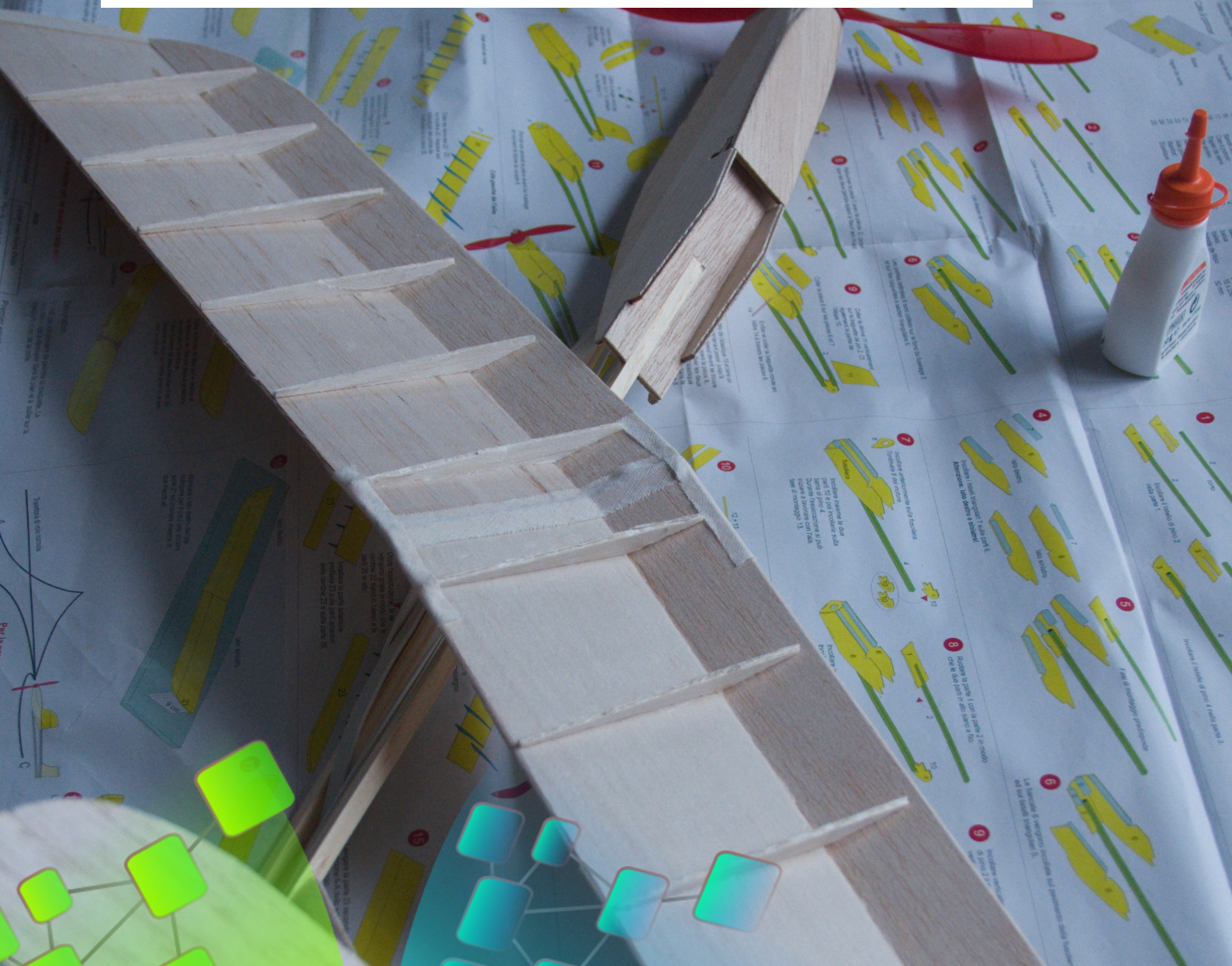




DIGITALISERINGSSTYRELSEN

Modelregler for grunddata

31 august 2016



Modelregler for grunddata

Modelreglerne er godkendt af Grunddatabestyrelsen 3. februar 2014

Version 1.1 af visse regler er godkendt af Grunddata Programkoordinationen 20. januar 2015

Reglerne udstilles og vedligeholdes på <http://arkitekturguiden.digitaliser.dk/grunddata-modelregler>

Indledning

Modelreglerne er udarbejdet som led i etableringen af grunddatamodellen: en fælles datamodel for alle grunddata.

Etablering af grunddatamodellen omfatter en række andre produkter og leverancer, som modelreglerne skal ses i sammenhæng med:

- **Modelleringsværktøj**

Værktøj som understøtter etablering og vedligehold af grunddatamodellen i overensstemmelse med modelreglerne.

- **Ibrugtagningsplan**

Plan for ibrugtagning af modelreglerne i hele grunddataprogrammet. Udgivet sammen med modelregler version 1.0.0.

- **Udstillingsplatform**

Platform for udstilling af grunddatamodellen for databrugere.

- **Styringsrammer for grunddatamodellen**

Aftale om hvordan modelreglerne og grunddatamodellen vedligeholdes, herunder definition af organisation og ansvarsfordeling, regler for versionering samt fastlæggelse af beslutningsprosesser ved ændringer. Udgivet sammen med modelregler version 1.0.0.

Grunddatamodellen skal indgå i den samlede dokumentation af grunddata, som også vil omfatte dokumentation fra Datafordeleren.

Indhold

Kapitel 1 - Introduktion	5
Formål	5
Hvad betyder en samlet og sammenhængende grunddatamodel?	6
Målsætninger	7
Fordele	7
Målgruppe	7
Læsevejledning	8
Indhold	8
Definitioner	9
Forkortelser	13
Kapitel 2 - Modelreglernes fokus og afgrænsning	14
Grunddatamodellen består af domænemodeller	14
Model for udstilling af grunddata	15
Informationsmodel	15
Objektniveau	17
Afgrænsning	17
Kapitel 3 - Arkitekturmæssige forudsætninger	18
Datafordeleren	18
Den Fælleskommunale Rammearkitektur	19
Organisationskomponent	19
Klassifikationskomponent	20
Beskedfordeler/Beskeddrevet arkitektur	20
Kapitel 4 - Anvendelse af modelregler	22
Reglerne er enten krav eller anbefalinger	22
Reglerne kan udbygges inden for forretningsdomænerne	22
Mønster for regler	23
Kapitel 5 - Generelle modelregler	24
Modeller skal udarbejdes som UML-klassediagrammer	24
Referencer til klassifikationer, forretningsmodeller og organisationsmodeller bør anvendes	25
UML-modellen skal organiseres i pakker	26
Modelentiteter skal genbruges	26
Attributter og relationer skal modelleres fyldestgørende	27
Standardiserede datatyper skal genbruges	28
UML-stereotyper skal anvendes	29
Navngivningsregler skal følges	30
Sprogregler skal anvendes	31
Datamodellen skal dokumenteres	31
Kapitel 6 - Regler om generelle egenskaber	33
Alle modelentiteter skal modelleres med persistent, unik identifikation	33
Alle modelentiteter skal modelleres med status	35
Alle modelentiteter skal understøtte dobbelthistorik og angivelse af aktører	36
Alle modelentiteter bør understøtte beskedfordeling	39

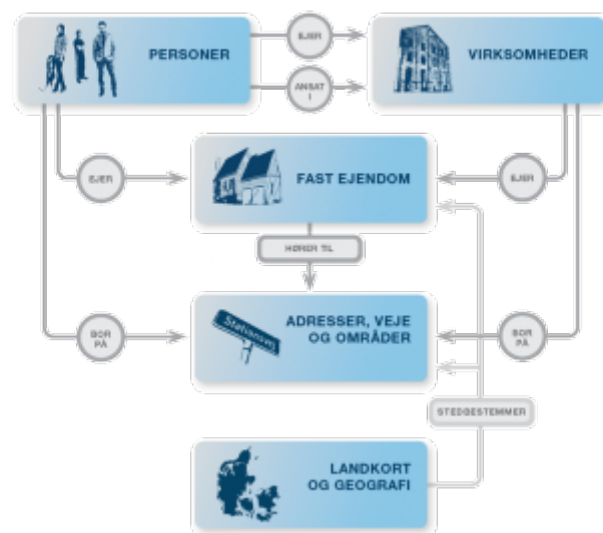
Kapitel 1 - Introduktion

1.1 Formål

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Grunddataprogrammet er igangsat med visionen om, at offentlige grunddata om personer, virksomheder, ejendomme, adresser og geografiske forhold opdateres ét sted og anvendes af alle. En mere detaljeret baggrund for grunddataprogrammet kan findes her <http://grunddata.dk>.

DEN OFFENTLIGE SEKTORS GRUNDDATA ER INFORMATIONER OM:



Figur 1 - Konceptuelt overblik over grunddataprogrammet

De offentlige grunddata bliver vedligeholdt og anvendt af flere forskellige myndigheder, og der er derfor behov for at tænke data sammen i en model, så man kan sikre et samlet overblik og på den måde undgå redundant vedligehold af data.

Grunddataprogrammet indeholder forskellige forretningsdomæner, der er relateret til hinanden og på visse områder overlappende. For at kunne skabe en datamodel for grunddata, der fremstår som sammenhængende for interessenterne er det vigtigt at sikre, at man har en fælles tilgang til modelleringsopgaven. Modelreglerne sikrer, at modelleringen af dataobjekter sker ud fra et fælles sæt retningslinjer, og at hele modellen bygger på fælles grundegenskaber.

Formålet med modelreglerne er derfor at sikre en samlet og sammenhængende grunddatamodel i et distribueret forvaltningsmiljø.



Figur 2 – Overblik over forretningsdomæner i grunddata baseret på Konceptuel datamodel version 0.8 (<http://www.digst.dk/Loesninger-og-infrastruktur/Grunddata/Delprogrammerne/Faelles-datamodel/Konceptuel-datamodel>) fra 2011. Et forretningsdomæne er fx "Virksomhed". Den korrekte afgrænsning af forretningsdomæner defineres af grunddataforvalterne.

