

★ Hands-on Geodata!

Fra Karl Koder til Geodata-Kriger på 4 timer

Af: Pimin Konstantin Kefaloukos (kostas@diku.dk)
Geodatastyrelsen, Grontmij A/S, Københavns Universitet



Materiale til denne præsentation findes på
Github

<https://github.com/skipperkongen/geodata-journalism>

Denne præsentation ligger i roden af ovenstående git
repository

★ Program

1. Geodata formater og standarder
2. Værktøjer og workflows for data journalister
3. Spørgsmål og svar

★ Hands-on Geodata! pt. 1

Formater og standarder

★ Formater og standarder

Standarder for mange aspekter af geodata:

- Data modeller (logisk)
- Repræsentation (formater, projektioner)
- Services (read, write & analytics)
- Programmering (API'er)
- Visualisering (kartografisk udtryk)

★ Data modeller

Vector data: geografiske objekter

- {id, geometry, properties} # feature
- FeatureCollection -> Feature -> Geometry
data set objekt attribute

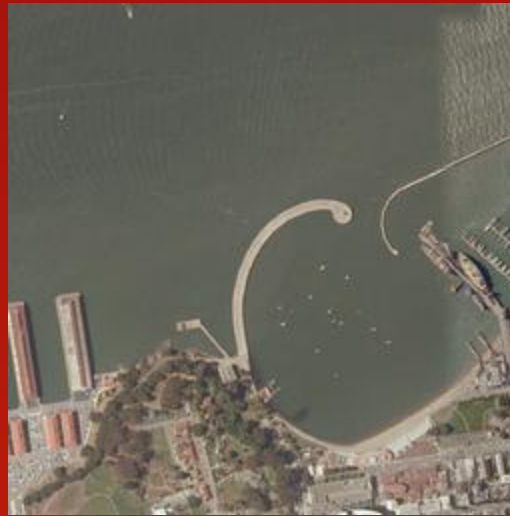
Raster data: (implicit/explicit) geokodet grid

- Data i celler

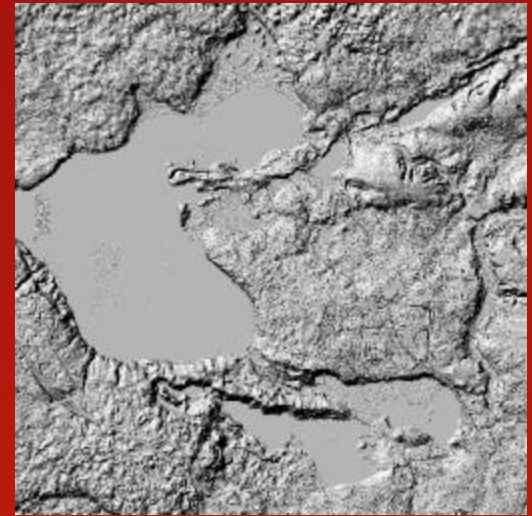
★ Eksempler: raster data



Tegnet kort



Luftfoto



Maskinlæsbart

★ Eksempel: Vector data

```
{  
  "type": "FeatureCollection",  
  "features": [  
    ...  
  ]  
}
```

```
{  
  "type": "Feature",  
  "geometry": {  
    "type": "Point",  
    "coordinates": [  
      55.676949,  
      12.56845  
    ]  
  },  
  "properties": {  
    "name": "Ekstrabladet"  
  }  
}
```


★ Repræsentation

Format:

- Vector data:
 - GeoJSON, ESRI Shape, PostgreSQL tabel
- Raster data:
 - GeoTIFF, JPEG, PNG

Sted-på-jorden:

- Flere måder at angive samme sted på:
 - Geografiske koordinater
 - projicerede koordinater

★ Sted-på-jorden

Hvordan angiver vi det her sted?



