

31. juli 2013

Septimas hørings svar vedrørende dokumenteterne FKG “datamodellen -  
Version 2 3 1 - Fysisk implementering.pdf” og “FKG\_2\_3\_1\_mssql.sql”

## Indholdsfortegnelse:

1	Indledning .....	2
2	Generelle bemærkninger .....	3
2.1	Forholdet til eksisterende implementeringer.....	3
2.2	Forholdet til andre databaseplatforme.....	3
2.3	Temagrupper .....	3
2.4	Sikkerhedsmodel .....	3
2.5	Versionering.....	3
2.6	Diverse .....	3
3	Specifikke bemærkninger til “datamodellen - Version 2 3 1 - Fysisk implementering.pdf” .....	4
3.1	Afsnit 1.3.....	4
3.2	Afsnit 1.4.....	4
3.3	Afsnit 2.2.....	4
3.4	Afsnit 3.2.....	4
3.5	Afsnit 4.1.....	4
3.6	Afsnit 4.2.....	4
3.7	Afsnit 4.3.....	5
3.8	Afsnit 4.5.....	5
3.9	Afsnit 4.6.....	6
3.10	Afsnit 4.7.....	6
3.11	Afsnit 4.9 - 3. afsnit .....	6
3.12	Afsnit 4.9 - 4. afsnit .....	7
3.13	Afsnit 4.9 - 5. afsnit .....	7
3.14	Appendiks A.....	7
4	Specifikke bemærkninger til “FKG_2_3_1_mssql.sql” .....	8
4.1	Postnumre .....	8
4.2	Indexer .....	8
4.3	Insert triggers.....	8
4.4	Update triggers .....	8
4.5	Funktionen FKG_geometry_is_valid .....	8

# 1 Indledning

Nærværende dokument er Septimas besvarelse af høring vedr. Fælles Kommunale Geodata FKG - offentliggjort 8/7-2013 på KL's hjemmeside.

Septimas besvarelse er udarbejdet efter en granskning af dokumentet "FKG datamodellen - Version 2 3 1 - Fysisk implementering.pdf" samt en stikprøvevis gennemgang af script til SQL-server generering: "FKG\_2\_3\_1\_mssql.sql".

Høringssvaret indeholder *ikke* en komplet gennemgang af samtlige temaer, idet den dermed forbundne arbejdsbyrde har været for omfattende.

Vores høringssvar vedrører udelukkende de IT-tekniske aspekter i materialet - de faglige kommentarer til omfang og indhold af de enkelte temaet har vi ikke baggrund for at besvare.

Er der eventuelle spørgsmål eller behov for uddybende kommentarer, kan Septima kontaktes:

- Christian Fischer – [christian@septima.dk](mailto:christian@septima.dk) – tlf: +45 9132 6942
- Gregers Petersen – [gregers@septima.dk](mailto:gregers@septima.dk) – tlf: +45 9132 6945

## 2 Generelle bemærkninger

### 2.1 Forholdet til eksisterende implementeringer

I de fysiske implementerings guidelines er der nogle forskelle i forhold til den logiske model (se særligt vores kommentarer til: Afsnit 4.2, 4.3, 4.5, 4.6, 4.7 og 4.9.

I den forbindelse, anbefaler vi FKG-gruppen at overveje de ekstra omkostninger, der vil være for kommunerne ved at ændre allerede eksisterende implementeringer, så de stemmer overens med implementeringsguidelines.

### 2.2 Forholdet til andre databaseplatforme

Afsnit 4 om implementeringsguidelines er fokuseret omkring en MS SQL-server implementering. En række af de konkret beskrevne guidelines, kan ikke følges for andre databaseplatforme. Derfor er bemærkningen i afsnit 4.1 *“Med udgangspunkt i implementeringen, som den foreligger til MS SQL Server, beskrives de forskellige guidelines, som bør følges, uanset hvilken databaseplatform, implementeringen er vendt mod.”* problematisk.

Eksempler:

- Datatyper - der er forskelle mellem MS SQL-server og andre platforme i navngivning af datatype (GUID / UUID, varchar() / character varying(), smallint / NUMBER(,) , bit / boolean) etc.
- Spatiale indexer - der findes ikke et Bounding Box begreb i Postgres

Vi anbefaler, at MS SQL-server specifikke begreber fjernes fra implementeringsguidelines - alternativt angives som eksempler og ikke som regler.