1.1.1 Hvad betyder en samlet og sammenhængende grunddatamodel?

[Link til reglen on-line - klik for at kommentere.](#)

Vi ønsker at give slutbrugerne (myndigheder, virksomheder og privatpersoner) en samlet og sammenhængende datamodel. Det betyder, at man som bruger oplever sammenhæng på tværs af forretningsdomænerne, og at man oplever ensartet begrebsanvendelse samt ensartede modellering og generelle egenskaber for modelentiteterne i grunddatamodellen. Denne oplevelse opretholdes og vedligeholdes på trods af, at data i modellen vedligeholdes af forskellige myndigheder.

1.1.2 Målsætninger

[Link til reglen on-line - klik for at kommentere.](#)

Det primære mål med modelreglerne er at skabe den fælles tilgang til at modellere grunddata, som er nødvendig for at kunne skabe en samlet og sammenhængende datamodel.

Konkret skal modelreglerne opfylde følgende målsætninger:

- Modelreglerne skal danne grundlag for ensartet modellering af grunddata.
- Modelreglerne skal sikre det nødvendige abstraktionsniveau til at imødekomme alle interessenters behov.
- Modelreglerne skal sikre genbrug af allerede eksisterende standarder, hvor det er muligt.
- Modelreglerne skal gøre det nemt for databrugere at bygge applikationer, der bruger grunddata, og at stille ensartede forespørgsler på tværs af grunddata.

1.1.3 Fordele

[Link til reglen on-line - klik for at kommentere.](#)

Ved at anvende de fælles modelregler opnår man som grunddatamyndighed en række fordele:

- Det er nemmere at sikre fælles retningslinjer for datamodellering internt i organisationen.
- Datamodellen går på tværs af alle forretningsdomæner og giver dermed mulighed for genbrug af data.
- Det bliver nemmere at udveksle dataobjekter.
- Det bliver nemmere at sikre en høj datakvalitet.
- Der vil være mindre begrebsforvirring.
- Der vil i mindre grad være redundante data på tværs af forretningsdomænerne.

1.2 Målgruppe

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line - klik for at kommentere.](#)

Modelreglerne har fire primære interessenter:

Databrugere

Databrugere er de slutbrugere, der gennem grunddataprogrammets anvendelse af modelreglerne vil opleve, at de får en samlet, sammenhængende og effektiv måde at tilgå og anvende grunddata på.

Dataejere

Dataejerne sidder i de enkelte registermyndigheder, der opbevarer, vedligeholder og udstiller grunddata. Dataejerne har stor interesse i en samlet, sammenhængende datamodel med mulighed for genbrug og effektiv governance. Modelreglerne er en vigtig del af grundlaget for at realisere denne gevinst.

Udviklere

Udviklere skal her ses som de ledere og projektledere, forretningseksperter og systemleverandører hos såvel dataejere som databrugere, der skal levere løsninger i grunddataprogrammet til såvel udstilling som brug af data. De har en stærk interesse i modelregler, der sikrer en samlet, sammenhængende datamodel. Denne gruppe vil have brug for, at den samlede datamodel kan præsenteres på flere forskellige abstraktionsniveauer: konceptuelt, logisk og fysisk.

Datamodellører

De datamodellører, der skal udforme grunddatamodellen gennem deres arbejde med modellering af forretningsdomænerne, er afhængige af, at modelreglerne er entydige, klare og meningsgivende.

1.3 Læsevejledning

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line - klik for at kommentere.](#)

1.3.1 Indhold

[Link til reglen on-line - klik for at kommentere.](#)

Dokumentet har følgende indhold:

- **Kapitel 2 - Modelreglernes fokus og afgrænsning**

Her beskrives fokus og afgrænsning for grunddatamodellen og modelreglerne.

- **Kapitel 3 - Arkitekturmæssige forudsætninger**

Her beskrives de arkitektur- og infrastrukturmæssige forhold, som har indflydelse på udformningen af modelreglerne.

- **Kapitel 4 - Anvendelse af modelregler**

Her forklares, hvordan modelreglerne er bygget op, og hvordan de skal efterkommes. Selve reglerne følger i kapitel 5 og 6.

- **Kapitel 5 - Generelle modelregler**

I dette kapitel opstilles generelle modelregler, som har fokus på datamodellens udformning og vedrører diagrammering. Her opstilles regler for fx modelleringssprog, navngivning af elementer, sprog og dokumentation mv.

- **Kapitel 6 - Regler for generelle egenskaber**

I dette kapitel opstilles regler, som har fokus på indhold i datamodellen, og som sætter rammer for dataindhold i forvaltningsobjekterne. Her opstilles regler med betydning for fx forvaltningsobjekters identifikation og historik. Reglerne udmøntes i specificering af generelle egenskaber for alle modelentiteter.

1.3.2 Definitioner

[Link til reglen on-line - klik for at kommentere.](#)

Definitioner af begreber brugt i modelreglerne - ord med fed henviser til andre definitioner. Så vidt muligt er definitioner, som ikke bygger på andre definitioner i listen hentet i eksterne definitoriske værker.

Begreb	Definition
Aktør	Et objekt, der deltager i en aktivitet. Der er ikke alle objekter, der kan være en aktør, fordi ikke alle objekter kan siges at deltage. Nogle objekter vil fx i stedet indgå som redskaber i en aktivitet. Den Fællesoffentlige Topontologi 3 Aktør
Arkitekturbyggekloks	En afgrænset del af it-arkitekturen, som specificerer et sæt af forretningsevner. Byggekloksen kan betragtes som en genbrugelig og udskiftelig del af arkitekturen, og kan være beskrevet mere eller mindre detaljeret. TOGAF 3.21 Building Block http://pubs.opengroup.org/architecture/togaf9-doc/arch/chap03.html#tag_0...
Begivenhed	En aktivitet, der er en enkeltstående helhed. En begivenhed foregår på et bestemt tidspunkt og på et bestemt sted. Den Fællesoffentlige Topontologi 1.2.1 Begivenhed
Beskedfordeler	Et computersystem, som skal udveksle hændelsesbeskeder mellem computersystemer. Beskedfordeleren realiserer en arkitekturbyggekloks med tilsvarende forretningsevner. Som led i den fælleskommunale rammearkitektur etableres en Beskedfordeler.
Computersystem	En eller flere computere, som i sammenhæng udfører databehandling. ISO/IEC 2382-1:1993, 01.01.20
Data	Information lagret med henblik på (gen)anvendelse ISO/IEC 11179-4:2004(en), 3.3 data
Datafordeleren	Et computersystem, som effektivt og stabilt distribuerer data fra grunddataregistrene. Datafordeleren realiserer en arkitekturbyggekloks med tilsvarende forretningsevner. Den fællesoffentlige Datafordeler etableres som led i Grunddataprogrammet .
Datahændelse	En ændring i data - modsat den begivenhed i "virkeligheden", som gav anledning til ændringen i data.
Datamodel	En grafisk (og/eller tekstuel) repræsentation (model) af data, med specifikation af deres egenskaber, struktur og interne relationer. ISO/IEC 11179-1:2004, 3.2.7 datamodel
Dataobjekt	En konkret datainstans af en modelentitet. Et eksempel er et personobjekt: Person("Jens Hansen", "010101-8881", "01-01-2001"). Se også figur 3 herunder.
Datasnitflader	De specifikationer (fx XML Schema, JSON Schema, WSDL) som sætter rammer for dataformatet i en service.
Domænemodel	Datamodel for et forretningsdomæne (fx "Adresse", "Stednavn", "Virksomhed"). Alle domænemodellerne tilsammen udgør grunddatamodellen.
Forretningshændelse	Begivenhed i "virkeligheden", som har udløst en ændring i data.
Forvaltning	Den sammenhængende, databaserede udøvelse af offentlig myndighed i Danmark.

Begreb	Definition
Forvaltningsobjekt	Forvaltningens repræsentation af det konkret - fysisk eller konceptuelt - eksisterende objekt (adresse, vandløb, virksomhed, udskrivningsgrundlag), som der udøves myndighed på og som der derfor opsamles data om. Forvaltningsobjektet er en selvstændig helhed, der kan beskrives enkelt og har tilknyttet oplysninger. F.eks. kan forvaltningsobjektet "Person" have tilknyttet følgende oplysninger: "Navn", "CPR-nummer" og "Fødselsdato". Se Arkitekturguide forretningsobjekt og figur 3 herunder.
Grunddata	De data, som opbevares og forvaltes af grunddataregistre.
Grunddatamodellen	Den samlede og sammenhængende datamodel for grunddata. Grunddatamodellen er sammensat af domænemodellerne.
Grunddataregister	En datasamling, der har til formål, at indsamle og videreformidle data om forvaltningsobjekter, og som deltager i Grunddataprogrammet .
Hændelsesbesked	Et dokument, som udfærdiges af et computersystem i forbindelse med en datahændelse, med det formål at kunne videredistribueres til andre computersystemer, for at advisere disse om datahændelsen, som de så eventuelt kan reagere på.
Information	Tegn, der giver mening. Den Fællesoffentlige Topontologi 1.6.1 Information
Informationsmodel	En datamodel hvor fokus er på beskrivelsen af modelentiteternes specifikke navne, egenskaber og relationer samt disses multiplicitet og kardinalitet. IETF RFC 3198 datamodel Svarer til Logisk Datamodel se Arkitekturguide informationsmodel
Klassifikations-komponent	Et computersystem som realiserer en arkitekturbyggekreds som har til formål at opbevare, synkronisere og distribuere klassifikationssystemer.
Konceptuel datamodel	En datamodel, hvor fokus er på beskrivelsen af forvaltningsobjekternes overordnede relationer.
Model	Et objekt, der repræsenterer en entitet ved at besidde en ægte delmængde af dennes egenskaber En model kan ligne originalen til forveksling, men den er ikke originalen. En model af Vor Frue Kirke kan have samme form som originalen, men afviger i fx materiale og størrelse. Til forskel har en kopi samme egenskaber som originalen. Den Fællesoffentlige Topontologi 1.5.5 Model
Modelansvarlig	Den myndighed der til enhver tid har ejerskab og ansvar for en domænemodel.
Modelentitet	Modelleringen af et forvaltningsobjekt, hvor dets egenskaber udtrykkes som klasser og attributter. Se også figur 3 herunder.
Modellering	Dét, at lave en model af noget.
Organisations-komponent	Et computersystem som realiserer en arkitekturbyggekreds som har til formål at opbevare, synkronisere og distribuere information om organisationer - deres kontaktoplysninger, personer og interne organisering.
Relation	En entitet, der forbinder entiteter. En relation er forholdet mellem to eller flere entiteter. Den Fællesoffentlige Topontologi 1.3 Relation

Begreb	Definition
Service	En forretnings, en arkitekturbyggekreds' eller et computersystems evner til at levere ydelser til interne eller eksterne aftagere.



