

Fleksibel modell for levende lover

Kirill Miazine
FØYEN Advokatfirma DA

24. september 2008

1 Innledning

I forrige nummer av Lov & Data skrev jeg om noe av problematikken knyttet til entydig identifikasjon av avgjørelsene fra Høyesterett. Denne gangen skal jeg se på noen spørsmål knyttet til en annen rettskildedefaktor – lovene.

En lov er som et levende vesen. En lov blir født ved sin kunngjøring, loven blir myndig i og med sin ikrafttredelse og den dør når den blir opphevet. I sin levetid vil en lov sannsynligvis oppleve både vesentlige forandringer i sitt utseende og i sitt innhold ved at nye paragrafer legges til, eksisterende bestemmelser endres eller det skjer omfattende interne restruktureringer. Eksempelsvis har åndsverksloven av 12. mai 1961 vært underlagt en rekke lettere plastiske operasjoner og større kirurgiske inngrep i løpet av de drøyt 50 år som har passert siden 1961, da den først ble kunngjort.

Slike endringer har noen interessante og mindre åpenbare følger. En bemerkelsesverdig egenskap ved lovene er at de fortsetter å leve i sitt gamle vesen også etter at det gjøres endringer. Bl.a. er det slik at såvel den nyeste som de gamle revisjonene er relevante ved avgjørelse av rettslige spørsmål. Et av årsakene er Grunnlovens § 97 som slår fast at *“Ingen Lov maa gives tilbagevirkende Kraft”*, og derav følger at forhold som fant sted på et gitt tidspunkt vil bli avgjort med utgangspunkt i ordlyden som lovene hadde da, selv om loven siden den gang hadde fått nytt innhold eller til og med er blitt opphevet. Det er også slik at de eldre bestemmelsene kan kaste lys over hvordan nyere lovgivning skal forstås.

2 Problemstilling og tema

En stor rolle i praksis spiller forskjellige versjoner av de forskjellige paragrafene og det kan fort oppstå forvirring dersom de forskjellige versjo-

nene inneholder vesentlige forskjeller i innholdet. I 1995¹ ble det foretatt “*omfattende redaksjonelle endringer*” i åndsverksloven², som bl.a. kom til uttrykk ved at § 24 ble flyttet foran og ble § 11, bestemmelsene i den gamle § 11 ble videreført i § 12, mens §§ 39 og 39a ville innholdsmessig dekke § 25 og § 49 første ledd og § 31 andre ledd³.

Hvordan vil en leser i 2008 reagere når han i en eldre publikasjon ser en henvisning til en paragraf som etter 1995 fikk et helt annet innhold enn den hadde før 1995? Enkelte vil oppleve en frustrasjon som kan sammenlinkes med den man får når det er en død lenke på Internett eller en uriktig henvisning til en dom i Retstidende i juridisk litteratur. Kai Krüger viser i sin artikkel i Jussens Venner 1988 s. 139 *Datamaskinutrustning og programvare – kontraktsvilkår* til “*åndsverklovens privatbruksregler i § 11*”. I loven, slik den lyder i dag, står reglene om eksemplarframstilling til privat bruk i § 12. En etterforskende leser eller en leser med kjennskap til opphavsrett vil nok klare å finne ut av det, mens andre blir forvirret, skuffet, frustrert. Kan man si at Krügers henvisning er feil?

Nei, man kan nok ikke si at Krügers henvisning i seg selv er feil, fordi den *er* riktig for perioden frem til § 11 endret sitt innhold. Men henvisningen er *ufullstendig*, Henvisningen er ufullstendig ved at den mangler en meget viktig dimensjonskomponent – tid.

Ved henvisninger til ressurser som ligger på Internett, hvor det er forventet at ressursene vil endre seg over tid, angir man en dato da dokumentet ble lest⁴. Det samme burde kanskje gjøres ved henvisning til lovene, hvor man ikke bare forventer, men vet at endringene vil komme. Krügers fullstendige referanse kan f.eks. være *åndsverklovens § 11 per 1. januar 1988*. Med *fullstendig* mener jeg kun at referansen på en entydig og dekkende måte identifiserer den bestemmelsen som Krüger siktet til, jeg mener ikke at referansen må se slik ut, det kan selvsagt brukes alternative formater, men tidskomponenten må ikke utelates.

3 Mål med innlegget

Dette er et innlegg om lov og data.