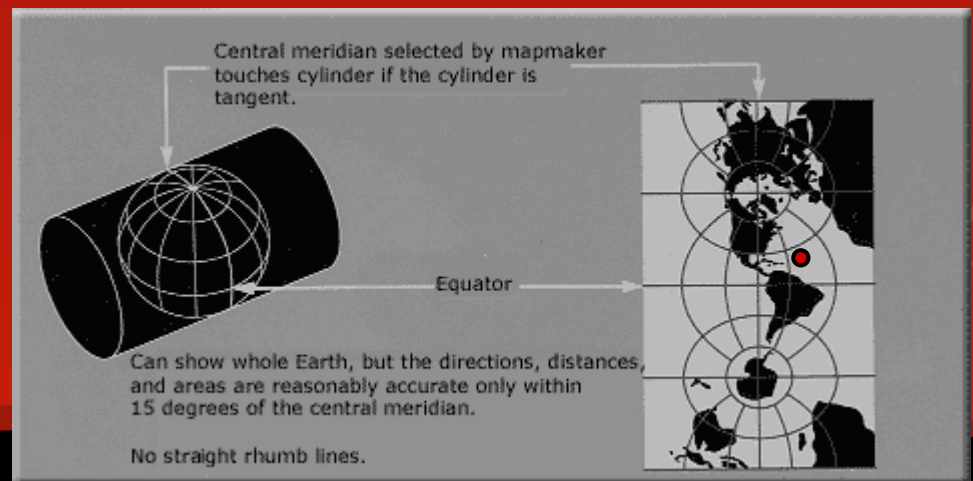
★ Sted-på-jorden

To valgmuligheder:

- Geografiske koordinater
- Projicerede koordinater

12.6° nord, 44.6° vest

5313851.73299827 -7305701.5960243



★ Sted-på-jorden

Vi kan omregne mellem koordinat-systemer:

```
SELECT ST_AsText(ST_Transform(  
    ST_GeomFromText(  
        'POINT(55.676949 12.56845)', -- EB  
        4326),  
        25832))  
:  
'POINT(6161631.70404481 1994183.27934432)'
```

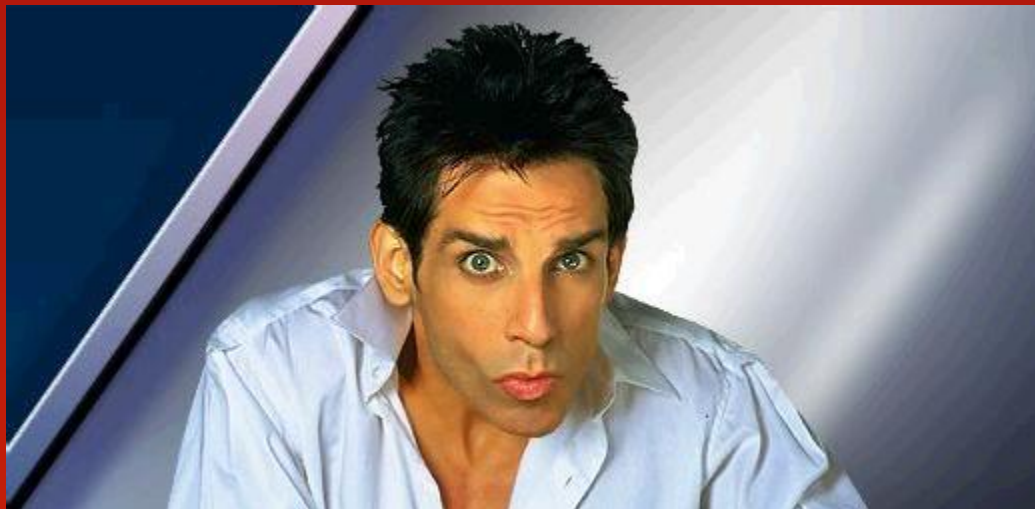
Geografiske koordinater

UTM projektion for DK

★ Sted-på-jorden

Der er to slags mennesker

- Type 1: "projektioner er vigtige"
- Type 2: "projektioner er røvkedelige"
(lad os bare bruge én til hele verden)



★ Sted-på-jorden

- Langt de fleste mennesker tilhører type 2
- Brasilien er 3 x større end Grønland...



★ Sted-på-jorden

- Den danske stat synes projektioner er vigtige...
- På download.kortforsyningen.dk kan man downloade offentlige danske data
- Data er i den såkaldte "ETRS89 / UTM Zone 32N" projektion (kode "EPSG:25832")
- Der kan være behov for at reprojicere...