Figur 3 - Illustration af forholdet mellem begreberne 'Forvaltningsobjekt', 'Modelentitet' og 'Dataobjekt'.

1.3.3 Forkortelser

[Link til reglen on-line - klik for at kommentere.](#)

Forkortelse	Beskrivelse
UML	Unified Modeling Language, www.uml.org
XML	Extensible Markup Language, www.w3.org/xml
RDF	Resource Description Framework, www.w3.org/RDF/

Kapitel 2 - Modelreglernes fokus og afgrænsning

Her beskrives modelreglernes fokus og afgrænsning.

2.1 Grunddatamodellen består af domænemodeller

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line - klik for at kommentere.](#)

Grunddatamodellen er sammensat af domænemodeller - dvs. datamodeller for alle forretningsdomænerne i grunddataprogrammet (fx "Person", "Adresse", "Virksomhed"). Se også figur 2.

Hver domænemodel har en modelansvarlig - dvs. den grunddatamyndighed, som til enhver tid har ejerskab og ansvar for domænemodellen. Det er op til den modelansvarlige at fastlægge domænets omfang og afgrænsning.

Den modelansvarlige har ansvaret for, at domænemodellen afspejler domænets data og er modelleret i overensstemmelse med forretningsbehovet samt i overensstemmelse med modelreglerne.

Den samlede grunddatamodel opbevares og udstilles centralt:

NOTE:

Dette uddybes i dokumentet "Udstillingsplatform": Domænemodeller indleveres af de modelansvarlige i XMI format. XMI er en standardiseret XML-baseret udvekslingsformat for UML-modeller. UML-modellen er organiseret i pakker, hvor pakkenavnet (se [afsnit 5.2](#)) modsvarer datadomænet, for eksempel: Person, Bygning, Bestemt Fast Ejendom. Disse modeller placeres af grunddatasekretariatet på en versionsstyrende udstillingsplatform - formodentlig en Subversion-server (<http://subversion.apache.org/>) på Digitaliser.dk <https://svn.softwareborsen.dk/>, hvorfra alle kan genbruge dem. De fleste modelleringsværktøjer kan importere XMI fra en subversion-server således at modellører kan genbruge elementerne i egen modellering. Grunddatasekretariatet sørger yderligere for, at modelleringen udstilles i diagramform på et centralt sted i Grunddata-regi, således at modellører og andre kan orientere sig i modellen, uden at skulle importere den i eget værktøj.

2.2 Model for udstilling af grunddata

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Grunddatamodellen skal være modellen for udstilling af grunddata fra Datafordeleren.

Modelreglernes fokus er derfor på udstilling og kommunikation af grunddata over for databrugere, som skal hente data via Datafordeleren.

Databrugere er både eksterne databrugere og de myndigheder, som deltager i grunddataprogrammet. Grunddatamyndighederne har forpligtet sig på at bruge hinandens data på tværs og vil dermed også trække data fra Datafordeleren. Grunddatamodellen vil dermed også udgøre det samlede programs overblik over og forståelse af grunddata.

Modelreglerne omhandler ikke datamodeller for lagring eller ajourføring af grunddata internt i grunddataregistrene. Modelreglerne omhandler heller ikke datamodeller for lagring internt i Datafordeleren eller for dataflowet mellem grunddataregistre og Datafordeler.

Bemærk dog, at modelreglerne i kapitel 5 og 6 stiller krav til, at visse informationer er indeholdt i grunddata.

2.3 Informationsmodel

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Modelreglerne har fokus på logisk informationsmodellering af de data, der udstilles som grunddata for databrugere. Se Arkitekturguiden for yderligere definition af logisk informationsmodellering: [Arkitekturguide informationsmodel](#).

Informationsmodellen skal beskrive al den information, der udstilles som grunddata.

Et af perspektiverne med grunddatamodellen er en modeldrevet arkitektur, der samler vedligeholdelse af datamodeller og dokumentation et sted. Det betyder i denne sammenhæng, at informationsmodellen er den centrale datamodel, som vedligeholdes af den modelansvarlige. Ud fra informationsmodellen kan der udledes datamodeller på andre abstraktionsniveauer: *konceptuel datamodel og datasnitflader*.

Den konceptuelle datamodel skal give et helt overordnet overblik over grunddata til brug for beslutningstagere samt til kommunikation, se fx [Konceptuel datamodel version 0.8](#). Den konceptuelle datamodel skal vedligeholdes sammen med informationsmodellen.

Datasnitflader (forstået som fx fysiske skemaer i form af XML Schema) har til formål at gøre datamodellen operationel for systemudviklere. Et af perspektiverne i modelreglerne er at muliggøre automatisk generering af datasnitflader ud fra informationsmodellen.

NOTE:

Hensigten med flere af dette dokumentets regler er, at informationsmodellen automatisk skal kunne oversættes til datasnitflader, formuleret for eksempel i XML Schema eller JSON Schema. Yderligere vil det skulle undersøges, om modellen kan oversættes til andre modelleringsmetoder, som for eksempel RDF/OWL, for at kunne danne basis for Linked Data/Semantic Web-relaterede dataudvekslingsmetoder.

Metodikkerne for, hvordan disse oversættelser skal foregå samt de deraf afledte forudsætninger i domænemodellerne, er pt ikke fuldt uddybede. Dertil kommer, at det ikke er afklaret, hvordan model og snitflader bringes i spil på Datafordeleren (se [afsnit 3.2](#)). Teknologiske pilotundersøgelser og forretningsrettede undersøgelser skal afklare disse forhold, opsætte processer for transformationer samt give indspil til hvad dette kommer til at kræve af domænemodellerne.

Dokumenterne “Modelleringsværktøj” og “Udstillingsplatform” som Grunddatasekretariatet forventes at publicere, vil indeholde detaljer om metodikker og processer. Kommende versioner af Modelreglerne vil være påvirket af processernes fordringer til modellen.



Figur 4 - Modelreglernes fokus illustreret i forhold til OIO EA-reolen: Hovedfokus er på informationsmodellering på logisk niveau. Herudfra kan udledes henholdsvis konceptuel datamodel og datasnitflader på fysisk niveau.

2.4 Objektniveau

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Modelreglerne fokuserer på modellering af forvaltningsobjekter, dvs. fx "Person" eller "Adresse". Det betyder, at modelregler, dokumentationskrav og generelle egenskaber gælder for grunddatamodellens modelentiteter. Der opstilles således ikke regler på datasætniveau, ligesom metadata for datasæt ikke omhandles.

2.5 Afgrænsning

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Modelreglerne indeholder ikke krav til en bestemt fremgangsmåde for udarbejdelse af datamodeller. Informationsmodeller, som indleveres som del af grunddatamodellen, skal overholde modelreglerne, men der er metodefrihed med hensyn til domænernes modelleringsproces.

En række tekniske specifikationer ligger uden for dette dokumentets område - primært fordi de vedrører udviklingsprojekter inden for grunddataprogrammet, som endnu ikke er fuldt afklarede. Således dækker modelreglerne ikke Datafordelerens servicespecifikation, herunder dataformater og protokoller for adgang til grunddata.

NOTE:

Følgende anbefalinger, baseret på publikationer fra DANTERMcentret (www.danterm.dk/) og [OIO-arbejdsmodellen](#), kan gives for processen bag domænemodelleringen:

Informationsmodeller kan fx udvikles "top-down" ud fra en højniveau-model som fx en

begrebsmodel, eller "bottom-up" fra en fysisk datamodel eller ved systematisk gennemgang af indholdet af det dataregister, der skal udstille data.

Top-down-måden indeholder følgende trin:

1. Terminologisk begrebsmodellering
2. Udarbejdelse af en terminologisk ontologi der indeholder oplysninger om begreber i form af karakteristiske træk og begrebsrelationer.
3. Konceptuel datamodelisering
4. Udarbejdelse af en datamodel der afspejler typer af entiteter og deres indbyrdes relationer, og som udgør en abstrakt repræsentation af data.
5. Logisk datamodelisering
6. Udarbejdelse af en datamodel der specificerer organiseringen af data på en måde, som afspejler den logiske struktur i et it-system.
7. Fysisk datamodelisering
8. Udarbejdelse af en datamodel der afspejler den fysiske struktur i et it-system.

Kapitel 3 - Arkitekturmæssige forudsætninger

Grunddatamodellen har snitflader til andre projekter og planlagte komponenter i den fællesoffentlige it-arkitektur. Disse snitflader er med til at opsætte rammer og formål for, hvordan modellen bedst udformes. Den fælles it-arkitektur er ikke nødvendigvis fastlagt eller stabil, og på samme måde må reglerne for modellering af grunddatamodellen have et vist mål af dynamik for at kunne understøtte integration med den omkringliggende arkitektur.