Jeg ser nærmere på problemstillinger knyttet til syntaktisk konsistens (eller snarere syntaktisk inkonsistens) i rettskildene som oppstår som følge av at lovenes innhold forandrer seg over tid. Slik inkonsistens

¹Jf. Lov om endringer i åndsverkloven m.m. nr. 27 av 2. juni 1995, i kraft fra 23. juni 1995

²Jf. Ot.prp. nr. 15 (1994-1995)

³Jf. merknadene til de enkelte paragrafene i Ot.prp. nr. 15 (1994-1995)

⁴Jf. f.eks. *Ofte stilte spørsmål om henvisninger og litteraturliste* på <http://www.uio.no/ujur/henvisninger/faq.php>

vil oppstå, f.eks. ved at en forskrift vil vise til en paragraf som tidligere inneholdt en hjemmel for å gi denne forskriften, men hvor hjemmelen av redaksjonelle grunner flyttes et annet sted i loven. Man kunne selvsagt oppdatere forskriftene og angi oppdatert og korrekt hjemmel, men slik redaksjonelt arbeid synes for noen å være overflødig i det “[d]et synes unødvendig å presisere at forskrifter gitt med hjemmel i gjeldende lov fortsatt vil gjelde, selv om hjemmelen av redaksjonelle grunner blir flyttet til en annen paragraf i loven”⁵.

Men primært – og det fremgår også av tittelen på innlegget – ønsker jeg å prøve å angi noen hovedtrekk i en datamodell som kan være egnet for å representere lover på en slik måte at deres levende natur blir respektert og en modell som vil tillate bruk av henvisninger hvor man ikke har en risiko at det etter en vis tid refereres til noe annet enn opprinnelig tenkt, jf. eksemplene ovenfor med henvisningen i Krügers artikkel og referanser til hjemlene i forskriftene til åndsverksloven. For å legge til rette for dette, må modellen kunne støtte henvisninger til de forskjellige versjonene av bestemmelsene. En praktisk nyttig følge av dette, er at det vil være mulig å instruere systemet om å vise hvordan lovene så ut f.eks. 1. januar 1970 og systemet vil da vise alle lover som gjaldt per 1. januar 1970, med korrekte henvisninger og innhold.

4 Lovens struktur

Det er hensiktsmessig å se nærmere på strukturen i lovene først. Jeg plukket ut en tilfeldig lov og vil basere min analyse på lov om tvangsfullbyrdelse av 26. juni 1992. Jeg kunne like gjerne fortsette med åndsverksloven, men tvangsfullbyrdelsesloven bruker flere nivåer i sin oppbygning og analysen vil derfor gi et mer fargerik bilde av hvordan lovene er bygget opp.

Umiddelbart ser vi at loven er bygget opp i en hierarkisk struktur og lovteksten er organisert ved hjelp av en rekke *beholdere*. Loven er i seg selv én slik beholder og består av 14 *kapitler* fordelt på 3 *deler*. I kapitlene 1–3, 5, 6 og 14 finner man *paragrafer*, mens i kapitlene 4 og 7–13 er paragrafene organisert i flere *kapitteldeler*. Overordnet har tvangsfullbyrdelsesloven således 5 nivåer med følgende holdere:

- Nivå 0 – Loven
- Nivå 1 – Del
- Nivå 2 – Kapittel
- Nivå 3 – Kapitteldel

⁵Jf. Ot.prp. nr. 15 (1994-1995)

- Nivå 4 – Paragraf

Av de nevnte beholdere er det kun *paragrafer* som inneholder selve lovteksten. De andre beholdere, inkludert loven, brukes for å organisere andre beholdere, herunder paragrafer, i en struktur.

Paragrafene består av ett eller flere *avsnitt* (som normalt kalles *ledd*), som i sin tur består av *setninger* (som normalt kalles *punktum*). Angivelse av flere alternative eller kumulative vilkår gjøres normalt i form av en liste med et eller flere *punkter*. § 1-9 om hvor visse formuesgoder skal anses for å være består av ett ledd, som inneholder en punktliste med 4 punkter (a–d). Hvert av disse består av ett eller flere setninger.

For å gjøre analysen mer fullstendig, kan det føyes til at setningene består av *ord*, som igjen består av *bokstaver*.

Det gjøres en utstrakt bruk av *henvisninger* i lovene: såvel internt i en lov som til bestemmelser i andre lovene.

