

# PostGIS 2.0 og Raster

Andreas Forø Tollefsen

*Andreas Forø Tollefsen, «PostGIS 2.0 and Raster»*

KART OG PLAN, Vol. 73, pp. 159–164, POB 5003, NO-1432 Ås, ISSN 0047-3278

This article provides a short discussion of the development of free and open-source software for geospatial analysis (FOSS4G), focusing in particular on raster innovations in PostGIS 2.0. While many previous FOSS4G products had limited functionality as compared to expensive proprietary software alternatives, this has indeed changed. With the introduction of PostGIS 2.0, both vector and raster support are included and the documentation has been improved considerably. This article first introduces some of the major advantages and disadvantages with open-source software. Second, this article shows using spatial raster and vector data, examples of new functionality in PostGIS emphasizing aggregation from raster to vector. I conclude by arguing that FOSS4G has become a real contender to proprietary software alternatives.

*Key words:* GIS, PostGIS, Raster, Vektor, PRIO

*Andreas Forø Tollefsen*, PhD Candidate, Department of Sociology and Human Geography, University of Oslo & Researcher, Peace Research Institute Oslo (PRIO), POB 9229 Grønland, NO-0134 Oslo. E-mail: a.f.tollefsen@sosgeo.uio.no

## Bakgrunn

Fri og åpen programvare for geografiske data (FOSS4G/FriGeoProgramvare) har lenge vært tilgjengelig som alternativer til proprietær programvare for geografiske data. Den senere tiden har mange av disse alternativene fått økt nytteverdi ved at programmene nå støtter både raster og vektor formater og har fått et rikere utvalg av analyseverktøy. Mange av prosjektene har vært under utvikling i lang tid og ofte hatt begrenset dokumentasjon med mindre man kunne lese kildekoden (Schäfer og Kranzlmüller 2007). Nye programmer kommer til som tilbyr vel så bra og bred funksjonalitet og like god dokumentasjon som kostbare GIS programvarer. I dag finnes det FriGeoProgramvare alternativer til de fleste proprietære programvarene, og utviklingsprosessen går raskt takket være store og aktive utviklersamfunn.

Denne artikkelen vil gi en kort introduksjon til hva FriGeoProgramvare er, og fokusere spesifikt på PostGIS 2.0 som er en fri og åpen utvidelsespakke til databasesystemet PostgreSQL som muliggjør støtte for geografiske data. PostGIS introduserer et bredt spekter av funksjoner for manipulering og analyse av romlige relasjoner. Jeg vil deretter trekke på erfaringer fra overgangen fra ArcGIS til PostGIS for romlig databehand-

ling ved Institutt for fredsforskning og vurdere noen av fordelene og ulempene ved PostGIS. Jeg vil til slutt gi et eksempel på hvordan ny rasterfunksjonalitet i PostGIS 2.0 kan benyttes for å kombinere både raster- og vektordata.

## Hva er FriGeoProgramvare?

FOSS4G er et akronym for Free and Open Source Software for Geospatial. I Norge benyttes gjerne begrepet FriGeoProgramvare. FriGeoProgramvare er en samlebetegnelse for alle frie og åpne programvarer med støtte for romlige data. En viktig aktør er OSGeo som er en ideell organisasjon som støtter utviklingen av FriGeoProgramvare og promoteringen av denne. De står bak den årlige internasjonale konferansen FOSS4G, som har blitt arrangert siden 2006 på ulike kontinenter. De senere årene har konferansen samlet et stort antall deltakere fra hele verden og tiltrukket seg sponsorer som eksempelvis Google, ESRI og Safe Software. Her samles utviklere, brukere og partnere til presentasjoner og workshops. I den senere tiden har det skjedd en stor utvikling innen FriGeoProgramvare og brukergruppene er ikke lenger begrenset til enkeltpersoner og akademia. Flere store selskaper har også gått

over til løsninger basert på fri og åpen lisensiering, både på grunn av forenklete muligheter for tilpasning, men også grunnet lave kostnader. Mens open-source prosjekter tidligere har virket som sorte bokser (for bru-

kere uten kodekunnskap) uten fullstendig dokumentasjon, begynner mange av disse løsningene å bli meget funksjonsrike og robuste programvarer. Noen eksempler på slik programvare er presentert i tabell 1.

Tabell 1. Eksempler på FriGeoProgramvare

Navn	Type	Utvikler	Proprietær ekvivalent
PostGIS	Romlig database (SQL)	Utviklersamfunn / Refractions Research	Oracle Spatial
QuantumGIS (QGIS)	Desktop GIS	Utviklersamfunn /OSGeo	ArcGIS
R	Romlig statistikk	Utviklersamfunn	Matlab, Stata, SAS
GRASS GIS	Desktop / scripting GIS	Utviklersamfunn	ArcGIS
Mapserver	WMS Server	Utviklersamfunn	ArcGIS Server
Geoserver	WMS Server	Utviklersamfunn	ArcGIS Server

### Hva er PostGIS?

PostGIS er en utvidelsespakke til database-systemet PostgreSQL som legger til støtte for geografiske datatyper som igjen muliggjør lokasjonsbetingede spørringer ved bruk av strukturert spørrespråk (SQL). SQL er et spesialisert programmeringsspråk for databaser og brukes for å kjøre spørringer mot databasen. I tillegg til datatyper legger PostGIS til et stort antall funksjoner samt flere indekstyper som sammen skaper et raskt, funksjonsrikt og robust databasesystem for romlige data. Geografiske datatyper som benyttes i PostGIS er *geometry* og *geography*

for vektordata og *raster* for rasterdata. Vektortypene bygger på OpenGIS Consortium (OGC) sine standarder for *simple features* (OGC 2011), som bidrar til at data er kompatible med annen programvare som støtter OGC sine standarder. Det betyr at geografiske data som er lagret i en PostGIS tabell kan leses direkte inn i desktop programvare som QuantumGIS gjennom en forbindelse til databasen. OGC sine standarder baserer seg på Well-Known Text (WKT) og Well-Known Binary (WKB) som kan benyttes for å uttrykke de romlige relasjonene som beskrevet i tabell 2.

Tabell 2. Oversikt over SQL, Well-Known Text og Well-Known Binary representasjoner i PostGIS

SQL Spørring	WKT	WKB
SELECT ST_PointFromText('POINT(1 1)');	POINT(1 1)	"0101000000000..."
SELECT ST_LineFromText('LINESTRING(0 0,1 1,1 2)')	LINESTRING(0 0,1 1,1 2)	"01020000000030..."
SELECT ST_PolygonFromText('POLYGON((0 0,4 0,4 0,4 0,0),(1 1, 2 1, 2 2, 1 2,1 1))')	POLYGON((0 0,4 0,4 0,4 0,0),(1 1, 2 1, 2 2, 1 2,1 1))	"01030000000020..."

I tabellen i databasen lagres de geografiske dataene i en kolonne av type *geometry* eller *geography*. Disse to typene kan begge benyttes for å lagre geografiske data, men skiller seg fra hverandre ved at i *geometry* er dataene lagret i et valgt kartesisk referansesystem med en valgt Spatial Reference System

Identifiser (SRID), mens i *geography* er data geografiske koordinater på en sfære og målinger vil returneres i meter. Valget mellom *geometry* eller *geography* avhenger hovedsakelig av hva slags data man benytter. Om geografiske koordinater i WGS1984 er lagret i en *geometry* kolonne vil avstandene returne-