I det følgende gennemgås en række af de elementer i den fællesoffentlige it-arkitektur, som har en direkte indvirkning på modelreglerne - primært for at danne referenceramme ved diskussion af formålet for de enkelte regler. For flere af elementerne gælder, at de ikke er færdigudviklede og i drift og dermed ikke udgør konkrete arkitekturkomponenter, som kan sætte præcise mål for grunddatamodelleringen. Derfor indeholder de følgende afsnit en række antagelser, som er blevet lagt til grund for udformningen af modelreglerne. Antagelserne er baseret på dialog med de organisationer, som er ansvarlige for implementeringen af komponenterne, og er tilstræbt så tro mod intentionerne, som det har været muligt. Skulle antagelserne vise sig ikke at afspejle det fremtidige landskab, må det bevirke justering af modelreglerne.

3.2 Datafordeleren

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Der etableres i regi af grunddataprogrammet en datafordeler, som effektivt og sikkert skal distribuere data fra grunddataregistrene, læs mere her [Datafordeler]. Data, som tilgås via Datafordeleren er også de data, som modelleres i grunddatamodellen. Da Datafordeleren endnu ikke er etableret, betyder det for modelleringsarbejdet, at der ikke kan opsættes regler for modelleringen, som direkte sigter til at understøtte en specifik servicesnitflade på Datafordeleren. Derimod må dette dokument basere sig på antagelser om, hvordan grunddatamodellen bedst understøtter dataadgang på Datafordeleren.

Antagelser:

- Det antages, at Datafordeleren på et tidspunkt vil kunne indgå i en modeldrevet arkitektur, hvor specifikationen af data håndteres i regi af modellen af data frem for i dokumentation, som er adskilt fra data.
 - Det antages, at Datafordeleren vil kunne understøtte distribution af hændelsesbeskeder - at den vil udsende beskeder om datahændelser til en fremtidig beskeddrevet arkitektur [EDA](#).
-

Derfor indeholder dokumentet regler, som skal gøre det nemmere at konvertere de udviklede UML-modeller til andre typer datamodellering. I første runde sigtes der til, at gøre det muligt, at autogenerere XML Schema på baggrund af UML-modellen. På den måde kan det på sigt gøres muligt, at udvikle grænseflader, hvor databrugere selv sammensætter udtræk og datasnitflader på baggrund af modellen.

Autokonvertering af UML-modeller til XML Schema foregår for eksempel i øjeblikket i regi af INSPIRE.

Samtidig vil det være muligt, at omtolke UML-modellen til [RDF](#). RDF-modellen kan danne udgangspunkt for en række forskellig repræsentationer af data, ligesom den kan bruges ved udstilling af grunddata i en Linked Data/Semantic Web sammenhæng.

Yderligere indeholder dokumentet regler, som skal sikre, at data, der er nødvendige for at danne hændelsesbeskeder, er tilstede i grunddata.

3.3 Den Fælleskommunale Rammearkitektur

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line - klik for at kommentere.](#)

I regi af KL og KOMBIT arbejdes der med, at implementere infrastrukturkomponenter som realisering af rammearkitekturs arkitekturbyggeklodser som skal varetage generelle funktioner i forbindelse med opbevaring og udstilling af organisations-strukturer og andre typer af referencedata, se [Den fælleskommunale rammearkitektur](#). Sammen med en beskedfordeler, er disse komponenter i udbud. Modellering og anvendelse af grunddata kan effektiviseres betydeligt ved, at anvende funktionalitet fra denne arkitektur, hvorfor modelreglerne er udfærdiget med hensyntagen til nedenstående antagelser om byggeklodserne og komponenterne.

3.3.1 Organisationskomponent

[Link til reglen on-line - klik for at kommentere.](#)

Standarderne for sag- og dokumentområdet beskriver en logisk struktur for, hvordan en given organisations organisering kan udstilles i et generelt format, som tillader eksterne systemer at referere til dele eller helheder i organisationen se [Organisation](#). De refererbare aktører i organisationen kan være både specifikke (person, it-system, funktion) eller overordnede (afsnit, kontor, organisation). Enkelte kommuner har allerede implementeret systemunderstøttelse for organisationskomponenten, og KOMBIT forbereder et udbud af en generelt anvendelig komponent - se [Den fælleskommunale rammearkitektur - støttesystemer](#).

Antagelser:

- Det antages, at organisationskomponenten bliver en permanent og entydig del af den offentlige it-arkitektur.
- Det antages, at staten vil anvende en tilsvarende komponent til udstilling af organisation.
- Det antages, at data om aktører i organisationskomponenten har identitet og bitemporale egenskaber, hvorved de kan anvendes distribueret.

Af disse grunde anbefaler reglerne i det følgende fx, at information om den aktør, som er ansvarlig for dataobjektets registrering i databasen og for dets virkning, lagres og udstilles som en reference til en organisation, udstillet i en organisationskomponent.

3.3.2 Klassifikationskomponent

[Link til reglen on-line - klik for at kommentere.](#)

På samme måde planlægger KOMBIT (se [Den fælleskommunale rammearkitektur - støttesystemer](#)) at den fælleskommunale rammearkitektur skal indeholde en klassifikationskomponent - også beskrevet i Sag & Dokument-standarderne [Klassifikation]. Klassifikationskomponenten skal opbevare og udstille strukturerede data - taksonomier og grafer - som afspejler forretningsviden, der har en intern struktur og kan bruges til at give forvaltningsobjekterne kontekst. Eksempler er FORM (se [FORM-online](#) og [KLE-online](#)), som er strukturerede beskrivelser af det offentlige forretning. I forbindelse med grunddata, kan de strukturerede data anvendes til at give forvaltningsobjekterne kontekst og sammenhæng - fx angivelse af hvilket forretningsområde, der er ansvarlig for opdateringen af data. Data, som opbevares i klassifikationskomponenten tildeles dobbelthistorik og revisionsspor, ligesom rammearkitekturen tilsigter distribution og opdatering af indholdet. Derfor er komponenten ideel til opbevaring af en lang række udfaldsrum - også simple lister.

Antagelser:

- Det antages, at klassifikationskomponenten bliver en permanent og entydig del af den offentlige it-arkitektur.
- Det antages, at staten vil anvende en tilsvarende komponent til udstilling af strukturerede data.
- Det antages, at de enkelte klasser i klassifikationskomponenten har identitet og bitemporale egenskaber, hvorved de kan anvendes distribuere.

Af disse grunde anbefaler reglerne i det følgende fx, at et forvaltningsobjekts forretningsområde, forretningsproces og forretningshændelse (se [regel 6.4](#)), lagres og udstilles som en reference til strukturerede data, udstillet i en klassifikationskomponent. Forvaltningsobjektets status kan også udstilles ved brug af klassifikationskomponenten.

3.3.3 Beskedfordeler/Beskedrevet arkitektur

[Link til reglen on-line - klik for at kommentere.](#)

Formålet med beskedfordeling (se http://en.wikipedia.org/wiki/Event-driven_architecture) er, at kæde administrative processer sammen på tværs af myndigheder. Når en administrativ forretningshændelse resulterer i en datahændelse, vil hændelsen ofte udløse en række andre administrative processer - typisk hos en anden myndighed. Relevante myndigheder skal derfor adviseres om datahændelsen ændringen i dataobjektet.

Beskedrevet arkitektur handler om, at det skal være muligt for et brugersystem at abonnere på datahændelser - i form af en besked. Formålet er, at modtageren af beskeden, fx et IT-system eller en sagsbehandler kan reagere på beskeden og igangsætte administrative processer som følge heraf. Fx kan en ændring i en persons adresse udløse en ændret ydelse (Effektiv Sagsbehandling og Kontrol ([ESK](#)) forudsætter en helt automatiseret løbende sagsbehandling, hvor en systemgenereret besked udløser en automatisk genberegning af fx en ydelse). Dette forudsætter, at beskeder om ændringer i data kan distribueres på en meningsfuld måde, hvor det er muligt at opsætte filtre (abonnementer), som kan fordele beskeder til de relevante modtagere.

Indeholder beskeden kun oplysning om selve datahændelsen, vil modtageren selv skulle analysere sig frem til, i hvilket omfang hændelsen har relevans for modtagerens forretning. Ved en ændring af en persons adresse, er en sagsbehandler fx nødt til at vide, om ændringen skyldes, at personen faktisk er flyttet, eller der blot er tale om at vejen har fået nyt navn. Langt større kvalitet opnås derfor, hvis beskeden indeholder informationer om den *forretningshændelse* (i virkeligheden), som udløste ændringen samt om det *forretningsområde* og den *forretningsproces*, som har været involveret i ændringen i data. Derved kan beskeden fordeles til den relevante anvender og indgå i en relevant proces i modtagerorganisationen. Automatiseret beskedfordeling kan derfor forudsætte, at disse informationer er til stede i den rigtige form, således, at de kan understøtte distributionen af beskeder.

Antagelser:

- Det antages, at datahændelser i grunddata skal afføde hændelsesbeskeder, som skal distribueres af Datafordeleren
- Det antages, at disse beskeder skal tilflyde den fælleskommunale rammearkitektur/serviceplatform
- Der gøres ingen antagelser om den infrastruktur, som skal udsende og modtage beskeder, hvorfor der ikke i dette dokument er yderligere behandling af teknisk protokol eller aftalegrundlaget for beskedfordeling.

Derfor opsætter regel [6.4](#) en række forslag til, hvordan datamodellen kan understøtte systematisering og udstilling af egenskaber ved datahændelsen (forretningshændelse, -område og -proces) som kan understøtte automatiseret beskedfordeling.

Kapitel 4 - Anvendelse af modelregler

Her forklares, hvordan modelreglerne er opbygget, og hvordan de skal efterkommes.

4.1 Reglerne er enten krav eller anbefalinger

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line - klik for at kommentere.](#)

Modelreglerne specificeres som enten krav eller anbefalinger:

- Regler angivet med "**skal**" er krav, som skal efterkommes.
- Regler angivet med "**bør**" er anbefalinger, som bør efterkommes, men der er ikke krav herom.

Se desuden bilag 1, som giver en oversigt over alle regler med angivelse af, om der er tale om krav eller anbefalinger.

Reglerne er ikke udtømmende - hvor et domæne er bundet af regler som er specificeret i andre sammenhænge, kan det følge disse, når de ikke er i modstrid med reglerne i dette dokument.

4.2 Reglerne kan udbygges inden for forretningsdomænerne

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line - klik for at kommentere.](#)

De modelansvarlige kan vælge at skærpe reglerne eller udbygge egenskaberne efter behov. Ligeledes kan de modelansvarlige have behov for at specificere yderligere regler eller egenskaber, som er gældende for det/de pågældende forretningsdomæne(r). Fx vil de grunddata, som er omfattet af INSPIRE direktivet skulle indeholde egenskaber, som er generelle for INSPIRE-data, men ikke for øvrige grunddata.

4.3 Mønster for regler

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Modelreglerne bliver beskrevet efter følgende mønster:

Navn	Angiver navnet på reglen
Regel	Beskriver klart og præcist reglen
Rationale	Beskriver forretningsværdien ved at følge reglen
Implikation	Beskriver hvilken indvirkning reglen har på forretning og teknisk implementering

Reglerne omhandler generelt den egentlige modellering - de udtaler sig om modelentiteterne, deres attributter, og modellen som sådan.

Rationalet for reglen vil typisk være at finde i et argument omkring dataobjekterne - modelleringen skal sikre, at et specifikt dataindhold kan udstilles.

Efter reglen vil der i nogle tilfælde være et kort praktisk eksempel, der viser, hvordan man kan anvende reglen.

Kapitel 5 - Generelle modelregler

De generelle modelregler vedrører udformning af datamodellen. Formålet med reglerne er, at sikre det niveau af ensartethed i domænemodellerne med hensyn til diagrammering, som er nødvendigt for at kunne etablere en samlet grunddatamodel.

5.1 Modeller skal udarbejdes som UML-klassediagrammer

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Regel

Datamodeller for grunddata skal beskrives i UML (Unified Modeling Language) version 2.4.1 som UML-klassediagrammer.

Rationale

En samlet og sammenhængende model forudsætter anvendelsen af et fælles modelleringssprog. UML er valgt, fordi det er et internationalt anerkendt modelleringssprog, hvor den overordnede forståelse af måden at relatere elementer til hinanden er fastlagt.

Implikationer

Der skal for alle grunddata udarbejdes UML-klassediagrammer med klasser, attributter, relationer og kardinaliteter samt tilhørende dokumentation. Der er krav om attributkomplethed, således at al den information, der udstilles som grunddata, skal kunne findes i grunddatamodellen.

Der henvises til uml.org for yderligere information. Almene UML symboler og notationer forklares ikke yderligere i dette dokument.

5.10 Referencer til klassifikationer, forretningsmodeller og organisationsmodeller bør anvendes

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Regel

Referencer til ekstern information bør så vidt muligt være referencer til publicerede klassifikationer, forretningsmodeller og organisationsmodeller.

Rationale

De fleste data-objekter skal relateres til klassifikationer og andre typer af strukturer, for derved at give objektet kontekst og sammenhæng. For at styrke genbrugelighed og fremfinding, bør disse referencer pege på eksisterende, publicerede og strukturerede datasæt, som fx FORM. Tilsvarende kan det være relevant, at pege på specifikke organisatoriske enheder - disse kan med fordel genbruges fra publicerede organisationsmodeller. Se endvidere [afsnit 3.3](#).

Der eksisterer ikke på nuværende tidspunkt en infrastruktur, som understøtter sådanne strukturerede data. Derfor kan det være nødvendigt - eventuelt som en migrationsstrategi - at give data sammenhæng på simple måder - fx med reference til en liste over organisatoriske enheder. På et senere tidspunkt kan sådanne lister så importeres i infrastruktur-komponenter, og referencerne tilpasses.

Implikationer

Området er under udvikling, så derfor kan der ikke opstilles en udtømmende implikation.

På samme måde kan denne regel ikke entydigt specificere konkret modellering eller danne grundlag for udvikling af konkret infrastruktur.

Anbefalinger:

Ved angivelse af:	Brug:
Forretningsområde:	FORM:Anbefales generelt anvendt ved angivelse af fællesoffentligt forretningsområde. Anbefales specifikt anvendt som udfaldsrum for attributten 'forretningsområde' i regel 6.4 .
Organisatorisk enhed:	ORGANISATION:Anbefales generelt anvendt ved referencer til organisatoriske enheder. Anbefales specifikt anvendt som udfaldsrum for attributterne registreringsaktør og virkningsaktør i regel 6.3 .
Strukturerede data:	KLASSIFIKATION: Anbefales generelt anvendt ved referencer som kræver en mere kompleks struktur, end enumeration eller kodeListe kan understøtte. Anbefales specifikt anvendt som udfaldsrum for attributterne status i regel 6.2 og forretningsproces i regel 6.4 .

5.2 UML-modellen skal organiseres i pakker

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Regel

UML-modellen skal organiseres i pakker, med en pakke for hvert forretningsdomæne.

Rationale

En logisk organisering af modellens elementer i pakker gør det mere overskueligt at navngive og referere til elementer.

Implikationer

Hver domænemodel placeres i en UML-pakke. En pakke kan have underpakker. Hvor det er relevant, gøres elementer offentlige ('public'), således at elementer i andre pakker kan referere til dem. Grunddatasekretariatet sørger desuden for, at der kan refereres til UML-pakker med elementer, som indgår i generelle egenskaber (se [kapitel 6](#)), samt til pakker udviklet af ISO, fx standardiserede datatyper ([se regel 5.5](#)). Se endvidere [fodnote til afsnit 2.1](#)

5.3 Modelentiteter skal genbruges

Version: 1.1.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Regel

Modelentiteter skal genbruges på tværs af grunddatamodellen.

Rationale

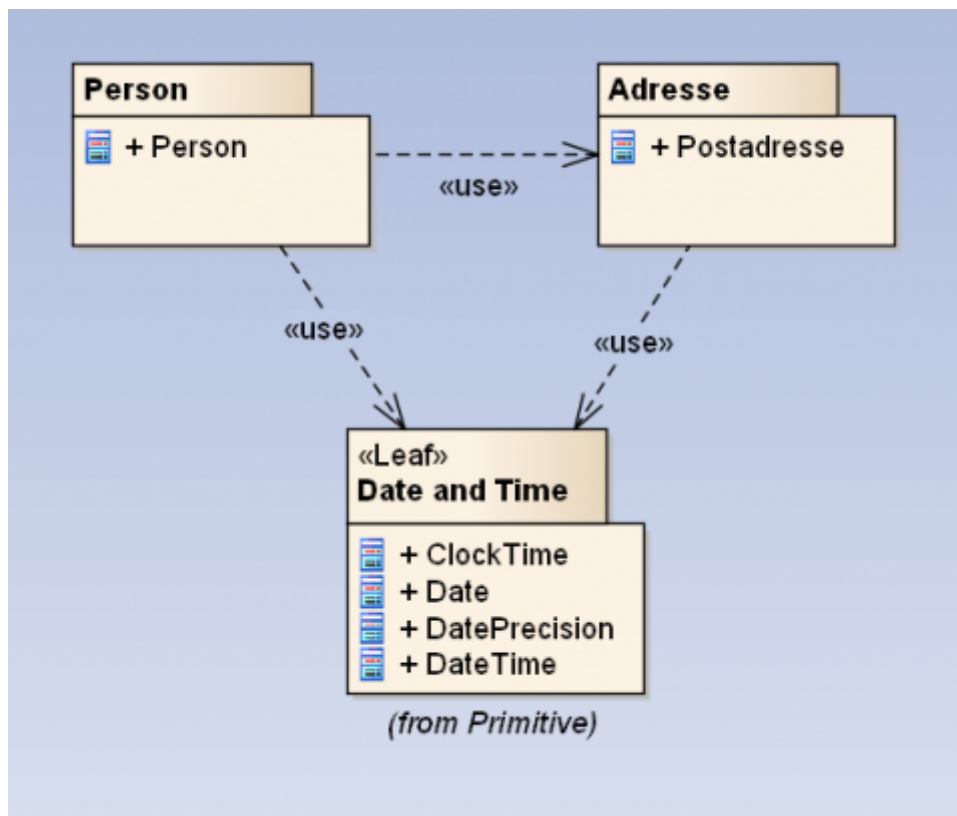
Det er en forudsætning for grunddataprogrammet, at modelentiteter genbruges på tværs af grunddata for at sikre sammenhæng og undgå redundant vedligehold af data.

Implikationer

Som modelansvarlig skal man modellere modelentiteterne for sit eget forretningsdomæne samt sørge for at relatere til modelentiteter i andre forretningsdomæners UML-pakker. [Se endvidere fodnote til afsnit 2.1](#)

Eksempler

Hvis fx person-domænet har behov for at modellere "Adresse", skal modelleringen referere/genbruge («use») den korrekte modelentitet inden for Adresse-domænet



Figur 5 - Pakken "Person" anvender («use») pakkerne "Adresse" og "Date and Time" (fra ISO/TC 211 Harmonized Model) for at anvende klasserne Postadresse og DateTime.

5.4 Attributter og relationer skal modelleres fyldestgørende

Version: 1.1.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Regel

Attributter og relationer mellem UML-elementer skal modelleres fyldestgørende - dvs. at attributter modelleres med multiplicitet og datatype, relationer modelleres med multiplicitet, retning og roller.

Rationale

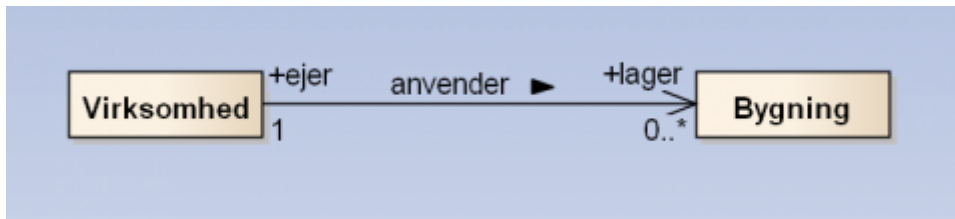
For at modellen skal være entydig samt danne basis for semantisk forståelse af data, er det nødvendigt, at attributterne og relationerne er beskrevet grundigt.

Implikationer

Attributter skal modelleres med en eksplicit type - se [regel 5.5 Standardiserede datatyper skal genbruges](#) - samt med multiplicitet dvs. antalsregler for, hvor mange værdier attributten kan rumme på en enkelt instans af klassen. Relationer (andre end generalisering/specialisering) mellem UML-elementer (associationer, kompositioner, aggregeringer) skal modelleres med navn, læseretning, navigabilitet, navngiven rolle/relation-ende samt multiplicitet (multiplicitet skal ikke angives for generalisering/specialisering). Se endvidere fodnote til [afsnit 2.3](#)

Eksempler

På figuren ses hvordan roller, læseretning, navigabilitet og multiplicitet udtrykker en kompleks association mellem to dataobjekter, som ellers ikke vil kunne kommunikeres i regi af modellen.



Figur 6 - Virksomhed i rollen som Ejer anvender Bygning som Lager. En virksomhed kan være alt fra lagerløs til lagermonopol (multilageret) - en Bygning kan kun have én ejer.

5.5 Standardiserede datatyper skal genbruges

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Regel

Alle attributter skal enten tildeles en standardiseret datatype eller en datatype, der er modelleret som et UML-element i samme eller en anden pakke.

Ved brug af en standardiseret datatype skal der henvises til ISO 19103, hvor en række standardiserede datatyper er samlet og modelleret i UML. ISO 19107 anvendes til geografi.

Rationale

ISO giver en anerkendt ramme for standardiserede datatyper. Anvendelse af en standard for datatyper gør det nemmere at bygge snitflader, der udstiller grunddata som nemt anvendelige services.

Implikationer

Standardiserede datatyper i modellen skal hentes fra ISO/TC 211 Harmonized Model.

Denne er en samling af datatyper, som primært anvendes af geografi-relaterede modelleringsprojekter - fx INSPIRE. Ikke desto mindre indeholder ISO/TC 211 Harmonized Model også generelt anvendelige datatyper (CharacterString, Integer, DateTime mv.) modelleret som UML.

Hvis domænet ikke finder typerne i ISO/TC 211 Harmonized Model anvendelige, kan det opbygge sit eget typebibliotek, som skal publiceres sammen med domænemodellen.

ISO/TC 211 Harmonized Model findes frit tilgængelig på adressen <https://inspire-twg.jrc.it/svn/iso/>, hvorfra den kan importeres i et modelleringsværktøj. Dokumentet "Modelleringsværktøj", som Grunddatasekretariatet forventes at publicere, vil beskrive anvendelsen af ISO/TC 211 Harmonized Model i detaljer.

5.6 UML-stereotyper skal anvendes

Version: 1.1.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Regel

Alle UML-elementer tildeles en UML-stereotype.

Rationale

På sigt skal det være muligt, at fortolke modellen til andre modeltyper samt fx datasnitflader. UML kan ikke i sig selv udpege elementernes rolle (modelentitet, datatype, enumeration) i modellen, hvorfor det er nødvendigt, at udvide modellen med disse roller. UML-stereotyper er udvidelser af modelleringssproget, som gør det muligt at specificere yderligere egenskaber, samt at kategorisere model-elementerne. Med stereotyper kan man udpege specifikke klasser som havende bestemte roller i datamodellen, hvilket igen gør det muligt for et eksternt værktøj, at fortolke modellen til fx datasnitflader og ontologier. Stereotyperne tilføjer roller til elementerne og strukturerer de tagged values, som indeholder dokumentationen af modelementerne (se Note om tagged values).

Implikationer

Følgende stereotyper anvendes i Grunddatamodellen*:

- **«DKObjekttype»:** Alle forvaltningsobjekter skal modelleres som UML-klasser med stereotypen «DKObjekttype».
- **«DKEnumeration», «DKKodeliste», «DKKlassifikation»:** Attributter med kodede værdier skal modelleres med UML-elementer med stereotyperne «DKEnumeration», «DKKodeliste» eller «DKKlassifikation», som værdisæt - [se Note](#) om kodede værdier.
- **«DKDatatype»:** Attributter, som ikke er kodede værdier og som ikke tildeles en standardiseret datatype fra ISO (se [regel 5.5](#)) skal modelleres med en datatype i samme eller en anden pakke, med stereotypen «DKDatatype», som værdisæt. Se endvidere fodnote til afsnit 2.3
- **«DKEgenskab»:** Attributter og associationsender i Grunddatamodellen har stereotypen «DKEgenskab».
- **«DKDomænemodel»:** "Rod"-pakken i en afleveret domænemodel markeres med «DKDomænemodel»

Grunddatasekretariatet sørger for, at en UML-profil indeholdende stereotyperne er specificeret. Se <http://digitaliser.dk/group/2494445>

NOTE om tagged values

UML-elementer kan indeholde ekstra attributter, kaldet tagged values. Ved at anvende tagged values på stereotyperne, kan de på en gang tilføjes alle klasser, der har stereotypen. Disse tagged values kan indeholde specifikke instrukser til det software, som skal fortolke modellen til for eksempel XML schema - se fx [INSPIRE GCM](#) afsnit 9.6.3. På nuværende tidspunkt er der ikke overblik over, hvilke instrukser, der er behov for. Disse vil eventuelt kunne tilføjes senere med minimale ændringer af de godkendte modeller. Stereotyperne indeholder tagged values til dokumentation af modellens elementer. Disse dokumentations-tagged values (Definition, Note, Alternativt navn, Lovgrundlag, Eksempel) beskrives i [Regel 5.9](#)

NOTE om kodede værdier

Kodede værdier kan modelleres på tre måder:

«DKEnumeration», hvor værdierne findes eksplicit som en liste i modellen. Bruges typisk, hvor udfaldsrummet er velforstået, og hvor der ikke forventes ændringer hurtigere end modellens overordnede versioneringscyklus gør muligt. Eksempelvis ugedag: {mandag, tirsdag, onsdag, torsdag, fredag, lørdag, søndag}

«DKKodeliste», hvor værdierne opbevares eksternt, men tilgængeligt på internettet. Se [INSPIRE GCM](#) afsnit 9.4.9, 9.5.2 samt Annex G. Dette giver mulighed for dynamisk tilpasning og udvidelse af kodelisten efter behov, samtidig med, at der er et vist mål af styring af historik og proveniens for værdierne.

«DKKlassifikation», hvor værdierne opbevares i en taksonomi - for eksempel i en klassifikationskomponent, som følger OIO-standard for Klassifikation. Klassifikationskomponenten kan udstyre værdierne med metadata, opdaterings-metadata og dobbelthistorik. En klassifikationskomponent indgår i en fremtidig KL/KOMBIT rammearkitektur. Se [afsnit 3.3.2](#).

For enkelte domæner gælder der regler for, hvordan eksterne data styres og håndteres - eksempelvis INSPIRE reglerne, som gælder for nogle af de geodata, som også er grunddata. Sådanne regler kan overholdes samtidig med, at modelreglerne overholdes.

Domænet skal varetage, at de eksterne data, som domænedata refererer til, er tilgængelige og af den fornødne kvalitet. Ligeledes skal domænet overholde de regler, som gælder for arkivering, og som kan tilsige at eksterne data skal kunne afleveres til arkiv sammen med domænedata.

* Modellen kan udvides med domæne-specifikke stereotyper ligesom Grunddata-stereotyperne kan extendes i domænet fx geodatas DKFeaturetype, som extender DKObjekttype

5.7 Navngivningsregler skal følges

Version: 1.1.0 **Status:** Gældende - [Link til reglen on-line - klik for at kommentere](#).

Regel

Elementer i UML-modellen skal navngives på ensartet vis i hele grunddatamodellen. Navngivning skal være entydig inden for domænet - i praksis inden for UML-pakken.

Rationale

En ensartet navnekonvention giver datamodellen et ensartet udtryk og gør det nemmere at identificere og skelne de forskellige klasser af modelementer fra hinanden.

Implikationer

Klasser og associationer skal navngives entydigt inden for deres pakke.

Klasser, attributter og associationer navngives efter følgende skema:

- **Elementer som repræsenterer forvaltningsobjekter (med stereotypen «DKObjekttype»), som er datatyper («DKDatatype»), klassifikationer («DKKlassifikation»), enumerationer («DKEnumeration») eller kodelister («DKKodeliste»)** navngives med "UpperCamelCase" - dvs. med stort begyndelsesbogstav i både første ord og alle efterfølgende ord i navnet og uden anvendelse af mellemrum i navnet.
- **Attributter, associationer og relationsender** navngives med "lowerCamelCase" - dvs. med lille begyndelsesbogstav i første ord samt stort begyndelsesbogstav i alle efterfølgende ord i navnet og uden anvendelse af mellemrum i navnet *.

*Hvor navnene består af akronymer kan man fravige denne regel; fx. "NUTS1værdi", "CPRnummer"

NOTE: Af hensyn til modellens anvendelse og transformation i software, som ikke kan håndtere de danske tegn Æ, æ, Ø, ø, og Å, å, kan disse eventuelt translittereres til hhv "Ae", "ae", "Oe", "oe", "Aa" og "aa".

5.8 Sprogregler skal anvendes

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line - klik for at kommentere.](#)

Regel

Dansk skal anvendes ved navngivning af elementer, som indgår i generelle egenskaber.

Den modelansvarlige fastsætter det sprog, som anvendes ved navngivning af domænets elementer.

ISO-standarder følger deres engelske betegnelser (fx "Integer" og "codeList").

Datamodellen dokumenteres på dansk, se [regel 5.9](#).

Rationale

Idet grunddata er det offentlige forvaltningsgrundlag, beskrives de som udgangspunkt på dansk, som er forvaltningssproget. Nogle grunddata kan imidlertid være underlagt internationale forpligtelser, som forudsætter brug af andre sprog. Fx er en række grunddata omfattet af EU-direktivet INSPIRE og skal som følge heraf genbruge UML-elementer, hvor sproget er engelsk. Det er derfor op til den modelansvarlige for domænet at træffe beslutning om sprog for domænespecifikke elementer.

Implikationer

Det er op til den modelansvarlige at vælge sprog for domænets elementer af modellen. Der kan derfor forekomme modeller, hvor der er en blanding af dansk og andre sprog.

5.9 Datamodellen skal dokumenteres

Version: 1.1.0 **Status:** Gældende - [Link til reglen on-line - klik for at kommentere.](#)

Regel

Datamodellen skal dokumenteres gennem beskrivelser af elementerne i UML-modellen.

Rationale

Dokumentationen gør det muligt for modellens brugere at forstå modellens elementer. Både når man udvikler og anvender modellen, er det essentielt at kommunikere og forstå betydningsindholdet af de enkelte dele af modellen. Det at dokumentationen er indlejret i datamodellen muliggør automatisk dannelse af et katalog over klasser, attributter og relationer. Yderligere kan dokumentationen indgå i datasnitfladerne, hvis der er ønske herom.

Implikationer

Modellens klasser og deres attributter beskriver forretningsobjekter og deres egenskaber, ligesom de roller, objekterne har i forhold til hinanden beskrives som roller/relationer i hver ende af en association. Disse forretningsobjekter, egenskaber og roller skal beskrives i overensstemmelse med den begrebsmodellering, der er i domænet.

Dokumentation i UML med tagged values

Klasser, attributter og roller/relationsender dokumenteres ved brug af tagged values (se [Note om tagged values](#)), som specificeret i tabellen nedenfor. De enkelte tags i dokumentationen modsvarer i videst muligt omfang vidensmodelleringsvokabulariet [SKOS](#). Denne strukturerede måde at dokumentere på, muliggør fleksibel generering af tekstdokumentation på baggrund af modellen, ligesom dokumentationen med stor præcision kan transformeres ind i datasnitflader og afledte modeller.

Tagged values som bruges til dokumentation i datamodellen:			
Tag definition	SKOS-ækvivalent	Krav	Indhold
definition	skos:definition	Krævet	Objektets/egenskabens/rollens definition. Kort tekst, som entydigt beskriver objektet/egenskaben/rollen -kan også indeholde et afsnit med en længere beskrivelse, fx formål, referencer og kilde.
note	skos:note	Valgfri	Uddybende beskrivelse af objektet/egenskaben/rollen
alternativtNavn	skos:altLabel	Valgfri	Andre navne, som objektet/egenskaben/rollen kan have
lovgrundlag	--	Valgfri	Angivelse af det lovgrundlag, som hjemler indsamlingen af data for objektet/egenskaben/rollen
eksempel	skos:example	Valgfri	Eksempler på anvendelse af objektet/egenskaben/rollen

Værdierne angives i ren tekst

De tagged values, som nævnes i skemaet, er indeholdt i Grunddatamodellens stereotyper, som defineres i [regel 5.6 UML-stereotyper skal anvendes](#).

Værktøjsunderstøttelse: I den MDG og base project, som distribueres af modelsekretariatet (<http://digitaliser.dk/resource/2742281>), er de relevante tagged values indføjet i stereotyperne i en separat gruppe - Grunddata:dokumentation

[Figur som viser Class properties med tagged values] - kommer senere

NOTE: Nogle domæner anvender store mængder metadata, som har til formål at indgå i brugervendt information om data (forklaringer til rapporter, hjælpetekst på web-sider etc.) Det vil ikke være formålstjenligt, at indlejre disse metadata i selve modellen, men de kan modelleres og distribueres som alle andre data og fordeles gennem den generelle data-arkitektur.

Eksempler

[Eksempler på udfyldte tagged values - enten grafisk i EA eller som tabeller for klasse, attribut og relation end]

Kapitel 6 - Regler om generelle egenskaber

Kort om Generelle Egenskaber

6.1 Alle modelentiteter skal modelleres med persistent, unik identifikation

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Regel

Alle entiteter skal modelleres med persistent, unik identifikation.

Rationale

Alle data-objekter, skal have et globalt unikt id, som ikke ændres i data-objektets levetid. Det er nødvendigt at have en unik identifikation af et data-objekt på tværs af grunddatamodellen, for at sikre en fælles høj datakvalitet.

Ofte vil data-objektet, ud over den unikke identifikation, have en eller flere forretningsnøgler, fx har en matrikel et matrikelnummer. Men forretningsnøglerne kan ikke stå alene, da grunddatamodellen generelt skal understøtte historik, hvilket betyder, at data-objektet kan have forskellige forretningsnøgler over tid, ligesom den samme forretningsnøgle løbende kan indgå i flere forvaltningsobjekter.

Derfor er det essentielt, at objektets identifikation er persistent i hele data-objektets levetid.

Ved at lade ID være en HTTP-URI (se Implikation) bliver URI'en "opløsbar", se

http://en.wikipedia.org/wiki/Dereferenceable_Uniform_Resource_Identifier

Dette giver ID'en to funktioner:

- Global entydig identifikation af en entitet på tværs af Grunddatamodellen og på tværs af øvrige modeller.
- Reference direkte til data-objektet, således at et givet data-objekt potentielt kan adresseres direkte - også vha. en generisk web-applikation.

Implikationer

Alle entiteter skal modelleres med attributten 'id'

id

Betydning	Unik identifikation af objektet
Værdi	Objektets unikke id
Datatype	Identifikation
Krav	Obligatorisk

Datotypen **Identifikation**.

Datotypen er en del af Grunddatatyperne, som publiceres og vedligeholdes af Grunddatasekretariatet; se <http://digitaliser.dk/group/2494445>

Attributterne i Identifikation kan kombineres til en HTTP-URI, som specificeret i [IETF RFC 3986](#) (http-scheme). Denne vil være universelt unik, og vil være opløslbar.

En HTTP-URI kan sammensættes på følgende måde - baseret på IETF RFC 3986, afsnit 3, Syntax Components:

Skemaangivelse + "://" + Autoritet + "/" + Sti + "#" + Objektidentifikator

Indhold	
Komponent	Værdi
Skemaangivelse	"http"
Autoritet	"data.gov.dk"
Sti	Valgfri, men se nedenfor
Objektidentifikator	Valgfri, men se nedenfor

HTTP-URI'en skal i sin helhed være universelt unik.

Dette kan opnås på en række måder:

- Manglende eller generisk Sti, universelt unik ObjektidentifikatorEksempel:
<http://data.gov.dk#e1f9a650-21c7-11e3-8224-0800200c9a66>
- Unikt afgrænsende Sti, lokalt unik ObjektidentifikatorEksempel:
<http://data.gov.dk/lufthavnskoder/IATA#CPH>
- Unikt afgrænsende Sti, universelt unik ObjektidentifikatorEksempel:
<http://data.gov.dk/person#08f88d52-4add-4ec6-9a80-18caaf0f0e7b>

Det er domænets ansvar, at sikre, at den samlede, identificerende HTTP-URI er universelt unik.

Det anbefales, at Objektidentifikatoren er af typen UUID (Universally Unique Identifier - [IETF RFC 4122](#)) - på den måde bliver identifikationen universelt unik uanset hvilket namespace, der anvendes - dette vil være relevant i et fremtidigt, løst koblet data-scenarie, hvorfor der også på sigt vil være behov for, at alle datakilder udstyrer data med UUID.

Komponenterne Skemaangivelse, Autoritet og Sti kan sammensættes til et Namespace, som lagres i den ene attribut - namespace i datatypen. Objektidentifikatoren lagres i den anden attribut, som lokalId

Attributter

namespace

Betydning	Identifikation af et namespace inden for hvilket lokalId er unik
Værdi	En HTTP-URI uden Objektidentifikator (Skemaangivelse + “://” + Autoritet + “/” + Sti)
Datatype	CharacterString
Krav	Obligatorisk

localID	
Betydning	Identifikation af objektet
Værdi	Objektets id
Datatype	CharacterString
Krav	Obligatorisk

6.2 Alle modelentiteter skal modelleres med status

Version: 1.0.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Regel

Alle modelentiteter skal modelleres med status, der klart angiver, hvor et forvaltningsobjekt er i sin livscyklus.

Rationale

Forvaltningsobjekter gennemløber typisk en livscyklus. Fx kunne en livscyklus for en bygning være: “Forslag > Under projektering > Under opførelse > Ibrug > Under nedrivning > Historisk”. Forretningsdomænet kan opstille regler for hvilke statusser, der er gyldige for et givet objekt, og for, hvordan forvaltningsobjektet kan gennemløbe disse.

Disse tilstande skal placeres og udstilles i et vedtaget udfaldsrum, som modelentiteten skal referere til.

Dataobjektets status udtrykker dataobjektets relevans for databrugeren. Forretningsmæssigt giver det god værdi at udstille et dataobjekts status eksplicit, frem for at lade databrugeren analysere sig frem til informationen. Anvendelsen af status er med til at sikre en høj datakvalitet og potentielt nedbringe udviklingsomkostninger, da der skal implementeres mindre forretningslogik og fejlhåndteringslogik.

Implikationer

Alle modelentiteter skal have attributten ‘status’

Status	
Betydning	Forvaltningsobjektets status
Værdi	En status
Datatype	enumeration, kodeliste, klassifikation
Udfaldsrum	Domænespecifik liste, værdien må ikke være tom
Krav	Obligatorisk

Der skal i domænemodellerne defineres statustilstande for alle forvaltningsobjekter. Disse statustilstande defineres af den modelansvarlige, og publiceres i den fælles model. Tilstandene kan modelleres som enumeration, kodeliste eller klassifikation (se Note om kodede værdier i [afsnit 5.6](#)).

NOTE: Ideelt bør statusser betegnes med de reelle tilstande, ikke senest overståede milepæl. Fx "i brug" frem for "ibrugtaget" - ibrugtaget vil objektet jo vedblive at være.

6.3 Alle modelentiteter skal understøtte dobbelthistorik og angivelse af aktører

Version: 1.1.0 **Status:** Gældende - [Link til reglen on-line - klik for at kommentere.](#)

Regel

Alle modelentiteter skal modelleres med angivelse af registrering, virkning og aktører

Rationale

Dobelthistorik:

På tværs af grunddata er der behov for at implementere dobbelthistorik, for at kunne understøtte et samlet krav om revisionsspor. Dataobjekter skal med andre ord kunne rekonstrueres på en måde, hvor der er styr på objektets sammensætning eller tilstand til et givet tidspunkt. Formålet hermed er bl.a. at dokumentere det konkrete historiske beslutningsgrundlag i forbindelse med fx sagsbehandling.

Dobelthistorik modelleres ved hjælp af bitemporale egenskaber. Det dobbelte består i, at to tidsaspekter "virkningstid" og "registreringstid" håndteres i sammenhæng.

Registreringstid:

Tidsrummet fra versionen registreres i databasen, indtil den enten erstattes af en nyere version eller afregistreres

Virkningstid: Tidsrummet, hvor en given version af data svarer til de forhold i virkeligheden, som versionen afbilder.

Aktører:

Oplysning om hvilke aktører, der er ansvarlig for datas indhold, tilfører sporbarhed i forbindelse med revision og anvendelse af data. Aktøren kan være en af en række forskellige typer, fx en organisation, et it-system, en arbejdsfunktion eller en konkret bruger.

Implikationer

Et data-objekt kan bestå af en række (1-*) versioner (ændres en enkelt attributværdi, betragtes dataobjektet som ændret og dermed versioneret). Alle versioner betragtes som dele af et "stamdataobjekt", og har den samme Identifikation.

Alle versioner skal være karakteriseret ved hjælp af deres registreringstid og deres virkningstid. Disse tidsaspekter modelleres ved anvendelse af attributterne registreringFra, registreringTil, virkningFra og virkningTil.

Enhver version af et stamdataobjekt identificeres entydigt af objektets unikke identifikation kombineret med attributten registreringFra.

Når en bruger til et bestemt formål skal fremfinde den eller de relevante versioner af et objekt, anvendes objektets identifikation samt en kombination af registreringstid, virkningstid og/eller objektets status.

Virkningstidsrummet skal fortolkes sådan at virkningsintervallet er inklusiv starttidspunktet men eksklusiv sluttidspunktet.

Hvorvidt der kan være "huller" i Virkningstiden - perioder hvori virkeligheden ikke er afspejlet i data - er op til det enkelte datdomæne at beslutte og dokumentere.

Til hver version af et dataobjekt skal der knyttes aktører i betydningen:

- Reference til den aktør, der afstedkommer iværksættelse af virkningsperioden
- Reference til den aktør, der har foretaget registreringen

Værdien kan være en reference til fx en organisation, et system eller en sagsbehandler se afsnit [3.3.1](#) og regel [5.10](#).

Fuld temporal rekonstruktion af dataobjekter forudsætter, at alle ændringer i dataobjektet gemmes med de nødvendige tidsmarkeringer. Derfor indeholder denne regel et krav om, at data opbevares i registre, som er implementeret på en måde som gør dette muligt. En sådan implementering indbefatter dels en fysisk datamodel som tilføjer en ny række til datatabellen for hver ny version a data, på en måde så alle ændringer kan genfindes dels forretningsregler for, hvordan tidsmærkerne tilknyttes. Et dokument, som uddyber hvordan dette kan gøres i et RDB/SQL miljø kan downloades [her](#)

Attributter:

Hver modelentitet med stereotypen DKObjekttype - modsvarende et forretningsobjekt - modelleres med følgende attributter:

Registrering:

registreringFra	
BETYDNING	Tidspunktet hvor registreringen er foretaget
VÆRDI	Tidspunkt
DATATYPE	DateTime (ISO 8601), værdien må ikke være tom
KRAV	Obligatorisk

registreringTil	
BETYDNING	Tidspunktet hvor en ny registrering er foretaget på dataobjektet, og hvor denne version således ikke længere er den seneste.
VÆRDI	Tidspunkt
DATATYPE	DateTime (ISO 8601), værdien kan være tom
KRAV	Obligatorisk

registreringsaktør

BETYDNING	Den aktør der har foretaget registreringen
VÆRDI	Angivelse af en aktør fx som en reference til en organisationsmodel (se regel 5.10)
DATATYPE	Domænespecifik aktør, værdien må ikke være tom
KRAV	Obligatorisk

Virkning:

virkningFra	
BETYDNING	Tidspunktet hvorfra forvaltningsobjektet har virkning
VÆRDI	Tidspunkt - virkningsperioden inkluderer dette tidspunkt
DATATYPE	dateTime (ISO 8601), værdien må ikke være tom
KRAV	Obligatorisk

virkningTil	
BETYDNING	Tidspunktet hvor forvaltningsobjektets virkning ophører
VÆRDI	Tidspunkt - virkningsperioden stopper umiddelbart før dette tidspunkt
DATATYPE	dateTime (ISO 8601), værdien kan være tom
KRAV	Obligatorisk

virkningsaktør	
BETYDNING	Den aktør der har afstedkommet forvaltningsobjektets virkning
VÆRDI	Angivelse af en aktør fx som en reference til en organisationsmodel (se regel 5.10)
DATATYPE	Domænespecifik aktør, værdien må ikke være tom
KRAV	Obligatorisk

Undtagelse

CVR's modeller er undtaget fra regel 6.3 fsva. attributterne registreringFra og registreringTil, da CVR ikke indeholder de pågældende data

6.4 Alle modelentiteter bør understøtte beskedfordeling

Version: 1.1.0 **Status:** Gældende - [Link til reglen on-line](#) - [klik for at kommentere](#).

Regel

Modelentiteterne i grunddata bør modelleres således, at dataobjektet kommer til at indeholde informationer, som kan forbedre kvaliteten af hændelsesbeskeder, der udsendes i forbindelse med opdatering af dataobjektet. Disse informationer omfatter den forretningsmæssige kontekst, hvori dataobjektet opdateredes, samt den bagvedliggende forretningsmæssige årsag til opdateringen.

Rationale

Automatiseret beskedfordeling beskrives i kapitel 3.3.3. Der findes på nuværende tidspunkt ikke et samlet billede af, hvordan beskedfordeling skal foregå, hvorfor denne regel er formuleret med det formål, at beskrive rammerne for de data der er nødvendige for beskedfordeling. Yderligere arbejde med fællesoffentlig beskedfordeling kan meget vel resultere i justering af denne regel. Automatiseret beskedfordeling forudsætter, at det for hver ændring af data registreres hvilken forretningsammenhæng, der ligger til grund for ændringen.

Forretningsammenhængen beskrives med tre parametre:

- forretningshændelse, som betegner den begivenhed i virkeligheden (se afsnit 1.3.2), som udløste ændringen i data
- forretningsområde - den del af den offentlige forretning, som håndterer hændelsen og derved udvirker ændringen i data
- forretningsproces - den manuelle eller it-understøttede proces, hvori forretningsområdet håndterer hændelsen

Hændelse, område og proces er udfaldsrum, som typisk vil skulle modelleres af domænet

- Forretningshændelser vil typisk (ligesom status - se regel 6.2) være specifikke for det enkelte forvaltningsobjekt - der skal opbygges en datastruktur, der afspejler forretningens viden om hvilke hændelser, der kan påvirke et forvaltningsobjekt
- Forretningsområdet kan specificeres ud fra FORM
- Forretningsprocesser forudsætter en kortlægning af domænets processer

De tre udfaldsrum kan modelleres mere eller mindre komplekst - som simple lister, indlejret i modellen eller som reference til eksterne data, enten i form af kodelister eller klassifikationskomponenter - se NOTE om kodede værdier i afsnit 5.6 samt afsnit 3.3.2. Den modelansvarlige må afgøre på hvilket niveau modelleringen bedst muligt modsvarer forretningskravene.

Implikationer

Hver modelentitet med stereotypen DKObjekttype - modsvarende et forretningsobjekt - kan modelleres med følgende attributter:

forretningsområde	
BETYDNING	Det forretningsområde som har opdateret dataobjektet
VÆRDI	Specifikation af et forretningsområde. Domænespecifik liste, værdien kan være tom
DATATYPE	enumeration, codeList, klassifikation
KRAV	Valgfri

forretningsproces

BETYDNING	Den forretningsproces, som har opdateret dataobjektet
VÆRDI	Specifikation af en forretningsproces. Domænespecifik liste, værdien kan være tom
DATATYPE	enumeration, codeList, klassifikation
KRAV	Valgfri

forretningshændelse	
BETYDNING	Den forretningshændelse, som afstedkom opdateringen
VÆRDI	Specifikation af en forretningshændelse. Domænespecifik liste, værdien kan være tom
DATATYPE	enumeration, codeList, klassifikation
KRAV	Valgfri