I tillegg til det som hittil er sagt kommer to meget vesentlige elementer, som kommer frem når en lov ses i på bakgrunn av tiden – *versjoner* og *revisjoner*. Med versjoner forstår jeg de forskjellige tilstandene som enkeltelementene, f.eks. setning, har i løpet av et gitt tidsintervall. Men en revisjon forstår jeg tilstanden til et sammensatt element, f.eks. en lov, på et gitt tidspunkt. En ny revisjon vil bestå av ett eller flere nye versjoner. Når en lov endres, vil det komme nye versjoner av de enkelte paragrafene i loven, samt en ny revisjon av selve loven. Hvis perspektivet utvides noe, slik at man ser på alle lovene under ett, vil det ved en endring i en eller flere lover komme en ny revisjon av Norges lover. En revisjon er som et øyeblikksbilde eller en fullstendig utskrift, og de tykke lovsamlingene som kommer hvert ut jevnlig er således gode eksempler på slike revisjoner.

Versjoner og revisjoner er helt nødvendige for å kunne behandle lovene i et tidsperspektiv. En enkel henvisning til åndsverkslovens § 11 er henvisningen til den seneste versjonen av denne bestemmelsen. Hvis man derimot føyer til et tidskomponent, blir det en henvisning til en gitt versjon⁶ av bestemmelsen.

Det er ikke umulig at noen lover vil inneholde andre elementer i tillegg til de som er nevnt ovenfor. Jeg har ikke hatt som mål å gi en fullstendig beskrivelse av oppbygningen og komponentene i samtlige lovene. Når det er sagt, vil jeg gjerne nevne at såvel Kong Christian Den Femtis Norske Lov fra 1687 som Kongeriget Norges Grundlov fra 1814 lar seg beskrive ved den angitte strukturen.

⁶Man kan snakke både om versjon og revisjon av bestemmelsen, alt avhengig av hvilket perspektiv som velges for vurderingen

5 De enkelte elementene i modellen

Jeg nevnte allerede at jeg ønsker å prøve å angi noen hovedtrekk i en datamodell som kan være egnet til å representere lovenes på en slik måte at deres interne struktur blir bevart, samtidig som at det skal være mulig å vise til de enkelte versjonene av bestemmelsene i loven og forskjellige revisjoner av loven. Sett i større sammenheng vil en slik datamodell gjøre det mulig å vise et øyeblikksbilde av hvordan *alle* lover så ut på et av brukeren angitt tidspunkt, slik at brukeren vil være sikker på at både bestemmelsenes ordlyd og lovenes struktur, herunder henvisninger innad i loven og ut til bestemmelser i andre lovene er korrekte.

Analysen ovenfor avdekket to grunnleggende forskjellige typer elementer som en lov vil typisk bestå av: *beholdere*, som sørger for at loven får ønsket struktur, og *teksten*. Beholderen kan inneholde såvel andre beholdere som teksten. Det var beholderen *paragraf*, som inneholdt teksten og det egner seg etter min mening godt å forutsette at all lovtekst i datamodellen skal befinne seg i paragrafer, som vil være en av flere typer *tekstbeholdere*.

En *lov* er en vanlig beholder, men fortjener uten tvil sitt eget modellobjekt. Når det gjelder andre beholdere, som *del*, *kapittel* og *kapitteldel*, tror jeg ikke det er hensiktsmessig å definere egne modellobjekter for disse, iallfall ikke fra begynnelsen av. Det kan imidlertid tenkes at med tiden vil man se et behov for å definere egne modellobjekter for samtlige beholder typer. I prinsippet er det ikke noe som skiller beholdere fra hverandre, bortsett fra navn og den hierarkiske dybden, og man kan gjerne bruke dybde e.l. og navn som attributter til det generelle modellobjektet *beholder*. Alternativt kan man definere mer spesielle modellobjekter hvor navn og dybde er en del av selve modellobjektet.

Paragraf får en egen modellobjekt av typen *tekstbeholder*, jf. ovenfor. Analysen viste imidlertid at det finnes andre tekstbeholdere, som det kan være hensiktsmessig å ta med i modellen. Teksten i paragrafene er delt opp i ledd og til tider også i punkter. Både ledd og punkter er også beholdere, i det begge består av ett eller flere punktum. Jeg har sans for å definere egne modellobjekter med egne egenskaper og attributter for disse tekstbeholdere, men dette utelukkende av hensyn til visuell presentasjon av lovteksten til sluttbrukere.