res i grader, mens geografiske koordinater lagret i en *geography* kolonne vil returneres i meter. Avstandene mellom punktene for Los Angeles og Paris er i *geometry* 121.9 grader, mens i *geography* er avstanden 9124 km. Det eksisterer også færre funksjoner for *geography* typen enn for *geometry* typen. Det vil dermed ofte oppstå situasjoner hvor konvertering mellom de to typene må utføres om ønsket funksjon ikke er tilgjengelig. Dette kan gjøres forholdsvis enkelt ved å *CASTE* en `::geometry` til `::geography`, eller omvendt, for eksempel `'POINT(0 0)::geometry` til `geometry` eller `::geography` til `geography`. Valget mellom *geometry* eller *geography* bør gjøres basert på dataene som benyttes. Store globale datasett kan dra nytte av *geography*, da målinger gjøres i stor-sirkel avstand og returneres i meter, mens *geometry* kan benyttes når vi arbeider med data på lokalt nivå hvor lokal presisjon er viktig.

### PostGIS raster

Rasterstøtte er nå fullt implementert i PostGIS gjennom støtte for raster datatyper og import ved hjelp av GDAL biblioteket. Det vil si at PostGIS nå støtter de samme formater som GDAL støtter. GDAL er et bibliotek for oversettelse mellom ulike rasterformater som er distribuert i henhold til OSGeo sine standarder. Det vil si at omtrent alle kjente rasterformater kan importeres til PostGIS. Fra og med PostGIS 2.0 er raster en fullverdig del etter at prøveprosjektet WKT Raster ble integrert i PostGIS som standard. Det gjør at brukere kan importere rasterdata og utføre akkurat de samme operasjonene på rasterdata som de tidligere har kunnet gjøre med vektordata. Dette inkluderer eksempelvis konvertering av piksler til polygoner (`ST_PixelAsPolygons`), terrenganalyser (`ST_Aspect`, `ST_Slope`, `ST_HillShade`), returnere intersection av rastere eller av raster og vektor (`ST_Intersection`), summere raster i vektor (`ST_SummaryStats`) eller klipping av rastere (`ST_Clip`). Ved implementering av PostGIS raster har man nå støtte i funksjoner for både vektor og raster. Dette muliggjør mange avanserte operasjoner da felles geografisk lokalisering i raster og vektor kan benyttes til å kalkulere ny sta-

tistikk samt aggregere og disaggregere data uavhengig av datatype. Senere i artikkelen vil jeg komme tilbake til noen eksempler på disse mulighetene.

### Fordeler

PostGIS har forenklet mange operasjoner som tidligere kunne være kompliserte å gjøre i et desktop GIS. En klar styrke i PostGIS er kombinasjonen av spørringer som er både geografiske og ikke-geografiske samtidig. Det å finne ut i samme operasjon hvilke romlige objekter som overlapper hverandre, men som bare har en felles ID på en spesifikk variabel kan være en utfordring i desktop GIS systemer som kun kan finne den geografiske overlappen. Om vi i PostGIS ønsker å finne ut hvilke punkter som ligger innenfor et sett polygoner, men bare der hvor punktets ID er lik polygonets ID kan vi spørre:

```
«SELECT pt.id, pt.geom FROM punkt1 pt,
polygon1 poly WHERE ST_Intersects
(pt.geom, poly.geom) AND pt.id = poly.id;».
```

Denne spørringen returnerer ID og geometrien til punktene bare hvor punktene er innenfor et polygon, og bare hvis punktet har samme verdi i variabelen *id* i punktet og polygonet. Dermed filtrerer vi vekk punkter som eksempelvis kan være georeferert på feil side av en administrativ grense. Ved å skrive en kort kode i SQL kan vi altså utføre en avansert spørring som både er romlig og ikke-romlig på samme tid, samtidig som vi dokumenterer hva vi har gjort for senere bruk. Dette belyser også en klar fordel ved bruk av PostGIS og SQL til dataadministrasjon da dokumentasjon av hva som har blitt gjort forenkles betraktelig i forhold til desktop GIS løsninger som mangler kodefunksjonalitet eller benytter mer innviklede språk enn SQL. Viktigheten av å vite hva vi har gjort tidligere blir fort relevant om vi tar frem data som ikke har blitt benyttet på en stund. Samtidig stiller flere og flere akademiske tidsskrifter krav om replikasjonsfiler for å dokumentere hvordan datasett er konstruert og hvordan analyser er utført. SQL tilbyr her en enkel måte å distribuere kode på, enten det er til tidsskrifter eller samarbeidspartne-

re innen et større GIS-prosjekt. Et enkelt kodespråk er også en transparent måte å dokumentere hva vi har gjort ovenfor andre fagfeller innen disiplinen vi arbeider i.

Romlig SQL er ikke noe PostGIS har monopol på (Oracle Spatial tilbyr også dette), men PostGIS er det ledende alternativet som er utgitt under fri og åpen lisensiering. Dermed har PostGIS bidratt til økt tilgjengelighet for funksjonalitet og replikasjonsmulighet til et bredt spekter av brukere, som ellers ikke ville hatt anledning til å benytte romlige databaser på grunn av dyre lisenser.

At PostGIS er open-source gjør at kildekode er tilgjengelig for brukerne. Dette gjør at utviklingsprosessen og koden bak PostGIS er transparent, noe som gir en trygghet omkring hva funksjonene faktisk gjør. Brukermiljøet i PostGIS er stort og det er lett å få brukerstøtte fra andre brukere samt at utviklerne er tilgjengelige på epostlister. Feil rettes raskt, og rettelleskode distribueres uten at man trenger å vente på neste oppdatering av programvaren.

Tilgjengelighet er et viktig begrep i PostGIS. Fra mange andre FriGeoProgramvarer kan vi koble oss opp mot vår PostGIS database og laste ned akkurat de dataene vi ønsker. Det betyr at vi ikke trenger å lagre enkeltfiler på harddisken, men kan ha de geografiske data lagret i tabeller i en database. Om vi eksempelvis ønsker å laste data fra PostGIS inn i analyseverktøyet R, eksisterer det allerede R pakker som støtter dette (RPostgreSQL). Det samme gjelder QuantumGIS for visualisering. QuantumGIS har som standard støtte for oppkobling mot PostGIS databaser og man kan velge de tabellene man ønsker å vise i QuantumGIS. Dette fører til at mange av de eksisterende FriGeoProgramvarene snakker sammen. Vi kan eksempelvis se for oss at vi lagrer våre geografiske data i PostGIS, laster de ned til R for å utføre en analyse på dataene, hvor resultatene lagres i databasen. Deretter kan resultatet av analysen visualiseres i QuantumGIS eller gjennom en WMS tjeneste ved å lese direkte fra databasen.

At flere brukere samt flere ulike programmer kan koble seg opp mot samme database og dermed arbeide med de samme dataene er en klar fordel. Ved grundig versjonering, sik-

rer man at alle aktører i et prosjekt arbeider med de samme grunndataene og minimerer derved potensialet for inkompatibilitet.

### Ulemper

En av fordelene til PostGIS er tilgjengelighet til dataene som er lagret i databasen. Dette betinger imidlertid at man bruker programvare som har PostGIS støtte og kan lese fra databasen. Det oppstår dermed utfordringer knyttet til dataenes mobilitet om de skal distribueres til plattformer som ikke kan benytte oppkoblinger mot databasen. Mens man i programvare som ArcGIS kan jobbe direkte mot filer som kan distribueres enkeltvis, betinger tilsvarende i PostGIS at man eksporterer geografiske data til filer. En mulig løsning på dette er bruk av programmene *pgsql2shp* og *pgsql2raster* som eksporterer en tabell til henholdsvis shapefiler og rasterfiler. Dette muliggjør distribusjon av innholdet i en database selv for brukere, som er avhengige av shapefiler eller rasterfiler for videre bearbeiding. Denne utfordringen er ikke unik for PostGIS, men for databaser generelt. Å utarbeide effektive distribusjonsrutiner av geografiske data kan dermed være en utfordring.

En annen ulempe med open-source er faren for diskontinuitet i programvareutviklingen. Om en bedrift er avhengig av PostGIS for daglige gjøremål og det oppdages en feil i programmet vil det ikke bli rettet om utviklingen av PostGIS skulle avsluttes (med mindre andre tar over prosjektet). Dette er selvsagt også et problem med proprietære programvare, men mens man i proprietære selskaper er avhengig av betalingsvillighet vil man i open-source sammenheng være avhengig av kontinuitet i utviklermiljøet. PostGIS utvikles i dag av Refrations Research, som er et kommersielt selskap. Lisensen er imidlertid formulert slik at videreutvikling er mulig i tilfelle Refrations Research avslutter sitt arbeide med PostGIS.

Brukere av open-source er primært avhengig av utvikler- og brukermiljø for å få brukerstøtte. Per i dag er disse miljøene en uvurderlig ressurs og støtte for brukere, men har selvsagt ikke noe ansvar for å følge opp problemer brukerne har med den daglige bru-

ken. I proprietære programvarer har man gjerne supporttjenester som sikrer brukerne den daglige brukerstøtten. Det har imidlertid i den senere tiden dukket opp mange konsulentselskaper som tilrettelegger tjenester for kunder som benytter seg av open-source GIS slik som PostGIS. I USA er denne utviklingen på full fart fremover da mange større selskaper går fra proprietær programvare til open-source løsninger.

### Hva er nytt i PostGIS 2.0?

PostGIS 2.0 er en milepæl i utviklingen av PostGIS. Et høydepunkt er inkluderingen av støtte for raster og raster/vektor funksjonalitet samt ulike målefunksjoner for 3D objekter, topologistøtte, nye indekstyper for fler-D objekter. Rasterstøtte har lenge vært under utvikling gjennom prosjektet WKT Raster, men heter nå PostGIS raster og er en fullverdig del av PostGIS 2.0. En ny datatype *raster* er lagt til hvor en rad i tabellen representerer en brikke (tile) i rasteret. En tabell representerer et helt rasterdekke. Rasterdata kan lastes inn i databasen ved hjelp av *raster2pgsql* fra kommandolinjen, på samme måte som man har kunnet benytte *shp2pgsql* for shapefiler til databasen. Med raster i databasen kan man i versjon 2.0 utføre spørringer mellom vektor og raster, eller raster og raster. Tidligere funksjoner som *ST\_Intersects()* for å finne ut om to kartfigurer overlapper, kan nå benytte både raster og vektor som input. En annen funksjon er *ST\_PixelAsPolygons()* som returnerer hver piksel som en kartfigur med tilhørende geometri, sammen med egenskaper knyttet til denne pikselen, samt X og Y koordinatene til pikselen.

Om man arbeider med vektor og raster er det ofte nødvendig å aggregere rasterdata til vektordata. I den neste delen vil jeg ved hjelp av erfaringer belyse noen av de nye mulighetene i PostGIS 2.0, med fokus på kombinasjonen av raster og vektor for å omforme rasterdata til polygondata.

### Erfaringer

Ved Institutt for fredsforskning har vi den senere tiden i økende grad arbeidet med geo-

grafiske data relatert til konflikter, med spesiell vekt på borgerkriger. Borgerkriger betinger at informasjon er på et annet aggregeringsnivå enn for internasjonal konflikter, og lokale forhold spiller ofte en større rolle i hvor og hvordan konflikter oppstår. Subnasjonal informasjon er derfor av stor interesse for borgerkrigsforskningen. Interessen for GIS og geografiske data innenfor samfunnsvitenskapene er økende, og det utarbeides i den sammenhengen også jevnlig nye, relevante datasett. Universitetet i Uppsala (Sundberg et al., 2010) har for eksempel registrert og georeferert alle konflikthendelser i Afrika fra 1989 til i dag. Slike datasett i kombinasjon med statistikk om fattigdom og andre sosioøkonomiske faktorer, ressurser og naturgeografi, muliggjør nye typer statistiske analyser ved hjelp av både romlig analyse og romlig økonometriske metoder. Parallelt med denne utviklingen, har PRIO arbeidet med utvikling av et felles rammeverk for innhenting og kombinasjon av romlige data fra ulike kilder, for å skape en felles analyseenhet. Dette rammeverket er et vektor grid bestående av grid celler, en løsning som er i tråd med tilsvarende arbeid i SSB og andre statistikkbyråer (Strand og Bloch 2009; EFGS 2013). PRIO-GRID 1.0 (Tollefsen et al., 2012) ble lansert i 2012 og inneholder en mengde variabler fra eksisterende geografiske data generert ved bruk av GIS. Utviklingen av rammeverket startet ved hjelp av ArcGIS men behovet for enkel replikasjon ble ikke dekket av dette systemet. For bruk innen academia ville det være nødvendig å distribuere kode. Dette forutsetter programvare som ikke krever kostbare lisenser. Vi gikk derfor over til PostGIS.

Datagrunnlaget besto av et stort antall ulike kilder og involverte mange ulike formater. Eksempler er konflikthendelser registrert som enkeltpunkter; sosioøkonomiske data knyttet administrative enheter registrert som vektorpolygoner; veier registrert som linjer; arealdekke inkludert fjell og skog registrert som rasterdata; og klimadata som nedbør og temperatur levert i NetCDF format. For at vi skulle kunne importere alle disse data inn i vårt helhetlig rammeverk, ble PostGIS valgt som løsning. Dette ga videre mulighet for å importere data til statis-

tikkprogrammer som R. Siden PostGIS raster benytter seg av GDAL biblioteket er det svært få formater som ikke kan importeres til databasen. I ettertid ser vi at løsningen har fungert bra til vårt arbeid med geografisk informasjon.

### Konklusjon

Denne artikkelen har introdusert hva fri og åpen programvare for geografiske data er, samt gitt en kort introduksjon til hva PostGIS er og kan brukes til. Jeg har vist noen av funksjonene i PostGIS 2.0, med et spesifikt fokus på den nylig introduserte rasterstøtten. Mens de tidligste versjonene av PostGIS har preg av å ha begrenset funksjonalitet, tilbyr PostGIS 2.0 nå så rik funksjonalitet, at vi i vårt arbeid ikke lenger er avhengige av proprietære løsninger. Dette forenkler også distribusjon av data til brukere, samt deling av kode da det ikke betinger tilgang til kostbare lisenser. De siste årene har det skjedd en kraftig økning i antall FriGeoProgramvare, samt at kvaliteten på disse har økt betraktelig. De senere versjonene bærer preg av grundig testing og forbedret dokumentasjon, noe som gjør fri- og åpen programvare for geografiske data mer tilgjengelig, også for brukere uten kunnskap om koding og programmering.

### Referanser

- European Forum for GeoStatistics, 2013, <http://www.efgs.info>, last accessed 17.07.2013
- Open Geospatial Consortium, 2011, *OpenGIS Implementation Standard for Geographic information – Simple feature access – Part1: Common architecture*. Open Geospatial Consortium Inc, OpenGIS Project Document Number 06-103r4.
- Schäfer T. og Kranzlmüller, Patrick 2007, RTFM! Teach-Yourself Culture in Open Source Software Projects, i Hug, Theo (ed.): *Didactics of Microlearning*, Berlin, New York: Waxmann, s. 324–240
- Strand, Geir-H. og Bloch, Vilni V.H. 2009, Statistical grids for Norway. Documentation of national grids for analysis and visualisation of spatial data in Norway, Documents 2009/9, Statistisk sentralbyrå, Oslo
- Sundberg, Ralph, Lindgren, Mathilda og Ausra Padskocimaite, 2010, *UCDP GED Codebook version 1.5-2011*, Department of Peace and Conflict Research, Uppsala University.
- Tollefsen, Andreas F., Strand, Håvard og Buhaug, Halvard, 2012, *PRIO-GRID: A unified spatial data structure*. *Journal of Peace Research* 49(2), s. 363–374