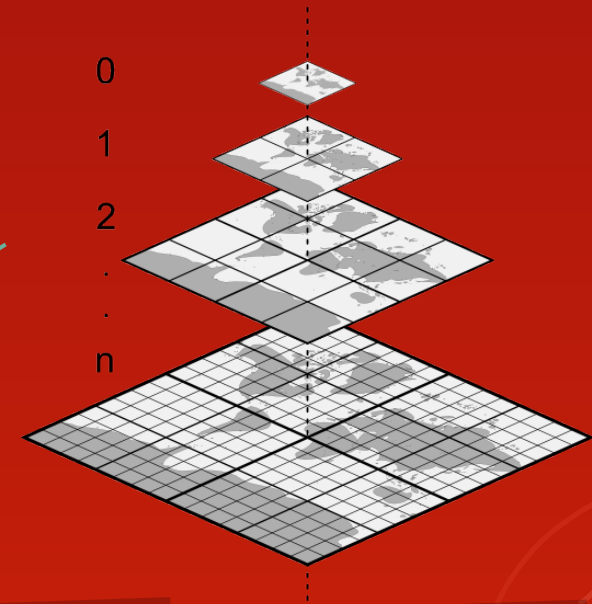
★ Services

Sofistikerede:

- Web Map Service (WMS)
- Web Coverage Service (WCS)
- Web <foo> Service

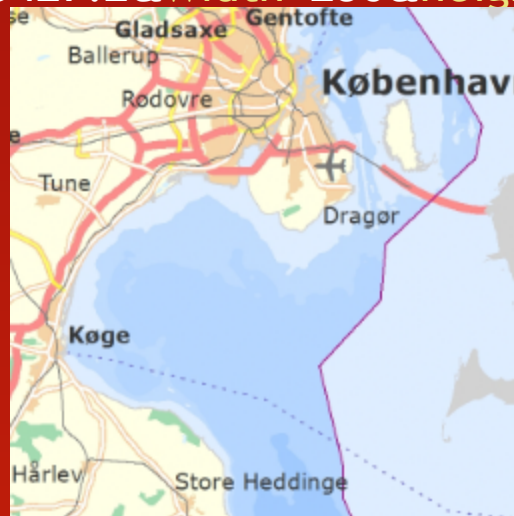
Grid-baserede:

- XYZ
- TMS
- WMTS



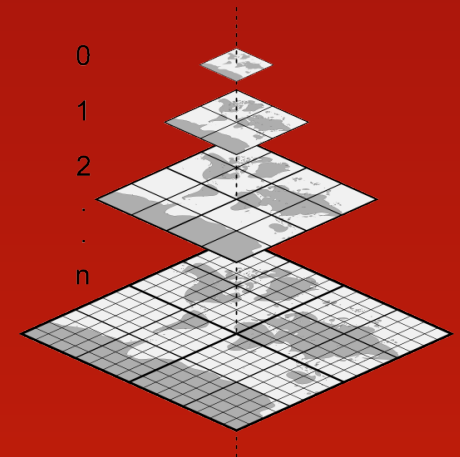
★ Services: WMS

[http://b.kortforsyningen.kms.dk/service?
servicename=topo_skaermkort&layers=dtk_skaermkort&transparent=FALSE&f
ormat=image/png&service=WMS&version=1.1.1
&request=GetMap&styles=&srs=EPSG:25832&bbox=696716.
8,6132998.4,749145.6,6185427.2&width=256&height=256](http://b.kortforsyningen.kms.dk/service?servicename=topo_skaermkort&layers=dtk_skaermkort&transparent=FALSE&format=image/png&service=WMS&version=1.1.1&request=GetMap&styles=&srs=EPSG:25832&bbox=696716.8,6132998.4,749145.6,6185427.2&width=256&height=256)



✦ Services: XYZ

<http://a.tile.stamen.com/toner/13/4382/2563.png>



✦ Eksempel: XYZ

<http://a.example.com/13/4382/2563.json>

```
{ "type": "FeatureCollection",  
  "features": [  
    { "type": "Feature",  
      "geometry": { "type": "Point", "coordinates": [102.0, 0.5] },  
      "properties": { "prop0": "value0" }  
    },  
    { "type": "Feature",  
      "geometry": {  
        "type": "LineString",  
        "coordinates": [  
          [102.0, 0.0], [103.0, 1.0], [104.0, 0.0], [105.0, 1.0]  
        ]  
      },  
      "properties": {  
        "prop0": "value0",  
        "prop1": 0.0  
      }  
    },  
    { "type": "Feature",  
      "geometry": {  
        "type": "Polygon",  
        "coordinates": [  
          [[100.0, 0.0], [101.0, 0.0], [101.0, 1.0],  
            [100.0, 1.0], [100.0, 0.0]]  
        ]  
      }  
    }  
  ]  
}
```