Hvor mye skal teksten splittes opp? Skal setning eller ord være det minste, elementære og udelelige objektet for teksten? Jeg vil foreslå at man bruker *setning* som grunnobjektet for all tekst i datamodellen. Enkeltordene bærer i seg selv for lite informasjon, til at det skal være noen grunn til å la de få et eget modellobjekt. Setninger, derimot, spiler en selvstendig rolle både som meningsenheter og som komponenter i forskjellige tekstbeholdere. Når det er sagt, er også ord viktige, i det det

er et eller flere ord som danner utgangspunktet for en henvisning, så ordene må også kunne representeres på et eller annet vis. Jeg behandler spørsmålet nærmere i min drøftelse i tilknytning til henvisninger.

Dette gir oss følgende modellobjekter så langt⁷:

- Beholder (kan inneholde andre beholdere og tekstbeholdere)
- Lov (arver egenskapene til *beholder*)
- Tekstbeholder (arver egenskapene til *beholder*, kan inneholde tekstbeholdere og tekst)
- Paragraf (arver egenskapene til *testbeholder*)
- Ledd (arver egenskapene til *tekstbeholder*)
- Punkt (arver egenskapene til *tekstbeholder*)
- Tekst (inneholder teksten)
- Setning (arver egenskapene til *tekst*)

Dette er imidlertid bare bestanddelene, som må knyttes sammen for at loven ikke skal se ut som en usystematisert ordbok. Et viktig spørsmål her er om koblingene mellom de forskjellige modellobjektene skal være egne modellobjekter eller om de skal representeres som attributter på de respektive modellobjektene. Begge tilnærmingene vil kunne brukes, men å analysere hvilken av disse som vil være den mest rasjonelle og effektive faller utenfor rammene til dette innlegget og vil bli gjort i et særskilt arbeid senere.

En type kobling må imidlertid ha et eget modellobjekt – *henvisninger*. Til forskjell fra koblingene omtalt i avsnittet over, som kun skal sørge for at de forskjellige elementene settes sammen i ønsket struktur, er henvisningene egne elementer i selve loven og står på egne ben. Henvisninger har normalt utgangspunkt i et eller flere ord i en bestemmelse og viser til en lov, del, kapittel, paragraf, ledd, punktum osv. Ovenfor argumenterte jeg for å la setningen være det minste modellobjektet for tekst. Hvordan skal man da kunne identifisere det ordet eller de ordene som henvisningen skal være utgangspunktet for henvisningen? Dette er ikke et særlig stort problem. Det er f.eks. mulig å si at henvisningen tar utgangspunkt i ordet med et gitt nummer i den angitte setningen. En annen mulighet er å merke ordet på en spesiell måte som en henvisning i selve teksten. Begge metodene lar seg anvende dersom henvisningens utgangspunkt består av flere ord.

⁷Tilhengere av den objekt-orienterte programmeringsparadigmen får unnskyldt mitt bruk av ordet "objekt" hvor det egentlig er snakk om *klasser* i objekt-orientert forstand

Det kan knytte seg en utfordring til hvordan henvisninger til intervaller skal håndteres⁸. Annet punktum i tvangsfullbyrdelseslovens § 2-3 lyder som følger: “*Domstoloven § 90 første ledd første punktum og annet ledd og §§ 101–105a gjelder tilsvarende.*” I bestemmelsen er det to enkelthenvisninger til § 90 første ledd første punktum og til § 90 annet ledd og en envisning til et intervall – §§ 101–105a. I Lovdata er det satt inn en hyperlenke i teksten som binder ordene “Domstoloven § 90” med bestemmelsen i domstolslovens § 90, mens intervallhenvisningen §§ 101–105a ikke ble gjenstand for etablering av en eller flere hyperlenker. Dette er forståelig, fordi intervallet “101–105a” egentlig består av enkeltelementene “101”, “102”, “103”, “104”, “105” og “105a”, slik at man i tilfelle ville måttet etablere hyperlenker til hver av disse paragrafene eller etablert et eget dokument eller dokumentudrag hvor kun disse bestemmelsene var tatt med. Med min foreslåtte datamodell lar problemet seg løse på en svært enkel måte: det etableres en egen beholder i domstolsloven med §§ 101–105a og det etableres en henvisning til denne beholderen.