### 2.3 Temagrupper

Modellen indeholder ikke definition af de enkelte temaers tilhørsforhold til de 14 anvendte temagrupper. Temagrupperne er beskrevet i den logiske datamodel (afsnit 3.3.1).

Vi anbefaler, at tilhørsforholdet modelleres i den fysiske implementering f.eks. som en parent-child relation, hvor hvert tema tilhører een temagruppe.

### 2.4 Sikkerhedsmodel

Der findes i implementeringsbeskrivelsen ingen vejledning i hvordan en sikkerhedsmodel ovenpå implementeringen kan / bør implementeres. Vi vil anbefale, at man indføjer et kort afsnit om at der bør oprettes brugere, der kun kan skrive/læse i de valgte interfaceviews samt læse i kodelisterne.

### 2.5 Versionering

Der beskrives i implementeringen ikke noget sted, hvor implementeringsversion gemmes i databasen. Idet det må forventes, at der kommer mange implementeringsversioner (mindst én pr. version af den logiske datamodel), vil vi anbefale at disse metadata ligeledes gemmes i den implementerede database (f.eks. een version for implementeringen og een version for den tilsvarende logiske datamodel - impl\_ver='2.3.1b', fkg\_ver='2.3.1').

### 2.6 Diverse

Dokumentet indeholder en del stave- og slåfejl, som bør rettes.

### 3 Specifikke bemærkninger til “datamodellen - Version 2 3 1 - Fysisk implementering.pdf”

#### 3.1 Afsnit 1.3

Versionsnummer bør fremgå ved MS SQL server. Jf. scriptet er dette MS SQL server 2008, som bør indføres i dokumentet. Øvrige krav til databaseinstallation bør ligeledes anføres (f.eks. collation).

#### 3.2 Afsnit 1.4

Vi anbefaler, at understøttelse af QGIS tilføjes. QGIS er et udbredt OpenSource Desktop GIS produkt, som vi forventer vil være af interesse for en del kommuner.

#### 3.3 Afsnit 2.2

Teksten “oprettelse af alle kodeværdier” bør rettes til “oprettelse af alle kodeværdier, som er fælles for alle kommuner”. F.eks. er lookup tabellen d\_vejnavn ikke oprettet i scriptet (og skal heller ikke).

#### 3.4 Afsnit 3.2

Der er uoverensstemmelser mellem den logiske datamodel og den fysiske implementering:

Forskelle i udvekslingsnavn:

Logisk datamodel - udvekslingsnavn	Interfaceview - Viewnavn
t_5009_Pumpelag	t_5009_pumpelaug

Vi anbefaler at den logiske datamodel altid følges i implementeringen, og at navnet i den logiske datamodel rettes i forbindelse med en opdatering af denne.

#### 3.5 Afsnit 4.1

Teksten “... andre kommende implementeringer af FKG datamodellen ligeledes **skal følge**.” og senere “...som den foreligger til MS SQL Server, beskrives de forskellige guidelines, som **bør** følges...”

Vi anbefaler, at der i begge sætninger står **bør** med den argumentation, at implementeringsguidelines netop er FKG's anbefalinger for en implementering. Andre implementeringer kan anvendes. Efter vores vurdering er det væsentlige, at interfaceviews og kodelister standardiseres for så vidt angår navngivning, datatyper og forretningsregler.

#### 3.6 Afsnit 4.2

Angående argumentet med risiko for inkonsistens ved at opdele temaer i en specifik og en generel del (1:1 relation): vi har meget vanskeligt ved at se, hvordan dette kan ske med tilgangen til databasen gennem interfaceviews. Metoden med at anvende interfaceviews sikrer jo *netop*, at der ikke kan forekomme denne type inkonsistens (og strider i øvrigt imod argumentet for at bruge interfaceviews som det fremføres i afsnit 3.2). Modsat har en udskillelse af den generelle del i en separat tabel nogle fordele i forhold til den angivne metode:

- Forretningsregler for attributter i den generelle del, kan implementeres én gang i stedet for 82 gange.
- De 3 relationer fra den generelle del (cvr\_kode, oprindkode og statuskode) kan reduceres til 3 i stedet for 246 (3 \* 82)
- Det bliver mere klart i den fysiske model, hvordan strukturen med en generel del og en specifik del fungerer
- Der kan enkelt sikres eksplicit at det globalt unikke objekt\_id ikke kan gentages for flere objekter på tværs af temaer.

Vi vil anbefale FKG-gruppen nøje at overveje fordelene ved en udsplitning i en specifik tabel. Se i øvrigt kommentarer til afsnit 4.3 om generel sikring af unikt objekt\_id.

### **3.7 Afsnit 4.3**

Jævnfør vore erfaringer med implementering af FKG til de større GIS-platforme (QGIS, GeoMedia, MapInfo og ArcGIS gennem plugin), er der kun MapInfo, der ikke direkte understøtter brugen af GUID som primærnøgle. MapInfo kræver udover dette en række ændringer, f.eks. understøttes (status: december 2012) ikke direkte editering af (varchar-)felter større end 254 tegn. Dette udelukker editering af en lang række af modellens felter i MapInfo. Vi anbefaler derfor, at det undersøges, hvormange "...mange af dagens GIS-programmer..." der er tale om og at det desuden undersøges om ikke - bl.a. pga. problemet med editering af felter med mere end 254 tegn - der alligevel skal editeres f.eks. ved hjælp af et MBX-plugin til MapInfo.

Vi anbefaler, at der *ikke* introduceres en ny primærnøgle (ID - integer), men at version\_id som beskrevet i den logiske datamodel bibeholdes som primær nøgle.

Efter vores overbevisning mangler der - jvf. kommentarerne til afsnit 4.2 - desuden en tilsikring af at objekt\_id i sammenstilling med systid\_fra behandles som en globalt unik nøgle. Dette nævnes ikke i dette afsnit og jvf. kommentarerne i afsnittet ovenfor er der derfor med den nuværende implementering ingen garanti for at man ikke kan indsætte flere samtidige objekter med samme objekt\_id - i den nuværende beskrivelse sikres dette ikke engang i samme tabel. Dette strider mod ordlyden og meningen som beskrevet i logiske datamodel.

### **3.8 Afsnit 4.5**

Der benyttes i den givne implementering varchar(x)-felter til implementering af tekst. Brugen af varchar gør det alene muligt at anvende ASCII-tegn og det er således ikke muligt at benytte internationale tegn. Vi anbefaler, at der i stedet benyttes nvarchar (hvor der gemmes i UTF-8), således at også særlige udenlandske specialtegn kan medtages.

### 3.9 Afsnit 4.6

Vi savner en forklaring på hvorfor navngivningen af attributter i kodelistetabeller ikke følger den logiske datamodelbeskrivelse. I den implementerede model er benyttet:

- **kode** (for primærnøglen)
- **tekst** (for den tilhørende tekstuelle beskrivelse)
- **aktiv** (for markering af om kodeværdien er i brug)
- **begrebsdefinition** (for en detaljeret beskrivelse)

I den logiske datamodelbeskrivelse benyttes en anden semantik - f.eks **d\_basis\_hastighed**:

- **hastighed\_kode**
- **hastighed**
- **aktiv**
- **begrebsdefinition**

Der er fordele og ulemper ved begge metoder, men der bør være overensstemmelse mellem den logiske beskrivelse og den fysiske implementering. Den valgte semantik vil være i strid med allerede implementerede FKG datamodeller, og vil vanskeliggøre tolkningen af den bagvedliggende datamodel. Det er desuden vores erfaring, at når alle felter ikke er navngivet ens, vil muligheden for fejl ved udarbejdelse af triggers og scripts mv. øges betragteligt.

Vi anbefaler, at FKG gruppen:

- Enten ændringer retningslinjer for den fysiske implementering, så de følger den logiske datamodel *eller*
- Ændrer den logiske datamodelbeskrivelse, så den svarer til retningslinjerne for den fysiske implementering

### 3.10 Afsnit 4.7

I den logiske datamodel afsnit "2.5.3 Tilladt geometri", er angivet at *"Geometri for objekter skal opfylde "OGC Simple Feature Specification" standarderne for punkt, linje, polygon, multipunkt, multilinje og multipolygon. Standarderne skal opfyldes på databaseniveau. (Eksempelvis må en linje ikke krydse sig selv.) For hvert tema er geometritypen enten et punkt, en linje eller en polygon eller tilsvarende multi-punkt, -linje eller -polygon."*

Dette er i modstrid med implementeringsbeskrivelsen, hvor multi-versionerne ikke findes. Alle- rede eksisterende implementeringer (og data) ligger med multigeometrier - en udveksling vil derfor ikke være mulig med den beskrevne implementering.

Vi anbefaler FKG gruppen, at retningslinjer for den fysiske implementeres ændres så den svarer til beskrivelsen i den logiske datamodel.

### 3.11 Afsnit 4.9 - 3. afsnit

Det er meningsløst og bør ikke kunne lade sig gøre at ændre temakoden / temanavnet for et objekt i et tema. Dette vil betyde at man f.eks. kan have parkinventar liggende i temaet vandløb (med vandløbsattributter).

Derudover vil vi anbefale, at man ikke kan ændre en kodeværdi i temaet ved at ændre kodelisten (dette er i øvrigt heller ikke understøttet med de triggerdefinitioner, som er implementeret i *"fkg\_2\_3\_1\_mssql.sql"*). Det vil skabe uklarhed om hvad der sker hvis både kodeværdien og teksten ændres i en update og værdierne ikke stemmer overens med hvad der ligger i kodelistetabellen. I stedet bør updates af kodelisten bortkastes.

### **3.12 Afsnit 4.9 - 4. afsnit**

Afsnit 4: Hvad er årsagen til at man vil indsætte teksten "Ukendt" i bruger\_id. Det er defineret som et obligatorisk og teksten "Ukendt" kan vel ikke fortolkes som andet end at man ikke har angivet bruger\_id. Vi anbefaler, at der checkes for NOT NULL i stedet - idet erfaringen ellers viser at der vil opstå et stort antal poster med "Ukendt" som bruger.

Samtidig anbefaler vi at definitionen i den logiske datamodel suppleres med en bemærkning om at obligatorisk / frit markeringen S/(O) skal forstås som at systemgenereret *ikke* er systemgenereret af databasen, men af den applikation, som tilgår databasen (det andet kan ikke lade sig gøre). Den samme bemærkning gør sig gældende for feltet cvr\_kode.

### **3.13 Afsnit 4.9 - 5. afsnit**

Teksten:

*"Triggers håndterer versionering i en update-transaktion således*

- En update af den aktuelle version af objektet, hvor systid\_til sættes til det aktuelle tidspunkt."*

Er gentaget.

### **3.14 Appendiks A**

Vi har ikke gennemgået appendikset i detaljer - vi antager at indholdet følger definitionen i den logiske datamodel.

## 4 Specifikke bemærkninger til “FKG\_2\_3\_1\_mssql.sql”

### 4.1 Postnumre

Basistabellen d\_basis\_postnr bliver oprettet med både Grønlandske, Færøske og “pseudo” postnumre.

Vi anbefaler, at basisversionen kun indeholder danske postnumre

### 4.2 Indexer

Der defineres flere index end der er beskrevet i guideline dokumentet. F.eks. defineres der på tabellen t\_5000\_vandl\_t følgende index:

- IDX\_t\_5000\_vandl\_t\_101 ON t\_5000\_vandl\_t (versions\_id)
- IDX\_t\_5000\_vandl\_t\_102 ON t\_5000\_vandl\_t (systid\_til)
- IDX\_t\_5000\_vandl\_t\_103 ON t\_5000\_vandl\_t (systid\_fra)
- IDX\_t\_5000\_vandl\_t\_104 ON t\_5000\_vandl\_t ([systid\_fra] ASC, [systid\_til] ASC)
- INDEX SPIDX\_t\_5000\_vandl ON t\_5000\_vandl\_t (GEOMETRI)

Vi anbefaler, at der kun implementeres de index, der er beskrevet i guidelines.

### 4.3 Insert triggers

Det er ikke muligt at indsætte data hvor man kender objekt\_id. Databasen tildeler altid et nyt objekt\_id. Dette er ikke hensigtsmæssigt, idet det umuliggør f.eks. off-line oprettelse af objekter eller udveksling af objekter på tværs af platforme.

### 4.4 Update triggers

Det kan lade sig gøre at opdatere objekt\_id. Dette bør undgås, idet en opdatering af objekt\_id ødelægger den bagvedliggende historik på objektet.

### 4.5 Funktionen FKG\_geometry\_is\_valid

Der tjekkes for om punkter er punkter, om linier er linier og om polygoner er polygoner *eller* multipolygoner. Det kan undre, hvorfor der kun for polygoner tillades multigeometrier.