GeoJSON

(Polymaps kan vise GeoJSON tiles)

★ Program

1. ~~Geodata formater og standarder~~
2. Værktøjer og workflows
3. Til data journalister
4. Til eb.dk

★ Hands-on Geodata! pt. 3

Værktøjer og workflows

★ Værktøjer

Værktøjer

- I stedet for at gennemgå værktøjerne slavisk, vil vi prøve dem i nogle øvelser...
- Vi hopper derfor direkte videre til **workflowet...**

★ Workflow

Workflow for data journalisten:

1. Få fat i geodata
2. Jonglere rundt med geodata
3. Kombinere/analysere geodata
4. Publicere data-visualiseringer i artikler

Ingenting -> data -> information -> historie

★ Workflow: Få fat i geodata

To muligheder:

1. Finde data der allerede er geodata
2. Transformere alm. data til geodata

1 er let nok (givet at vi snart lærer at jonglere med geodata), så lad os prøve 2...

★ Øvelse: Skabe geodata

Øvelse: Geokodning med Python, visualisering med Javascript.

<https://github.com/skipperkongen/geodata-journalism/tree/master/exercises/geocoding>

★ Workflow

Workflow for data journalisten:

1. ~~Få fat i geodata~~
2. Jonglere rundt med geodata
3. Kombinere/analysere geodata
4. Publicere data-visualiseringer i artikler

Lad os nu prøve at jonglere lidt med nogle geodata...

★ Jonglere med geodata

I den følgende øvelse, vil vi prøve at kaste lidt rundt med data.

1. Reprojectere koordinater
2. Konvertere mellem fil-formater
3. Importere data i en cloud geodata platform

★ Øvelse: Jonglere med data

Øvelse 2: Reprojektion af koordinater

<https://github.com/skipperkongen/geodata-journalism/tree/master/exercises/reprojection>

Øvelse 2b: Konvertering mellem formater

<https://github.com/skipperkongen/geodata-journalism/tree/master/exercises/converting>

OGR2OGR cheat sheet: <https://github.com/dwtkns/gdal-cheat-sheet>

Hvis du skulle få lyst til at lære andre tricks.

★ Workflow

Workflow for data journalisten:

1. ~~Få fat i geodata~~
2. ~~Jonglere rundt med geodata~~
3. Kombinere/analysere geodata
4. Publicere data-visualiseringer i artikler

Lad os nu prøve at jonglere lidt med nogle geodata...

★ Kombinere/analysere geodata

- Du kan nu flytte rundt på geodata som det passer dig
- SQL i PostgreSQL med PostGIS extension, er en god måde at analysere geodata på
- CartoDB er en cloud database baseret på PostgreSQL + PostGIS
- Lad os prøve at analysere nogle geodata i CartoDB...

★ Øvelse: Geodata analyse

Øvelse: kombination og analyse af geodata

Vi sammenstiller lande med floder
(*forestil dig at floder = mord eller noget
andet spændende...*)

<https://github.com/skipperkongen/geodata-journalism/tree/master/exercises/analytics>

★ Workflow

Workflow for data journalisten:

1. ~~Få fat i geodata~~
2. ~~Jonglere rundt med geodata~~
3. ~~Kombinere/analysere geodata~~
4. Publicere data-visualiseringer i artikler

★ Visualisering

Med CartoDB kunne dette dårligt være nemmere. Der er en Share knap...

★ Workflow

Workflow for data journalisten:

1. ~~Få fat i geodata~~
2. ~~Jonglere rundt med geodata~~
3. ~~Kombinere/analysere geodata~~
4. ~~Publicere data-visualiseringer i artikler~~

★ Program

1. ~~Geodata formater og standarder~~
2. ~~Værktøjer og workflows~~
3. Spørgsmål og svar

★ Hands-on Geodata! pt. 4

Spørgsmål og svar

★ Stikord

Kortproduktion:

- Mapnik, SLD, TileMill

Servere:

- GeoServer, SpatialSuite, GeoWebCache, MapProxy, MapServer

Javascript klienter:

- OpenLayers, Polymaps

Feeds:

- GeoRSS

Offentlige data kilder:

- Kortforsyningen, ...

Mere cloud:

- Amazon EC2, Fusion tables

Desktop:

- Quantum GIS

★ Tak for i dag! 😊

Pimin Konstantin Kefaloukos

- kostas@diku.dk
- tlf: 26 11 60 03

Organisationer jeg repræsenterer: