoja vaan suhdelukuja. (7, s. 10.)

5.5 Referoitujen geometroiden puhdistaminen

PostGIS-järjestelmässä Digiroad R -toimitusmuodon käyttöönotto vaatii käytännössä aineiston muuntamista Digiroad K -muotoon. Tämä tapahtuu lineaarisesti referoimalla (ks. liite 1, tiedosto 11). Muunnoksesta syntyy epävalideja geometrioita, jotka pitää puhdistaa, jotta aineistoa voidaan jatkomuokata käytettäväksi (ks. liite 2, tiedosto 3).

Vuosien 2006 ja 2008 Digiroad R -aineisto muokattiin PostGIS-järjestelmässä ja puhdistettiin. Aineistoja verrattiin tämän jälkeen samojen vuosien Digiroad K -aineistoihin. Kävi ilmi, että geometrioiden virheet ovat samanlaisia ja samoissa kohdissa. Tästä voidaan tehdä johtopäätös, että aineiston muunnoksessa ei ole järjestelmäkohtaisia eroja.

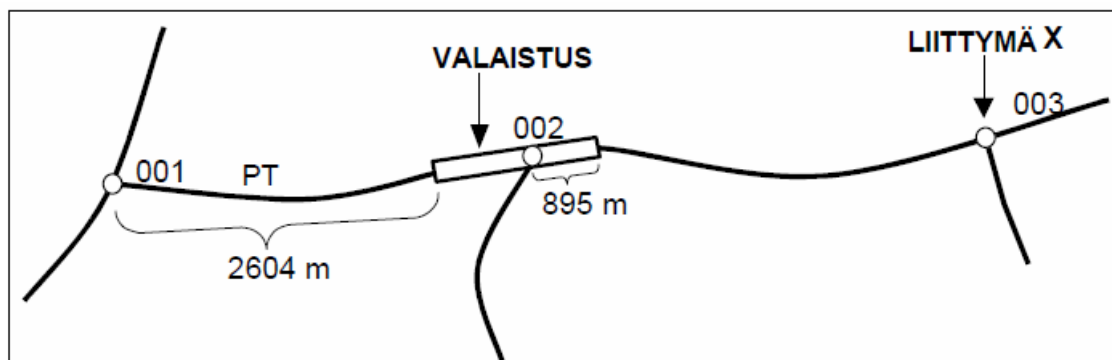
Jokaisesta sarakkeesta referoidulle tiedolle pitää saada LINESTRING-tyyppiä oleva geometria, jotta tietoa voidaan käyttää. Mikäli lineaarinen referointi ei tuota LINESTRING-tyyppiä, voidaan sanoa, että digitoinnissa on tapahtunut virhe, koska tällöin saadut MULTILINESTRING-tyypit eivät sisällä vain yhtä alku- ja loppupistettä tieosalle.

6 TIEOSOITTEIDEN GEOMETRIOIDEN LUOMINEN

6.1 Tieosoite ja tieosa

Tieosoite on yksiselitteinen, metrin tarkkuudella ilmaistu kohta Tiehallinnon hallinnoimalla tiellä. Se ei ole sama asia kuin lähi-, katu- tai postiosoite. Tieosoite koostuu tien numerosta, tieosan numerosta, ajoradan numerosta ja numeroarvosta, joka kuvaa metreissä etäisyyttä kyseessä olevan tieosan alkuun. Ajorataa ei merkitä aina. Tässä työssä sillä ei ole ollut relevanssia. Yleensä tieosa alkaa risteyksestä ja päättyy risteykseen. Samalla tiellä on kuitenkin useita risteyksiä, joten tie jakaantuu useisiin tieosiin. Tiet on jaettu osiin ”sen vuoksi, ettei kohteen sijaintia tarvitse mitata koko tien alusta”. (15, s. 3, 5.)

Tieosoitteen määrittämisestä on annettu havainnollistava esimerkki kuvassa 5. Tieosoitteen eri osat voidaan erottaa välilyönnillä tai kauttaviivalla. Varsinaisen määritelmän mukaan tieosa ilmaistaan aina kolmella numerolla alkaen 001:stä ja etäisyys-arvo viidellä numerolla, mutta tässä työssä on usein supistettu edeltävät nollat pois. On huomattava, että tieosanumerointi ei ole aina jatkuva. (15, s. 5.)



KUVA 5: Kuvassa on tie 11283, josta näkyy kolme tieosaa. Valaistuksen tieosoite määritellään tienumeron, tieosanumeroiden ja mainitun etäisyyden perusteella (kaistamerkintä on tarpeeton): 11283 001 02604 – 002 00895. Tieosoitteen eri osat voidaan myös erottaa kauttaviivalla, jolloin liittymän X tieosoite voidaan ilmaista muodossa 11283/003. (15, s. 3.)

6.2 Tieosoite Digiroad-aineiston segmenttitaulussa

Digiroad-aineistossa tieosoitteeseen viittaavat tiedot ovat tietokannan segmenttitaulussa. Taulu jakaantuu riveihin ja sarakkeisiin. Taulun tyyppi-sarakkeen arvo 2 ilmaisee, että kyseessä on tieosoitteeseen viittaava rivi. Työn tavoitteen kannalta se on olennainen. Jokaisella tieosoitteeseen viittaavalla rivillä – eli sellaisilla rivillä, joilla tyyppi-sarakkeen arvo on 2 – on arvot tienumerolle, tieosanumerolle ja ajoratanumerolle. (7, s. 26.) Etäisyys tieosan alusta tulee laskea erikseen.

Rivi ei välttämättä ilmaise koko tieosageometriaa. Tieosoitteen viimeisen vaaditun arvon eli etäisyys-arvon laskemiseksi tarvitaan koko tieosan sisältävä geometria. Segmenttitaulussa voi kuitenkin olla vaikkapa 20 riviä, jotka koskevat yhtä ja samaa tieosaa. Segmenttitaulu ei sisällä suoraan koko tieosan geometriaa, vaan se sisältää tieosan osageometriat, jotka pitää yhdistää yhdeksi, tieosaa kuvaavaksi viivageometriaksi (LINESTRING).

6.3 Tieosan osageometrioiden ominaisuudet

Digiroad R -aineiston segmentti-taulun muuntaminen paikkatiedoksi lineaarisen referoinnin avulla tuottaa tieosalle vaihtelevan määrän MULTILINESTRING-geometriatyyppiä olevia geometrioita. Käytännössä nämä geometriat ovat kuitenkin LINESTRING-tyyppiä, joten ne voidaan ja tulee muuntaa kyseiseksi tyyppiä. LINESTRING-tyyppi takaa sen, että kyseessä on yksi viiva; MULTILINESTRING-tyyppiä oleva geometria voi sisältää lukuisia viivoja. Esimerkiksi tyyppi voi sisältää yhden viivan Oulussa ja yhden viivan Helsingissä. Jokaisesta tieosasta osageometrioineen tulee pystyä muodostamaan oma, yksiselitteinen viivageometria (LINESTRING), joka kattaa koko tieosan, jotta sillä pystytään tekemään etäisyyttä koskevia laskutoimituksia.

Segmentti-taulussa on sarake ”vaikutussu”, jonka arvo määrittää, onko geometrian alkupiste sen digitoitu alkupiste vai digitoitu loppupiste. Sarakkeen arvo 3 ilmaisee vaikutussuunnan olevan digitointisuuntaa vastaan. Ennen yhdistämisen aloittamista pitää tällaiset geometriat kääntää `st_reverse(geom)`-funktiolla. Jokaisen geometrian, jota yritetään yhdistää, tulee olla PostGIS-järjestelmän funktion `st_isValid(geom)` mukaan validi. Muutoin yhdistäminen ei onnistu.

Tieosoitteiston aineiston lineaarinen referointi ja näiden MULTILINESTRING-geometrioiden muuntaminen LINESTRING-geometrioiksi tuottaa n. 83 000 LINESTRING-geometriaa ja kaksi virheellistä geometria, joiden kohdalla `st_isValid(geom)`-funktio palauttaa arvon FALSE. (Tulokset on ilmaistu taulukossa 2.)

TAULUKKO 2: Tieosoitetaulun lineaarisen referoinnin tuottamien MULTILINESTRING-geometrioiden LINESTRING-geometrioiksi muuntamisen tulokset.

"LINESTRING"	83 088
"<NULL>"	2

Aineiston mainitut virheet ovat kaksi pistegeometriaa, jotka ovat Ahvenmaalla tieosoitteissa 569/3 ja 19299/2. Digiroad-aineiston dokumentoinnissa ei ole mainintaa edellä mainittujen kohteiden erityisominaisuuksista. Päinvastoin, tietolajien kuvauksessa todetaan, että mainittujen geometrioiden tulisi olla viivoja (7, s. 26), joten on selvää, että nämä pistegeometriat ovat aineiston virheitä.

6.4 Tieosan osageometrioiden yhdistäminen tieosageometrioiksi

PostGIS-järjestelmässä funktio `st_linemerge(geom,geom)` tuottaa LINESTRING-geometrian, mikäli sen argumentteina olevista geometrioista voidaan muodostaa yksi yksiselitteinen, alku- ja loppupisteen omaava viivageometria. Segmentti-taulussa on ilmastu ajoratanumero sarakkeessa ”ajoratanum”. Ajoratanumero määrittää, onko

kyseessä digitointisuunnassa ensimmäinen ajorata tienumerosuunnassa oikealta (arvo 1), toinen ajorata tienumerosuunnassa oikealta (arvo 2) vai yksiajoratainen yleinen tie (arvo 9). Arvo 0 ei ole spesifikaation mukainen ajoratanumero. (7, s. 26.)

Tieosan osageometriat yhdistetään tieosan geometrioiksi yksinkertaisella SQL:n SELECT-lausekkeella, jossa muodostetaan yhdelle tieosalle kaksi geometriaa, ensimmäinen ajoratanumeroiden 1 ja 9 mukaan ja toinen ajoratanumeroiden 2 ja 9 mukaan.

Ongelmana ajorata-arvon perusteella tehtävässä yhdistämisessä on aineiston laatu. Vuoden 2006 aineistossa n. 4 200 segmentti-tilin sisältämää tieosoitteen osageometriaa (tyyppi 2) ei sisällä ajoratanumeroa, eli toisin sanoen ajoratanumero on 0 tai NULL, mitä spesifikaatio ei tunne. Vuoden 2008 aineistossa vastaava määrä on n. 3 600. Tarkemmat lukumäärät vuoden 2008 aineistosta on annettu taulukossa 3 ja taulukossa 4.

TAULUKKO 3: Ajoratanumeroiden esiintyminen vuoden 2008 Digiroad-aineiston segmentti-tilillä.

Ajoratanumero	lukumäärä
0	3 614
1	1 713
2	1 873
9	75 889
Yhteensä	83 090

TAULUKKO 4: Ajoratanumeroiden jakaantuminen vuoden 2008 Digiroad-aineistossa erillisille tieosille. On huomattava, että ajoratanumerolla 1 tulisi olla yhtä monta tieosaa kuin ajoratanumerolla 2. Eriävä numero osoittaa, että aineistossa on virhe.

Yksittäisiä tieosia ajoratanumeroiden mukaan	lukumäärä
0	1 047
1	438
2	427
9	17 435

Luotujen geometrioiden oikeellisuuden tarkistaminen on hankalaa, koska aineistossa ei ole tietoa, kuinka pitkiä kunkin tieosan geometrioiden tulisi olla, yksiselitteisesti mainittuja alku- ja loppupisteitä, tai muuta kenttää, johon aineistoa voisi verrata.

Geometrioiden oikeellisuuden tarkistamiseen on mahdollista käyttää muodostuksessa käytettyjen geometrioiden yhteenlasketun ja muodostetun geometrian pituuden erotusta. Mitä suurempi erotus, sitä suurempi virhe todennäköisesti on tapahtunut yhdistämisessä.

6.5 Yhdistettyjen tieosageometrioiden tyypit

Ajoratanumeroiden perusteella tapahtuva yhdistäminen aikaansaa jokaista tieosaa kohden kahden LINESTRING-tyyppisen tieosageometrian syntymisen. Tieosageometriat voidaan jakaa neljään päätyyppiin:

1) Tieosageometria, jolla on vain yksi alku- ja loppupiste. Syntyneet tieosageometriat ovat identtisiä ja päällekkäisiä, koska tieosalla esiintyy vain ajoratanumeroa 9. Näistä tieosageometrioista toinen voidaan poistaa tarpeettomana.

2) Tieosageometria, jolla on kaksi alku- ja kaksi loppupistettä, eivätkä näiden välissä olevat viivageometriat leikkaa toisiaan. Tieosalla ei esiinny ajoratanumeroa 9, vaan sekä arvoa 1 että arvoa 2. Tällaiset tieosageometriat voivat syntyä esimerkiksi moottoritien yhteydessä. Tästä on annettu esimerkki kuvassa 6.



KUVA 6: Oulun läpi kulkeva moottoritie (Pohjantie, tie 4), Kainuuntien (22) ja Poikkimaantien (8155) välissä oleva tieosa (410) kokonaisuudessaan molemmilla ajoradoilla. Vasemmanpuoleisessa kuvassa Digiroad-aineistosta QGIS-ohjelmalla renderoituna, oikeanpuoleisessa kuvassa Google Maps -palvelusta.

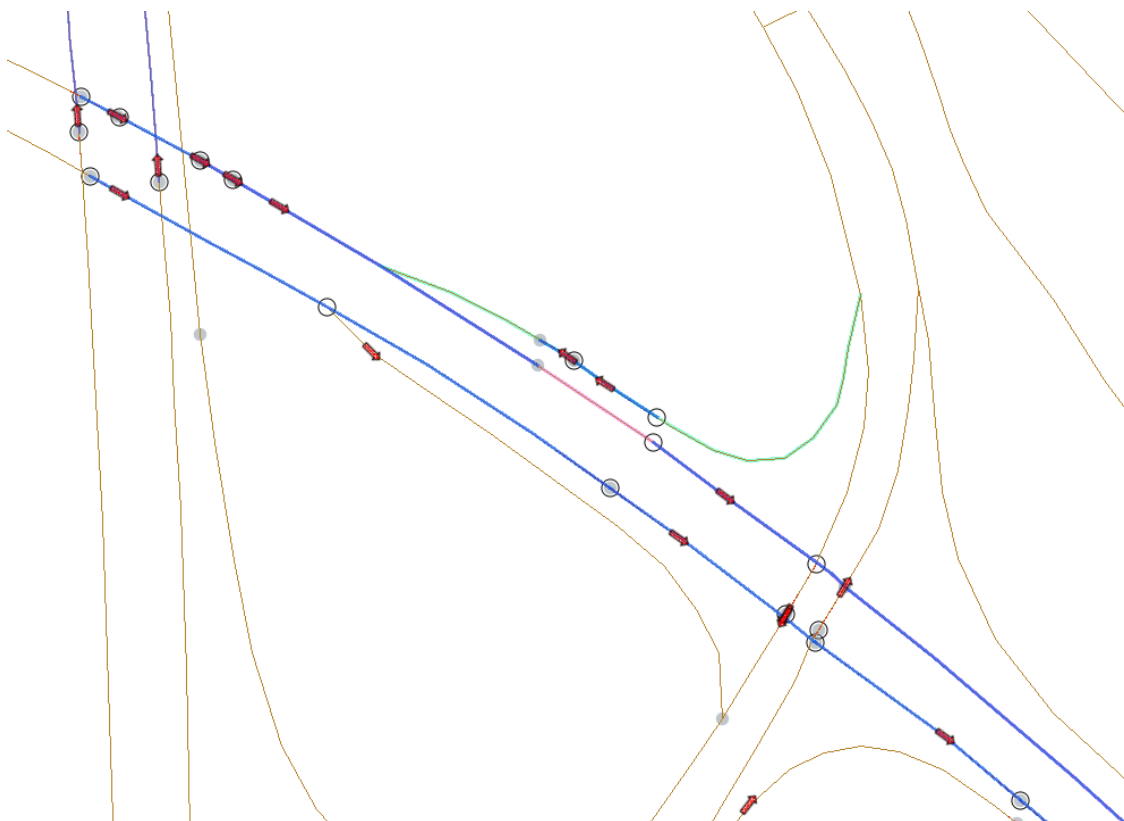
3) Tieosageometria, jolla on kaksi alkupistettä ja yksi loppupiste, tai kaksi loppupistettä ja yksi alkupiste. Tieosageometriat ovat osin päällekkäin. Tieosalla esiintyy tällöin sekä ajoratanumeroa 9 että arvoja 1 ja 2.

4) Tieosa, jolla on yksi alku- ja loppupiste, mutta joka jakaantuu välillä kahdeksi ajoradaksi, mutta palaa tämän jälkeen yhdeksi. Tieosalla esiintyy tällöin sekä ajoratanumeroa 9 että arvoja 1 ja 2. Esimerkiksi tilanne, jossa tieosalla on liikenneympyrä.

6.6 Yhdistämisen ongelmat

Aineiston laatu aiheuttaa yhdistämisessä ongelmia. Esimerkiksi kuvan 7 tapauksessa tieosalle pitäisi tulla kaksi LINESTRING-viivageometriaa, ensimmäinen

ajoratanumeroiden 1 ja 9 ja toinen ajoratanumeroiden 2 ja 9 perusteella. Geometriassa on kuitenkin keskellä epäjatkuvuuskohta, joka tekee mahdottomaksi sen muuntamisen LINESTRING-tyypiksi. Epäjatkuvuus johtuu luultavasti siitä, että tieosan osa on siirtynyt väärälle referenssiketjulle. Täten lineaarinen referointi on kohdistunut väärän ketjun geometrialle.



Kuva 7: Tieosa 22/2:n alku, renderoitu vuoden 2008 Digiroad-aineistosta QGIS-ohjelmalla ja muokattu Microsoftin Paint-ohjelmalla. Punaiset nuolet osoittavat digitointisuuntia ja sininen viiva ilmaisee käsiteltävää tieosageometriaa. Kuvan keskellä on esimerkki aineiston virheestä punaisella viivalla, joka kuvaa epäjatkuvuuskohtaa.

Esimerkkitapauksen kaltaista virhettä on mahdollista yrittää korjata käyttäen alku- ja loppupisteiden paikkatietoa ja niiden keskinäistä suhdetta; sen sijaan, että yhdistäminen tapahtuisi ajoratanumeron perusteella, voidaan geometriaa yrittää luoda sen perusteella, mikä osageometria jatkuu edellisen lopusta. Toinen mahdollisuus olisi venyttää viivageometrioita, joilla ei ole kummassakaan päässä toista, samalle tieosalle kuuluvaa viivageometriaa, ja näin pyrkiä löytämään tieosan seuraava osageometria. Kolmantena

mahdollisuutena olisi kaikkien viivageometrioiden bufferointi esimerkiksi 20 metrillä; saadun aluegeometrian keskilinjaviivaa käytettäisiin tieosoitteen geometriana.

Aineiston korjausmenetelmät aiheuttavat tarkkuuden menetyksen. Aineistossa ei ole tarjolla tietoa, jolla voisi tutkia aineiston muokkauksen aiheuttamia muutoksia tieosoitteen kannalta, joten geometrioita muuttavat ratkaisut todettiin tarkoitukseen sopimattomiksi.

7 YHDISTÄMISEN TULOKSET

7.1 Aineiston muuntaminen ja laatu

Käytössä oli vuoden 2006 ja vuoden 2008 versiot Digiroad R -materiaalista. Kumpikin aineisto sisältää jonkin verran samalla ainutlaatuisella tunnisteella olevia rivejä ketju-, liikenne_elementti- ja segmentti-tauluissa. Tämä aiheuttaa ongelmia, mikäli duplikaatteja ei huomaa poistaa.

Digiroad-aineiston muuntaminen ja tuominen PostGIS-ympäristöön on apuohjelmia hyväksi käyttäen jokseenkin triviaalia, mutta aikaa vievää. Muunnoksessa pitää ottaa huomioon, että UTF-8-merkistö on tällä hetkellä oletusmerkistönä kaikissa Linux-jakeluissa. Digiroad-aineisto käyttää ISO-8859-1-merkistöä. Aineisto tulee siis muuntaa UTF-8-merkistölle.

PostGIS-järjestelmässä Digiroad R -aineiston muuntaminen lineaarisesti refereroidusta aineistosta tieosageometrioiksi on ollut mahdollista versiosta 1.1.0 lähtien. Vertailtaessa Digiroad K -aineiston esimerkkiedoston sisältöä ja Digiroad R -aineiston muunnoksessa tuotettua tieosoiteaineistoa ovat ns. virheelliset (eli epäjatkuvat tieosageometrit) geometriat samoissa paikoissa. Siten voidaan päätellä, että epäjatkuvat tieosat eivät synny muunnoksessa, vaan ovat aineiston virheitä.

7.2 Tieosien geometrioiden luominen

Tieosien geometriat luotiin yhdistämällä kunkin tieosan osageometriat. Aineiston mukaan Suomen tiestössä on yhteensä 18 162 tieosaa. Aineiston huonon laadun vuoksi kaikista näistä ei kuitenkaan saada generoitua geometriaa. Yhdistäminen tuotti vuoden 2008 aineistossa taulukon 5 mukaisen määrän geometrioita.

TAULUKKO 5: Vuoden 2008 aineistossa yhdistämisen tuottamat tieosageometriat. Kullekin tielle tulee vähintään 2 geometriaa joka kerta. Jos tieosalla on vain 1 geometria, kyseessä on aineiston virhe.

Tieosageometrioita	31 419
2 LINESTRING-geometriaa tieosalla	31 412
1 LINESTRING-geometria tieosalla (ajoratanumeroiden 1 ja 9 sekä 2 ja 9 yhdistämisessä vain toinen puoli on onnistunut)	7
Uniikkeja generoituja tieosageometrioita	15 697

Taulukossa 6 on tarkemmin eritelty yhdistämisen tuloksia ja siinä ilmenneitä virheitä. Tuloksia on havainnollistettu kuvassa 8.

TAULUKKO 6: Vuoden 2008 aineistossa tieosien osageometrioiden yhdistämisen tuottamat tieosageometriat. Merkinnät 1-9 ja 2-9 tarkoittavat ajoratanumeroiden 1 ja 9 sekä 2 ja 9 perusteella tehtyjä yhdistämiä.

Tieosageometrioita 1-9 ja 2-9, LINESTRING	16 383
Tieosageometrioita 1-9 ja 2-9, MULTILINESTRING (epäjatkuvuusvirhe)	1 566
Tieosageometrioita 1-9, LINESTRING	16 098
Tieosageometrioita 1-9, MULTILINESTRING (epäjatkuvuusvirhe)	1 528
Tieosageometrioita 2-9, LINESTRING-	16 082
Tieosageometrioita 2-9, MULTILINESTRING (epäjatkuvuusvirhe)	1 543
Uniikkeja tieosageometrioita 1-9 ja 2-9, LINESTRING	16 124
Tieosageometrioita 1-9 ja 2-9, MULTILINESTRING-tyyppiä; joko 1-9 tai 2-9 puoli on rikki, eivätkä sisällä kaikkia tieosalle kuuluvia geometrioita (epäjatkuvuusvirhe)	427
Uniikkeja tieosageometrioita 1-9 ja 2-9, LINESTRING, ja sisältävät kaikki tieosalle kuuluvat geometriat - käyttökelpoiset tieosageometriat	15 697



KUVA 8: Kuvassa mustalla kaikki alueen tieosoitegeometriat, Vihreällä koko tieosalla virheettömät tieosageometriat 1-9 ja 2-9. Punaisella tieosageometriat 1-9 ja 2-9, jotka eivät käytä kaikkia tieosalle määriteltäviä geometrioita. Kuvan keskellä vihreänä Oulun läpi kulkeva moottoritie, tie 4; näkyvissä tieosa 401.

Yhdistäminen tuottaa noin 86 %:ssa tieosista yhden tai useamman LINESTRING-geometrian, joita voidaan käyttää tarkan tieosoitteen laskemiseen. (Taulukon 6 alin arvo.)

Yhdistäminen ajoratojen perusteella tuottaa aina kaksi geometriaa, myös, jos ajoratanumero on 9, koska tällöin kyseinen ajorata tulee sekä kyselyihin 1-9 että 2-9 mukaan. Näiden kahden geometrian pituuksien vertailu antaa osviittaa aineiston laadusta. Taulukossa 7 on annettu yhdistettyjen geometriaparien pituuserotukset.

TAULUKKO 7: Yhdistämisen tuottamisen geometriaparien pituuksien erotus.

Erotus suurempi kuin $\pm 1\ 000$ m	16
Erotus suurempi kuin ± 100 m	46
Erotus suurempi kuin ± 10 m	132
Erotus suurempi kuin ± 1 m	400
Ei erotusta	30 912

8 TIEOSOITTEEN LASKEMINEN

Menetelmä tieosoitteen laskemiseksi GPS-laitteella mitatussa pisteessä voidaan tiivistää seuraavasti. Ensin muunnetaan mitattu piste vertailuaineiston käyttämään koordinaattijärjestelmään. GPS-laitteen antamana tieto tulee muuntaa desimaaliasteiksi. Lisättäessä GPS:n antama pistegeometria järjestelmään tai käytettäessä järjestelmän tarjoamia funktioita on pisteellä oltava tieto käytetystä koordinaattijärjestelmästä. GPS-laitteet käyttävät WGS84-järjestelmää, jonka SRID on 4236. Tehtäessä vertailuja geometrioiden kesken on geometrioiden oltava samassa koordinaattijärjestelmässä. PostGIS-laajennus tarjoaa mahdollisuuden laskea viivan pituus kaksiulotteisesti tai kolmiulotteisesti. Kolmiulotteinen laskutapa ottaa huomioon käytetyn vertikaalisen datumin sekä litistyneisyyden pitkillä etäisyyksillä. Laskut geometria-objekteilla tehdään annetulla SRID:llä.

Sitten suoritetaan käsiteltävälle pistegeometrialle paikkatietohaku, joka palauttaa sitä lähinnä olevan tieosan geometrian sekä pisteen kyseisellä geometrialla, joka on lähinnä käsiteltävää pistegeometriaa. Tuloksena saadun geometrian ominaisuustiedoista saadaan selville tieosan kaksi ensimmäistä arvoa, tienumero ja tieosanumero, ja viimeinen arvo – etäisyys metreissä tieosan alusta – voidaan laskea.

Etäisyyden laskeminen voidaan toteuttaa käyttäen referenssiarvoa. Löydetylle pisteelle lasketaan referenssiarvoksi suhdeluku (0–1), joka kuvaa pisteen viivageometrian alkuun mitattavan etäisyyden suhdetta viivageometrian koko pituuteen (esim. ks. liite 2, tiedosto 4). Näin on laskettu etäisyys tieosageometrian alkuun, ja tieosoitteen viimeinen arvo on saavutettu. Asetettu tavoite, tieosoitteen laskeminen GPS-pisteessä, tai tarkemmin sanottuna pistettä lähinnä olevassa tieosageometrian pisteessä, on saavutettu.

GPS-pisteessä lasketun tieosoitteen tarkkuudessa päästään Digiroad-aineiston lupaamaan tarkkuuteen (+5 metriä). Itse GPS-laitteen mittaustarkkuus voi aiheuttaa vastaavaa kokoluokkaa olevan virheen. GPS-virheen todennäköisyyttä voi arvioida satelliittien määrästä ja vastaavista laitteen osoittamista arvoista. Kuluttajatasen

laitteella mittauksia tehdessä on todennäköistä, että itse laite aiheuttaa suuremman virheen kuin Digiroad-aineisto. Ammattilaisille suunnatuilla laitteilla on mahdollista päästä jälkikäsittelyn avulla 1 sentin tarkkuuteen, jolloin Digiroad-aineiston viiden metrin virhemarginaali on merkitsevä tekijä. Kuvassa 9 on havainnollistettu GPS-järjestelmän tarkkuutta. (17; 18.)

* horizontal only



Autonomous < 15 m

Real-time < 1-5 m

Post-processed code phase differential 0.5 – 5 m

H-Star Processing < 0.2 – 0.3 m

Post-processed carrier phase differential 0.01 – 0.3 m



KUVA 9: GPS-mittauksen tarkkuus eri menetelmin. (17)

9 YHTEENVETO JA POHDINTA

Työn tarkoituksena oli tutkia Digiroad-paikkatietoaineiston soveltuvuutta tieosan tieosoitteen laskemiseen GPS-pisteessä avoimen lähdekoodin ohjelmistojen avulla. Kehitysympäristö on enemmän kuin riittävä. Itse asiassa se on monilta ominaisuuksiltaan kehittyneempi kuin suurin osa kaupallisista ohjelmista. Avoimen lähdekoodin lähestymistapa on siten jopa suositeltavampi kuin kaupallisten ohjelmien käyttäminen. PostGIS-laajennuksen versio 1.3.3 soveltuu hyvin Digiroad-aineiston käyttöön. Digiroad R -toimitusmuodon vaatima lineaarinen referointi on tuettu PostGIS-ohjelmassa. Linearisessa referoinnissa syntyneet virheet ovat jäljitettävissä pohja-aineiston virheisiin kaikissa tarkastelluissa kohdissa. PostgreSQL-tietokannan suorituskyky oli hyvä käytetyssä kehityskoneessa, joka vastaa vuoden 2006 kuluttajatason pelitietokonetta.

Aineiston käyttöönotto vaati formaatti- ja koordinaattijärjestelmämuunnoksen. Tieosan osageometriat yhdistettiin avainominaisuuksien mukaan tieosia kuvaaviksi geometrioiksi. Aineiston puutteet ja virheet tekivät n. 10 %:ssa tieosageometrioiden luomisen mahdottomaksi. Tieosageometrioista mitattiin etäisyyksiä GPS-pisteestä muodostettuihin pistegeometrioihin. GPS-pistettä lähinnä olevan tiegeometrian löytäminen ja sen etäisyyden laskeminen geometrian alkupisteestä GPS-pistettä lähinnä olevaan pisteeseen vei koko Suomen kattavalla aineistolla 40–120 millisekuntia.

Aineisto on sinällään testatussa ympäristössä tarkkaa. Se soveltuu yksinkertaisen paikkatietopalvelun aineistoksi. Aineiston käyttöönotto tieosoitteen tuotantoympäristössä, jossa pyritään pitämään virheiden määrä alle 5 %:ssa, vaatii resursseja tarkistaa ja korjata aineistoa käsin tai yhdistää Digiroad-aineistoa muuhun aineistoon. Mielestäni Digiroad-aineisto ei sovellu yrityksille, joiden osaamisalue on palveluiden ja sovellusten kehittäminen, ei aineiston tuottaminen. Digiroad-aineisto voi kuitenkin olla hyödyksi paikkatietoaineiston tuottajalle yhdistettynä muuhun aineistoon.

LÄHTEET

1. OpenGIS® Geography Markup Language (GML) Encoding Standard. 2007. Saatavissa: <http://www.opengeospatial.org/standards/gml/>. Lataaminen vaatii lisenssin hyväksymisen. Hakupäivä 1.6.2010.
2. OGC® KML. 2008. Saatavissa: <http://www.opengeospatial.org/standards/kml/>. Lataaminen vaatii lisenssin hyväksymisen. Hakupäivä 1.6.2010.
3. OpenGIS® Implementation Standard for Geographic information - Simple feature access - Part 2: SQL option. 2010. Saatavissa: <http://www.opengeospatial.org/standards/is>. Hakupäivä 1.6.2011.
4. ISO/IEC 13249-3:2006. Information technology -- Database languages -- SQL multimedia and application packages -- Part 3: Spatial. 2006. Saatavissa: http://www.iso.org/iso/catalogue_detail.htm?csnumber=38651. Hakupäivä 1.6.2010.
5. JHS 154 ETRS89 -järjestelmään liittyvät karttaprojektiot, tasokoordinaatit ja karttalehtijako. Julkisen hallinnon suositus 154. 2008. Saatavissa: <http://www.jhs-suositukset.fi/suomi/jhs154>. Hakupäivä 1.6.2010.
6. JHS 163 Suomen korkeusjärjestelmä N2000. Julkisen hallinnon suositus 163. 2007. Saatavissa: <http://www.jhs-suositukset.fi/suomi/jhs163>. Hakupäivä 10.10.2010.
7. Digiroad -- Tietolajien kuvaus. Versio 1.2. 2006. Uusin versio (3.0) saatavissa: http://www.digiroad.fi/dokumentit/fi_FI/dokumentit/. Hakupäivä: 7.7.2007.
8. Maanmittauslaitos. Kartat - Koordinaatit. Saatavissa: <http://www.maanmittauslaitos.fi/kartat/koordinaatit/>. Hakupäivä 1.6.2012.
9. Wikipedia. Vapaa tietosanakirja. 2012. Saatavissa: <http://fi.wikipedia.org>. Hakupäivä 11.7.2012.

10. Wikipedia. Free Encyclopedia. 2012. Saatavissa: <http://en.wikipedia.org>. Hakupäivä 11.7.2012.
11. Obe, Regina & Hsu, Leo. Comparison Between Open Source and Commercial Software. 2008. Saatavissa: http://www.bostongis.com/?content_name=opensource_commercial#108. Hakupäivä 11.11.2011.
12. PostGIS 1.3.6 Manual -- Chapter 4. Using PostGIS. 2009. Saatavissa: <http://postgis.refractions.net/documentation/manual-1.3/ch04.html>. Hakupäivä 20.9.2009.
13. JHS 153 ETRS89-järjestelmän mukaiset koordinaatit Suomessa. Julkisen hallinnon suositus 153. 2008. Saatavissa: <http://www.jhs-suositukset.fi/suomi/jhs153>. Hakupäivä 10.10.2010.
14. Introduction to PostGIS. -- Section 17: Geography. [2010]. Saatavissa: <http://workshops.opengeo.org/postgis-intro/geography.html>. Hakupäivä 21.02.2012.
15. Tiehallinto. Tieräkisteri – Tieosoitejärjestelmä. Helsinki 2009. Saatavissa: http://portal.liikennevirasto.fi/portal/page/portal/f/urakoitsijat_suunnittelijat/konsulteille/tierekisteri/tieosoitejarjestelma.PDF. Hakupäivä 31.3.2010.
16. PostGIS 1.3.6 Manual -- Chapter 6. PostGIS Reference. 2009. Saatavissa: <http://postgis.refractions.net/documentation/manual-1.3/ch06.html>. Hakupäivä 20.10.2010.
17. What is the maximum Theoretical accuracy of GPS? Saatavissa: <http://gis.stackexchange.com/questions/43617/what-is-the-maximum-theoretical-accuracy-of-gps>. Hakupäivä 21.12.2012.
18. GPS Accuracy. Saatavissa: <http://www.gps.gov/systems/gps/performance/accuracy/>. Hakupäivä 29.12.2012.

Liite 1: Aineiston käsittelyyn käytetyt BASH-komentosarjat

Tiedosto 1: 00_prepare_working_dirs.sh

```
# Prepare enviroment
#!/bin/bash
ln -s $PWD /tmp/digi
mkdir digiroad
```

Tiedosto 2: 01-unzip-stuff-here

```
# unzip data
# example assumes that every zip file produces directory
# default with digiroad
# for i in /media/digiroad/*.zip ; do unzip $i -d ./digiroad/ ; done
echo "Example"
echo "for i in /media/digiroad/*.zip ; do unzip $i -d ./digiroad/ ;
done"
echo "Do this manually"
```

Tiedosto 3: 02_convert-shp.sh

```
# creates files for importing geometry tables
# creates table definitions and converts shp files
# change 2010 to needed
#!/bin/bash
convertdata(){
  cd $i
  shp2pgsql -s 4258 -a DIGIROAD_PALVELU.shp 2010palvelu
  >>../shpdata.sql
  shp2pgsql -s 4258 -a DIGIROAD_KETJU.shp 2010ketju >> ../shpdata.sql
  shp2pgsql -s 4258 -a DIGIROAD_LIIKENNE_ELEMENTTI.sh
  2010liikenne_elementti >>../shpdata.sql
  echo "$i ready"
  cd ..
}

# digiroad dir
cd digiroad
for i in * ; do
  if [ -d "$i" ]
  then convertdata $i
  fi
done
cd ..

exit
```

Tiedosto 4: 0_digi_r_dev_create.sh

```
# if databse is not ready
# add password for user postgres first
# sudo -u postgres psql postgres
# \password postgres
# type password
```

```
echo "Change this script if you have not installed postgis from
source"

createdb -U postgres -W digi_r_dev
createlang -U postgres -W plpgsql digi_r_dev
# default install from ubuntu repo 1.3
#psql -d digi_r_dev -h localhost -U postgres -f /usr/share/postgresql-
8.3-postgis/lwpostgis.sql
#psql -d digi_r_dev -h localhost -U postgres -f /usr/share/postgresql-
8.3-postgis/spatial_ref_sys.sql

## default install from postgis 1.4 source
psql -d digi_r_dev -h localhost -U postgres -f
/usr/share/postgresql/8.4/contrib/postgis.sql
psql -d digi_r_dev -h localhost -U postgres -f
/usr/share/postgresql/8.4/contrib/spatial_ref_sys.sql
# add comments to functions
psql -d digi_r_dev -h localhost -U postgres -f
/usr/share/postgresql/8.4/contrib/postgis_comments.sql

sh create_tables.sh
psql -d digi_r_dev -h localhost -U postgres -f
./digiroad/digi_tables.sql.info
psql -d digi_r_dev -h localhost -U postgres -f
./sql/support_function.sql
```

Tiedosto 5: 04-insert-shp.sh

```
# imports geometry tables
#!/bin/bash
psql -d digi_r_dev -h localhost -U postgres -f ./digiroad/shpdata.sql
```

Tiedosto 6: 05-convert-dbf.sh

```
# script for converting lone dbf files
#!/bin/bash

# this is leftover from dbfdump conversoin, alot faster to import
# using SQL statement COPY
convertdata_original(){
  cd $i
  dbfdump DIGIROAD_KAANTYMISMAARAYS.dbf >> ../km.csv
  dbfdump DIGIROAD_KAANTYMISMAARAYS_ELEMENTTI.dbf >>../kme.csv
  dbfdump DIGIROAD_SEGMENTTI_KAANTYMISMAARAYS.dbf >>../seqk.csv
  dbfdump DIGIROAD_SEGMENTTI.dbf >>../seq.csv
  dbfdump --fs ';' DIGIROAD_NIMI.dbf | recode ISO_8859-1..utf8
  >>../nimi.csv
  dbfdump --fs ',' DIGIROAD_OPASTUSTAULUN_INFORMAATIO.dbf | recode
  ISO_8859-1..utf8 >>../op.csv
  cd ..
}

# DBF convert using shp2sql, usable after 1.4 postgis tools
convertdata(){
  cd $i
  shp2pgsql -n -a DIGIROAD_KAANTYMISMAARAYS.dbf 2010kaantymismaarays >>
  ../dbfdata.sql
  shp2pgsql -n -a DIGIROAD_KAANTYMISMAARAYS_ELEMENTTI.dbf
```



```
2010kaantymismaarays_elementti >>../dbfdata.sql
shp2pgsql -n -a DIGIROAD_SEGMENTTI_KAANTYMISMAARAYS.dbf
2010segmentti_kaantymismaarays >>../dbfdata.sql
shp2pgsql -n -a DIGIROAD_SEGMENTTI.dbf 2010segmentti >>../dbfdata.sql
shp2pgsql -n -a DIGIROAD_NIMI.dbf 2010nimi | recode ISO_8859-1..utf8
>>../dbfdata.sql
shp2pgsql -n -a DIGIROAD_OPASTUSTAULUN_INFORMAATIO.dbf
2010opastustaulun_informaatio | recode ISO_8859-1..utf8
>>../dbfdata.sql

cd ..

}
# main
cd digiroad
for i in $(ls) ; do
    if [ -d "$i" ]
    then convertdata $i
    else echo ""
    fi
done
cd ..

exit
```

Tiedosto 7: 06-insert-dbf.sh

```
#!/bin/bash
# When using shp2psql we dont get .csv files
# psql -d digi_r_dev -h localhost -U postgres -f ./sql/insert-csv.sql
# this one for shp2psql converted .dbf files
# CSV import is 10x faster, but requires dbfdump or similiar program
to convert tables

psql -d digi_r_dev -h localhost -U postgres -f ./digiroad/dbfdata.sql
```

Tiedosto 8: 08_convert_to_kkj.sh

```
# convert_to_kkj.sh
# reprojects geoms to kkj (srid 2393)
#!/bin/bash
psql -h localhost -d digi_r_dev -U postgres -f
./sql/create_kkj_tables_and_insert.sql
```

Tiedosto 9: 09_create_linear_referencing_functions.sh

```
# create_linear_referencing_funtions.sh
# Creates neede methods to manipulate linear referenced tables to
geometry tables
#
#!/bin/bash
psql -h localhost -d digi_r_dev -U postgres -f
./sql/linear_ref_func.sql
```

Tiedosto 10: 10_create_roadnumber_tables.sh

```
# create_roadnumber_tables.sh
# Creates roadnumber tables
#!/bin/bash
psql -h localhost -d digi_r_dev -U postgres -f
./sql/create_and_insert_roads_segments.sql
```

Tiedosto 11: 11_create_POINT_lines_and_multilines_from_segmentti_and_segmentti_tiet.sh

```
# linear_referencing.sh
# Does linear referencing
#!/bin/bash
psql -h localhost -d digi_r_dev -U postgres -f
./sql/create_segment_geoms.highmem.sql
```

Tiedosto 12: 16_import_gps.sh

```
#!/bin/bash
cat gps.log | grep GPGGA | awk 'FS ="," {print":"$3":" $5};' >>
./digiroad/gps_import.csv
psql -h localhost -d digi_r_dev -U postgres -f ./sql/gps_import.sql
psql -h localhost -d digi_r_dev -U postgres -f ./sql/gps_to_POINTS.sql
```

Tiedosto 13: 17_import_gps_log_with_heading.sh

```
#!/bin/bash
cat gps.log | grep GPRMC | awk 'FS ="," {print":"$4":" $6 ":" $9} ;'
>> gps_head_import.csv
psql -h localhost -d digi_r_dev -U postgres -f
gps_head_import.sql.info
psql -h localhost -d digi_r_dev -U postgres -f
gps_head_to_POINTS.sql.info
```

Tiedosto 14: clear_working_dir.sh

```
#for i in * ; do
#    if [ -d "$i" ]
#        then rm -rf $i
#    fi
#done
rm *~
rm digi_tables.sql.info
rm shpdata.sql
```

Tiedosto 15: create_tables.sh

```
#!/bin/bash
# Create table definitions for digiroad
# change this and 02 script if you want to rename tables

createtables(){
    cd $i
    shp2pgsql -s 4258 -p DIGIROAD_PALVELU.shp 2010palvelu
>>../digi_tables.sql.info
    shp2pgsql -s 4258 -p DIGIROAD_KETJU.shp 2010ketju >>
    ../digi_tables.sql.info
```

```
shp2pgsql -s 4258 -p DIGIROAD_LIIKENNE_ELEMENTTI.shp
2010liikenne_elementti >>../digi_tables.sql.info
shp2pgsql -n -p DIGIROAD_KAANTYMISMAARAYS.dbf 2010kaantymismaarays >>
../digi_tables.sql.info
shp2pgsql -n -p DIGIROAD_KAANTYMISMAARAYS_ELEMENTTI.dbf
2010kaantymismaarays_elementti >>../digi_tables.sql.info
shp2pgsql -n -p DIGIROAD_SEGMENTTI_KAANTYMISMAARAYS.dbf
2010segmentti_kaantymismaarays >>../digi_tables.sql.info
shp2pgsql -n -p DIGIROAD_SEGMENTTI.dbf 2010segmentti
>>../digi_tables.sql.info
shp2pgsql -n -p DIGIROAD_NIMI.dbf 2010nimi | recode ISO_8859-1..utf8
>>../digi_tables.sql.info
shp2pgsql -n -p DIGIROAD_OPASTUSTAULUN_INFORMAATIO.dbf
2010opastustaulun_informaatio | recode ISO_8859-1..utf8
>>../digi_tables.sql.info
cd ..

}
cd digiroad
for i in * ; do
    if [ -d "$i" ] && [ ! -r "../digi_tables.sql.info" ]
        then createtables $i
    fi
done
cd ..
```


Liite 2: Aineiston käsittelyyn käytettyjä SQL-komentosarjoja

Tiedosto 1: clean_tables.sql

```
select digiroad_check_geoms();
select digiroad_create_index();
```

Tiedosto 2: do_linear_ref.sql

```
-- Tekee lineearisen refereoinnin koko segmentti taululle
select d.viite as viite, d.path as path, geom as the_geom into
segmentti_dlines from (
    select a.oid_tunnus as viite,
    (st_dump(st_locate_between_measures(a.the_geom, a.alkupiste,
a.loppupiste))).* from
    (select s.oid_tunnus, s.alkupiste, s.loppupiste, k.the_geom
FROM segmentti s ,ketju k
    WHERE s.ketju_oid = k.oid_tunnus) AS a) as d WHERE
geometrytype(geom) = 'LINESTRING' ;

ALTER TABLE segmentti ADD COLUMN the_geom geometry ;

update segmentti s set the_geom = a.the_geom from
    ( select st_linemerge(st_union(d.the_geom)) as the_geom, d.viite
        FROM segmentti s2 ,segmentti_dlines d where s2.oid_tunnus =
d.viite group by viite )
    as a
    where a.viite = s.oid_tunnus ;

-- tekee oman taulun jossa vain tieosoite dataa

select * into segmentti_tiet from segmentti WHERE tyyppi = 2 ;

-- spesifikaation mukainen yksinkertainen yhdistäminen tieosoitteille
-- käyttää vain aineistoja joita voi käyttää etäisyyden laskemiseen
alkupisteestä
-- eli LINESTRING tyyppiä olevian geometrioita
select y.the_geom, y.tienumero, y.tieosanume into segmentti_tiet_19
from
    (select st_linemerge(st_memunion(the_geom)) as the_geom ,
tienumero, tieosanume from segmentti_tiet where (ajoratanum = 1 OR
ajoratanum = 9) group by tieosanume, tienumero) y
    where geometrytype(y.the_geom) = 'LINESTRING' ;

select y.the_geom, y.tienumero, y.tieosanume into segmentti_tiet_29
from
    (select st_linemerge(st_memunion(the_geom)) as the_geom ,
tienumero, tieosanume from segmentti_tiet where (ajoratanum = 2 OR
ajoratanum = 9) group by tieosanume, tienumero) y
    where geometrytype(y.the_geom) = 'LINESTRING' ;

-- spesifikaation mukainen yksinkertainen yhdistäminen tieosoitteille
-- laittaa talteen väärää tyyppiä olevat tieosoite geometriat
```

```
-- mahdollista vian etsintää varten

select y.the_geom, y.tienumero, y.tieosanume into segmentti_tiet_19m
  from
    (select st_linemerge(st_memunion(the_geom)) as the_geom ,
     tienumero, tieosanume from segmentti_tiet where (ajoratanum = 1 OR
     ajoratanum = 9) group by tieosanume, tienumero) y
  where geometrytype(y.the_geom) = 'MULTILINESTRING' ;

select y.the_geom, y.tienumero, y.tieosanume into segmentti_tiet_29m
  from
    (select st_linemerge(st_memunion(the_geom)) as the_geom ,
     tienumero, tieosanume from segmentti_tiet where (ajoratanum = 2 OR
     ajoratanum = 9) group by tieosanume, tienumero) y
  where geometrytype(y.the_geom) = 'MULTILINESTRING' ;
```

Tiedosto 3: support_function.sql

```
-- support
-- create gist index for all tables which have geometry column in
-- schema
CREATE OR REPLACE FUNCTION digiroad_create_index() RETURNS boolean AS
$$
DECLARE
  r record ;

BEGIN
  FOR r in select f_table_name as name, f_geometry_column as col
  from geometry_columns
  LOOP
    EXECUTE 'DROP INDEX IF EXISTS ' ||r.name||'_index' ;
    EXECUTE 'CREATE INDEX '
      || r.name ||
      '_index ON '
      || r.name ||
      ' USING GIST ('
      || r.col ||
      ')';
  END LOOP;
  RETURN true;
END;
$$
LANGUAGE 'plpgsql' ;

-- Creates _broken tables and copy all broken geoms to there
CREATE OR REPLACE FUNCTION digiroad_check_geoms() RETURNS boolean AS $
$
DECLARE
  r record ;

BEGIN
  FOR r in select f_table_name as name, f_geometry_column as col
  from geometry_columns
  LOOP
    IF r.name NOT LIKE '%broken'
      THEN EXECUTE 'CREATE TABLE ' ||r.name||'_broken ( LIKE ' ||
r.name||' INCLUDING DEFAULTS INCLUDING INDEXES )' ;
    
```

```

EXECUTE 'INSERT into ' || r.name || '_broken select * from ' ||
r.name || ' WHERE the_geom IS NULL OR st_IsValid(the_geom) IS FALSE ' ;

EXECUTE 'DELETE FROM ' || r.name || ' WHERE the_geom IS NULL
OR st_IsValid(the_geom) IS FALSE ' ;
EXECUTE 'INSERT into ' || r.name || '_broken select * from ' ||
r.name || ' WHERE gid in (SELECT min(gid) from ' || r.name || ' group by
oid_tunnus having count(oid_tunnus) > 1 ) ' ;
EXECUTE 'DELETE FROM ' || r.name || ' WHERE gid IN (SELECT
min(gid) from ' || r.name || ' group by oid_tunnus having
count(oid_tunnus) > 1 ) ' ;
ELSE
RAISE NOTICE 'Its allready broken and duplicates are deleted' ;
END IF ;
END LOOP;
RETURN true;
END;
$$
LANGUAGE 'plpgsql' ;

```

```

-- Create and copy old data into new table with new srid
CREATE OR REPLACE FUNCTION digiroad_prepare_copy_newsrid(newsrid int,
suffix text) RETURNS boolean AS $$
DECLARE
    r record ;

BEGIN

FOR r in select f_table_name as name, f_geometry_column as col
, coord_dimension as dim, type as t from geometry_columns
LOOP
EXECUTE 'CREATE TABLE ' || r.name || '_' || suffix || '( LIKE ' ||
r.name || ' INCLUDING DEFAULTS INCLUDING INDEXES )' ;
EXECUTE 'INSERT INTO ' || r.name || '_' || suffix || ' SELECT *
FROM ' || r.name || ' ' ;
EXECUTE 'UPDATE ' || r.name || '_' || suffix || ' SET the_geom =
st_transform(the_geom , ' || newsrid || ' )';
EXECUTE 'INSERT INTO geometry_columns(f_table_catalog,
f_table_schema, f_table_name, f_geometry_column, coord_dimension,
srid, "type")
SELECT ''', 'public'', '' || r.name || '_' || suffix || '',
'the_geom'', ST_CoordDim(the_geom), ST_SRID(the_geom),
GeometryType(the_geom)
FROM ' || r.name || '_' || suffix || ' LIMIT 1' ;

END LOOP;
RETURN true;
END;
$$
LANGUAGE 'plpgsql' ;

```

```

-- check one specified table and Creates _broken table and moves all
broken geoms to there
CREATE OR REPLACE FUNCTION digiroad_check_geoms2(t text) RETURNS
boolean AS $$
DECLARE
    r record ;

```

```

BEGIN
    EXECUTE 'CREATE TABLE ' || t || '_broken ( LIKE ' || t || ' INCLUDING
DEFAULTS INCLUDING INDEXES )';
    EXECUTE 'INSERT into ' || t || '_broken select * from ' || t || ' WHERE
the_geom IS NULL OR st_IsValid(the_geom) IS FALSE';
    EXECUTE 'DELETE FROM ' || t || ' WHERE the_geom IS NULL OR
st_IsValid(the_geom) IS FALSE';
    EXECUTE 'INSERT into ' || t || '_broken select * from ' || t || ' WHERE
gid in (SELECT min(gid) from ' || t || ' group by oid_tunnus having
count(oid_tunnus) > 1 ) ' ;
    EXECUTE 'DELETE FROM ' || t || ' WHERE gid IN (SELECT min(gid)
from ' || t || ' group by oid_tunnus having count(oid_tunnus) > 1 ) ' ;

    RETURN true;
END;
$$
LANGUAGE 'plpgsql' ;

CREATE OR REPLACE FUNCTION make_lines_segmentti(segment_table
"varchar", ref_geom_table "varchar" )
RETURNS void AS
$BODY$
DECLARE
    r record ;
    lines record;

    segment_table_lines "varchar" := quote_ident(segment_table) ||
'_lines' ;
    x int ;
    i int:=0;
BEGIN
    RAISE NOTICE 'make_lines_t(segmentti table, ketju_table), when
using digiroad R, data must be in srid 2393' ;
    BEGIN
        EXECUTE 'DROP TABLE ' || segment_table_lines || ' ' ;
        EXECUTE 'DROP SEQUENCE ' || quote_ident(segment_table_lines ||
'_gid_seq' ) || ' ' ;
    EXCEPTION
        WHEN UNDEFINED_TABLE THEN
            END;

    EXECUTE 'CREATE SEQUENCE ' || segment_table_lines || '_gid_seq START
1';
    EXECUTE 'CREATE TABLE ' || segment_table_lines || '( gid int
DEFAULT nextval(' || quote_literal(quote_ident(segment_table_lines ||
'_gid_seq' )) || ') , viite int ) ' ;
    EXECUTE 'SELECT addGeometryColumn(' ||
quote_literal(segment_table_lines) || ', ' || quote_literal('the_geom')
|| ', 2393, ' || quote_literal('LINESTRING') || ', 4) ' ;
    EXECUTE ' CREATE INDEX ' || quote_ident(segment_table_lines) ||
'_index' || ' ON ' || quote_ident(segment_table_lines) || ' USING GIST
(the_geom) ' ;
    EXECUTE 'ALTER TABLE ' || quote_ident(segment_table_lines) ||
' ADD PRIMARY KEY (gid)';
    FOR r in EXECUTE 'SELECT * FROM (SELECT
ST_NumGeometries(st_locate_between_measures(k.the_geom, s.alkupiste,
s.loppupiste)) as t FROM '
        || quote_ident(segment_table) || ' s , '

```



```
        || quote_ident(ref_geom_table) || ' k WHERE s.ketju_oid =
k.oid_tunnus ORDER by t DESC limit 1 ) as t ' LOOP
        i=r.t;
END LOOP;

        LOOP
        IF i = 0 THEN EXIT ;
        END IF;
        EXECUTE 'INSERT INTO ' || quote_ident(segment_table_lines) ||
'( viite, the_geom )
        SELECT s.oid_tunnus as viite,
        st_geometryn(st_locate_between_measures(k.the_geom,
s.alkupiste, s.loppupiste), ' || quote_literal(i) || ' )
        as the_geom
        FROM ' || quote_ident(segment_table) || ' s , ' ||
quote_ident(ref_geom_table) || ' k
        WHERE s.ketju_oid = k.oid_tunnus
        AND
geometrytype(st_geometryn(st_locate_between_measures(k.the_geom,
s.alkupiste, s.loppupiste),
        ' || quote_literal(i) || ' )) = 'LINESTRING' ' ' ;

        i:=i-1;

END LOOP;

RETURN;
END;
$BODY$
LANGUAGE 'plpgsql' ;
ALTER FUNCTION make_lines_segmentti(geom_table "varchar",
ref_geom_table "varchar" ) OWNER TO postgres;

CREATE OR REPLACE FUNCTION make_multilines(segment_table "varchar")
RETURNS void AS
$BODY$
DECLARE
        r record ;
        segment_table_lines "varchar" := quote_ident(segment_table) ||
'_multi' ;
        i int:=0;
BEGIN
        BEGIN
        EXECUTE 'DROP TABLE ' || segment_table_lines || ' ' ;
        EXECUTE 'DROP SEQUENCE ' || quote_ident(segment_table_lines) ||
'_gid_seq' ) || ' ' ;
        EXCEPTION
        WHEN UNDEFINED_TABLE THEN
        END;

        EXECUTE 'CREATE SEQUENCE ' || segment_table_lines || '_gid_seq START
1';
        EXECUTE 'CREATE TABLE ' || segment_table_lines || ' ( gid int
DEFAULT nextval(' || quote_literal(quote_ident(segment_table_lines) ||
'_gid_seq' )) || ' ) , viite int ) ' ;
```

```
EXECUTE 'SELECT addGeometryColumn('||
quote_literal(segment_table_lines)|| ', ' || quote_literal('the_geom')
|| ', 2393,' || quote_literal('MULTILINESTRING') || ', 4) ' ;
EXECUTE ' CREATE INDEX ' || quote_ident(segment_table_lines) ||
'_index' || ' ON ' || quote_ident(segment_table_lines) || ' USING GIST
(the_geom) ' ;
EXECUTE 'ALTER TABLE ' || quote_ident(segment_table_lines) || '
ADD PRIMARY KEY (gid)';

EXECUTE 'INSERT INTO ' || quote_ident(segment_table_lines) ||
'( viite, the_geom ) SELECT s.viite as viite, st_collect(s.the_geom)
as the_geom '
|| ' FROM ' || quote_ident(segment_table) || ' s group by
s.viite ' ;

RETURN;
END;
$BODY$
LANGUAGE 'plpgsql' ;
ALTER FUNCTION make_multilines(segment_table "varchar") OWNER TO
postgres;

CREATE OR REPLACE FUNCTION make_points_segmentti(segment_table
"varchar", ref_geom_table "varchar" )
RETURNS void AS
$BODY$
DECLARE
r record ;
segment_table_points "varchar" := quote_ident(segment_table) ||
'_points' ;
x int ;
i int:=0;
BEGIN
RAISE NOTICE 'make_points_t(segmentti table, ketju_table), when
using digiroad R' ;
BEGIN
EXECUTE 'DROP TABLE ' || segment_table_points || ' ' ;
EXECUTE 'DROP SEQUENCE ' || quote_ident(segment_table_points
|| '_gid_seq' ) || ' ' ;
EXCEPTION
WHEN UNDEFINED_TABLE THEN
END;

EXECUTE 'CREATE SEQUENCE '||segment_table_points ||'_gid_seq START
1';
EXECUTE 'CREATE TABLE '|| segment_table_points || '( gid int
DEFAULT nextval('|| quote_literal(quote_ident(segment_table_points) ||
'_gid_seq' )) ||') , viite int )' ;
EXECUTE 'SELECT addGeometryColumn('||
quote_literal(segment_table_points)|| ', ' ||
quote_literal('the_geom') || ', 2393,' || quote_literal('POINT') ||
', 4) ' ;
EXECUTE ' CREATE INDEX ' || quote_ident(segment_table_points) ||
'_index' || ' ON ' || quote_ident(segment_table_points) || ' USING
GIST (the_geom) ' ;
EXECUTE 'ALTER TABLE ' || quote_ident(segment_table_points) ||
' ADD PRIMARY KEY (gid)';
```

```
FOR r in EXECUTE 'SELECT * FROM (SELECT
ST_NumGeometries(st_locate_between_measures(k.the_geom, s.alkupiste,
s.loppupiste)) as t FROM '
    || quote_ident(segment_table) || ' s , '
    || quote_ident(ref_geom_table) || ' k WHERE s.ketju_oid =
k.oid_tunnus ORDER by t DESC limit 1 ) as t ' LOOP
    i=r.t;
END LOOP;

LOOP
IF i = 0 THEN EXIT ;
END IF;
EXECUTE 'INSERT INTO ' || quote_ident(segment_table_points) ||
'( viite, the_geom )
SELECT s.oid_tunnus as viite,
st_geometryn(st_locate_between_measures(k.the_geom, s.alkupiste,
s.loppupiste), '
    || quote_literal(i) || ' ) as the_geom
FROM ' || quote_ident(segment_table) || ' s , ' ||
quote_ident(ref_geom_table) || ' k
WHERE s.ketju_oid = k.oid_tunnus
AND
geometrytype(st_geometryn(st_locate_between_measures(k.the_geom,
s.alkupiste, s.loppupiste),
    || quote_literal(i) || ' )) = 'POINT' ' ;

i:=i-1;

END LOOP;

RETURN;
END;
$BODY$
LANGUAGE 'plpgsql' ;
ALTER FUNCTION make_points_segmentti(geom_table "varchar",
ref_geom_table "varchar" ) OWNER TO postgres;

-- DROP FUNCTION make_lines_long(character varying);

CREATE OR REPLACE FUNCTION make_lines_long(segment_table character
varying)
RETURNS void AS
$BODY$
DECLARE
    r record ;
    segment_table_lines "varchar" := quote_ident(segment_table) ||
'_long' ;
    i int:=0;
BEGIN

BEGIN
EXECUTE 'DROP TABLE ' || segment_table_lines || ' ' ;
EXECUTE 'DROP SEQUENCE ' || quote_ident(segment_table_lines ||
'_gid_seq' ) || ' ' ;
EXCEPTION
```

```
        WHEN UNDEFINED_TABLE THEN
        END;

        EXECUTE 'CREATE SEQUENCE ' || segment_table_lines || '_gid_seq START
1';
        EXECUTE 'CREATE TABLE ' || segment_table_lines || ' ( gid int
DEFAULT nextval(' || quote_literal(quote_ident(segment_table_lines) ||
'_gid_seq' )) || ') , viite int, the_geom geometry )' ;
        -- EXECUTE 'SELECT addGeometryColumn(' ||
quote_literal(segment_table_lines) || ', ' || quote_literal('the_geom')
|| ', 2393, ' || quote_literal('MULTILINESTRING') || ', 4) ' ;
        EXECUTE ' CREATE INDEX ' || quote_ident(segment_table_lines) ||
'_index' || ' ON ' || quote_ident(segment_table_lines) || ' USING GIST
(the_geom) ' ;
        EXECUTE 'ALTER TABLE ' || quote_ident(segment_table_lines) || '
ADD PRIMARY KEY (gid)';

        EXECUTE 'INSERT INTO ' || quote_ident(segment_table_lines) ||
'( viite, the_geom ) SELECT s.viite as viite,
st_linemerge(st_collect(s.the_geom)) as the_geom '
|| ' FROM ' || quote_ident(segment_table) || ' s group by
s.viite ' ;

        RETURN;
END;
$BODY$
LANGUAGE 'plpgsql' VOLATILE
COST 100;
ALTER FUNCTION make_lines_long(character varying) OWNER TO postgres;
```