Det er tre viktige tidspunkter i en lovs historie – tidspunktet for kunngjøring, tidspunktet for ikrafttredelse og tidspunktet for oppheving. Det samme kan sies om de øvrige beholdere, selv om man da heller vil snakke om tidspunktene for når en versjon er *gyldig fra*, *i kraft fra* og *gyldig til*. Ved fastsettelse av disse, må hensyn tas til de formkrav som gjelder i lovgivningsprosessen.

Modellobjektene får dermed tre grunnleggende tidsrelaterede egenskaper:

- Gyldig fra – tidspunktet for når et element tidligst kan refereres, f.eks. fordi det er blitt vedtatt eller kunngjort
- I kraft fra (ikke relevant for henvisninger) – tidspunktet for når et element tidligst kan påberopes som et formelt grunnlag for en rettsregel
- Gyldig til – tidspunkt for når et element senest kan refereres, f.eks. det er blitt opphevet eller fjernet

Disse attributtene vil gjøre det mulig å referere til modellobjektene i et tidsperspektiv, f.eks. vil brukeren kunne instruere systemet om å vise alle lovene som var i kraft per 1. januar 1970 og deretter utføre evt. søk i de revisjoner som gjaldt da ved følgende forespørsel⁹: “*vis elementer med gyldig fra-dato tom. 1. januar 1970 og i kraft fra-dato tom*”

⁸Her behandles kun henvisninger til intervaller mellom lovene som bruker modellen; tredjeparter som ønsker å henvise til vilkårlige intervaller i lovene kan i og for seg også gjøre det, men jeg går ikke nå inn på hvordan dette i praksis skal kunne realiseres

⁹Forespørselen er angitt i pseudospørrespråk

1. januar 1970 og gyldig til-dato fom. 1. januar 1970.” Forespørseren vil ikke bare søke i lover, men også paragrafer, ledd, punktum og henvisninger!

6 Avslutning

Dette var en nokså overfladisk analyse av behov og angivelse av noen ideer i en mulig løsning foranlediget av disse behovene som forhåpentligvis kan inspirere til videre arbeid med disse spørsmål.

Forskrifter deler lovenes levende natur og den modellen som ble drøftet i dette innlegget kan, evt. med noen mindre justeringer og modifikasjoner, anvendes på forskriftene.

Jeg skal berolige den skeptiske og bekymrede leseren – det skal ikke være nødvendig for mennesker å bruke lange referanser når en henvisning skal legges inn. Jeg ser for meg at det lages et system som ved etablering av en henvisning legger bruker den gjeldende datoen i tidskomponentet til henvisningen. Et annet alternativ er at det i tilknytning til dokumentet angis hvilket tidskomponent henvisninger til lovebestemmelser skal bruke, med mindre det i tilknytning til den enkelte henvisningen er angitt et annet tidskomponent. Uansett hvilken tilnærming som velges, skal ikke brukeren trenge å vite hvordan den underliggende modellen er bygget opp, så lenge den fungerer etter sin hensikt og bidrar til å øke tilgjengelighet og brukervennlighet av rettskildene og å gi et bedre oversikten over disse.

Det er ikke lenge igjen til jul og jeg kunne ønske meg en JUS-IDE¹⁰ til jul. En JUS-IDE for å kunne arbeide med og skrive juridiske tekster i den. Jurister er også utviklere og har behov for verktøy som kan gjøre deres arbeid og hverdag lettere. Juristenes utviklingspråk er norsk og standardbiblioteket er stort og omfattende og består av alle de tilgjengelige og relevante rettskildene som bare lar seg skrive ned. Denne JUS-IDE-en skal på en rask og enkel måte kunne hente opp de rettskildene jeg etterspør, den skal kunne la meg se historikken til de dokumentene som ligger inne og automatisk sette inn fullstendige og entydige referanser når jeg ber om det.

Jeg har ikke noen stor tro på at en slik JUS-IDE kommer med det samme. Men jeg tror heller ikke at et slik utviklingsmiljø for jurister ikke vil komme. Snarere tvert imot. I min avhandling¹¹ nevnte jeg *Lovdata 3000*¹². JUS-IDE-en kan være en del av *Lovdata 3000*.

¹⁰IDE står for *Integrated Development Environment*

¹¹Avhandlingen kan lastes ned fra <http://km.krot.org/2006/iri-oppg.pdf> eller leses i *CompLex* nr. 6/2006

¹²Et tenkt fremtidig rettslig informasjonssystem