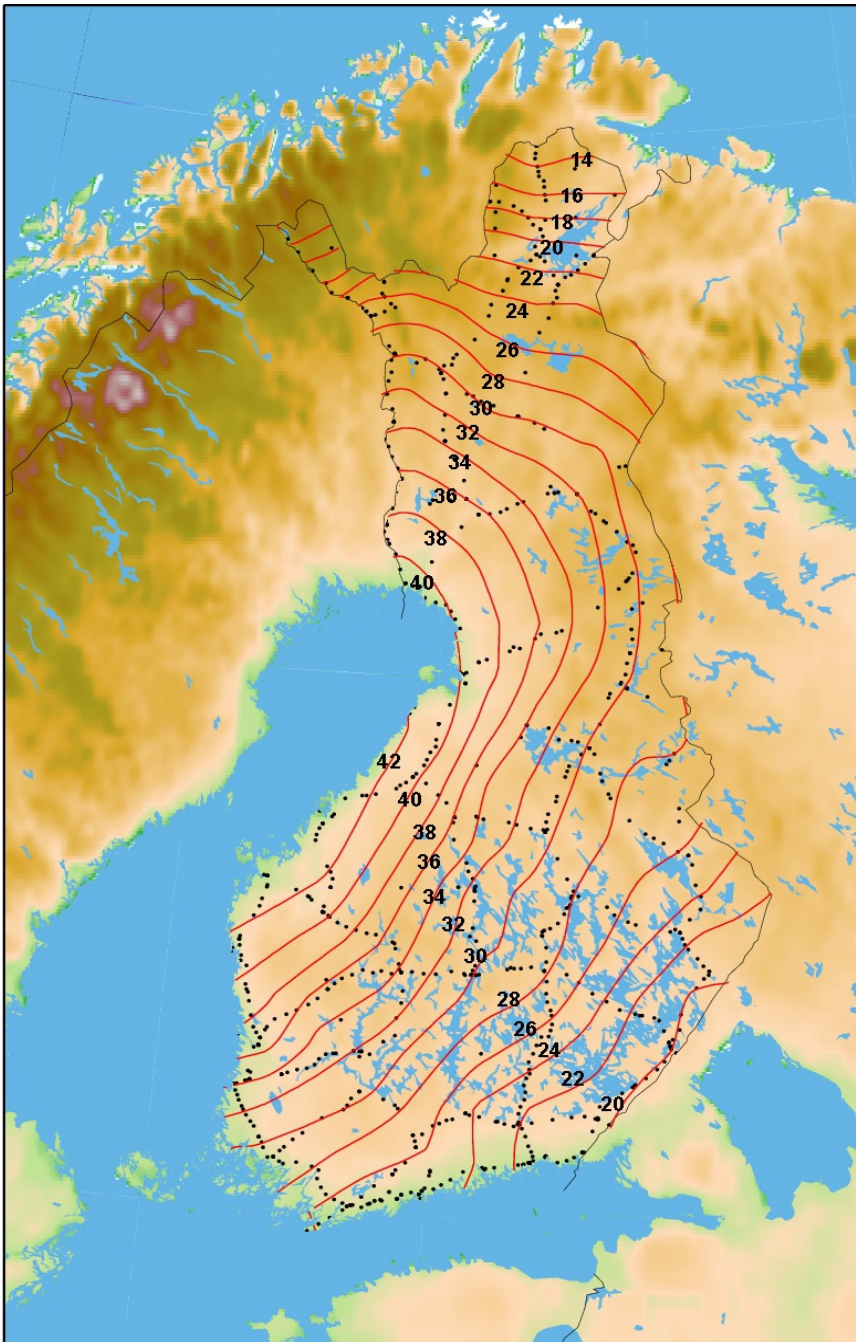
Tiedosto 4: find_nearest_point_on_data.sql

```
select st_distance(g.the_geom , st.the_geom) as distance_meters ,
st_line_locate_point(st.the_geom, g.the_geom) as fractions_2d_lenght
, st_m( st_line_interpolate_point(st.the_geom,
st_line_locate_point(st.the_geom, g.the_geom))) as measured
, st.*, g.* from seg_tiet_op st, gps_points_kkj g where g.gid = 22856
AND st.the_geom && ST_expand(g.the_geom, 2000) order by
distance_meters
```

Liite 3: Havainnollistavia kuvia

1. N60- ja N2000-järjestelmän väliset korkeuserot senttimetreinä

Lähde 6, s. 9.



2. FIN2005-geoidimallin mukaiset geoidinkorkeudet GRS80-ellipsoidista metreinä

Lähde 6, s. 11.

