

Semantisk Minside

Semantisk web og rettighetsinformasjon

Av

Kjetil Helberg og Steinar Skagemo

Avdeling for forvaltningsinformatikk
Det juridiske fakultet

UNIVERSITETET I OSLO

1. mai 2007

Innholdsfortegnelse

1	Innledning.....	1
1.1	Nærmere om problemstillingen.....	2
1.1.1	Offentlig informasjon.....	2
1.1.2	Offentlig informasjon på nye måter?	3
1.2	Oversikt over fremstillingens struktur.....	4
2	Metode, teori og scenario	5
2.1	Systemutvikling som forskningsmetode	5
2.1.1	Utfordringer ved vårt metodevalg.....	6
2.1.2	Nærmere om utviklingsarbeidets faser.....	7
2.1.2.1	Første fase: Behovsanalysen	7
2.1.2.2	Andre fase: Funksjonell kravspesifikasjon	7
2.1.2.3	Tredje fase: Løsningsforslagene og evalueringen.....	8
2.2	Kort anvisning over oppgavens teoribruk	8
2.3	Oversikt over oppgavens øvrige metodebruk.....	9
2.3.1	Systemutviklingsmetode	9
2.3.2	Analyse av LivsIT og Minside	10
2.3.3	Juridisk metode	10
2.3.4	Transformasjon av rettsregler.....	11
2.4	Mer om Peder og Marte – detaljering av scenario	13
2.4.1	Stønader for nybakte foreldre.....	14
2.4.2	Hva Peder og Marte ikke vet.....	14
2.4.2.1	Lånekassens ordning for rentefritak.....	15
2.4.2.2	Husbandens bostøtteordning	16
2.4.3	Konklusjon	18
3	Hva er Semantic Web?.....	19
3.1	Fra maskinlesbar HTML til maskinforståelige ”smarte data”	20
3.1.1	HTML – maskinlesbart, men helt uforståelig	21
3.1.2	Nærmere om metadata og opplysningstyper.....	22
3.1.3	XML – for høye forventninger, men god på bunnen	23
3.1.4	Smarte data: Felles vokabular på felles format	24
3.2	RDF: En datamodell for beskrivelse av data.....	26
3.2.1	Den semiotiske trekanten	26
3.2.2	RDF-tripler: Objekt, predikat, subjekt	27
3.2.3	RDF og opplysningstyper vs opplysninger	29
3.2.4	En web av opplysningstyper – hva så?.....	31
3.2.5	Semantic Web-basert ”kunnskap” for veiledning av innbyggerne?.....	32
3.3	Nærmere om de forskjellige standardene for Semantic Web (Semantic Web-stacken).....	33
3.3.1	URI, Unicode og XML.....	34
3.3.2	Data interchange: RDF.....	35
3.3.3	RDF-S.....	36
3.3.4	OWL.....	37
3.3.5	SPARQL.....	38
3.3.6	”Rules: RIF”	39
3.3.6.1	CWM.....	39
3.3.7	Unifying logic og proof.....	40
3.3.8	”Crypto” og ”trust”.....	41
3.3.9	User Interface & applications.....	41

3.4	Oppsummering	41
4	Behovsanalyse del 1 - offentlig informasjon.....	42
4.1	Tilgang til rettighetsinformasjon – et spørsmål om rettssikkerhet?	42
4.1.1	Veiledningsplikts rekkevidde	43
4.2	Informasjonspolitik for statsforvaltningen	46
4.3	Den statlige informasjonspolitikken (SIP)	46
4.3.1	De tre målene	47
4.3.2	De fem kommunikasjonsprinsippene	48
4.3.2.1	Kommunikasjonsprinsippet.....	48
4.3.2.2	Prinsippet om aktiv informasjon	49
4.3.2.3	Helhetsprinsippet.....	49
4.3.2.4	Linjeprinsippet	50
4.3.2.5	Prinsippet om informasjon som lederansvar	50
4.3.3	Oppsummering av behovene	51
5	Behovsanalyse del 2 – LivsIT og Minside – to eksisterende offentlige informasjonstjenester	52
5.1	LivsIT	52
5.1.1	Idé og formål	52
5.1.2	Funksjonalitet	52
5.1.3	LivsIT og scenarioet.....	53
5.1.4	Gjennomgang av LivsIT i lys av behovene fra SIP	57
5.1.4.1	Kommunikasjonsprinsippet.....	57
5.1.4.2	Prinsippet om aktiv informasjon	58
5.1.4.3	Helhetsprinsippet.....	59
5.1.4.4	Linjeprinsippet, og 5. Lederansvar.....	59
5.2	Minside.....	59
5.2.1	Idé/formål	59
5.2.2	Funksjonalitet	60
5.2.2.1	Registertjenester	60
5.2.2.2	Transaksjonstjenester	66
5.2.2.3	Kommunale tjenester.....	68
5.2.2.4	Meldingsboks	69
5.2.2.5	Kalender	70
5.2.3	Minside og scenarioet.....	70
5.2.4	Minside og SIP	72
5.2.4.1	Kommunikasjonsprinsippet.....	72
5.2.4.2	Prinsippet aktiv informasjon	72
5.2.4.3	Helhetsprinsippet.....	73
5.2.4.4	Linjeprinsippet, og Informasjon som lederansvar.....	74
5.3	Oppsummering av behovene for en Semantisk Minside.....	75
6	Funksjonell kravspesifikasjon	77
6.1	Forholdet til de øvrige behovene i SIP og andre krav.....	77
6.2	Utledning av krav for hvert behov	77
6.2.1	B-2. Det bør være mulig å tilpasse tjenesten til brukerens konkrete situasjon	78
6.2.2	B-5. Det bør være mulig å senke kravet til egeninnsats for prioriterte grupper	81
6.2.2.1	Konklusjon	81
6.2.3	B-6. Det bør være mulig å få tilgang til ulike målgruppers situasjon og behov gjennom tjenesten.....	81

6.2.4	B-7. Det bør være mulig å bruke tjenesten i samarbeide med private organisasjoner og interessegrupper	82
6.2.5	B-11. Det bør være mulig å dekke det faktiske informasjonsbehovet på en måte som er forståelig for målgruppen	83
6.2.6	B-14. Det bør være mulig å sette sammen tjenester fra flere virksomheter til en helhet	83
6.3	Oppsummering av kravene.....	84
7	Løsningsforslag og vurdering.....	85
7.1	Innledning.....	85
7.2	Overordnet arkitektur	86
7.2.1	Use-case-beskrivelse av S.M.S.	86
7.2.1.1	Brukeren	87
7.2.1.2	S.M.S.	87
7.2.1.3	Den felles innloggingsløsningen	88
7.2.1.4	Samtykke	88
7.2.1.5	Personopplysninger	88
7.2.1.6	Tjenesteleverandørene.....	89
7.2.1.7	Aktiv sesjon.....	89
7.2.1.8	Fødselsnummer	89
7.2.1.9	Transformert regelverk.....	89
7.2.1.10	S.M.S.' eget lager for transformert regelverk	89
7.2.1.11	Tredjeparter/interesserorganisasjoner	89
7.2.1.12	Transformert regelverk på nettstedet til tjenesteleverandører og tredjeparter/interesseorganisasjoner	90
7.2.1.13	Arbeidsminne	90
7.2.1.14	Brukerprofil.....	90
7.2.1.15	Resonnerer.....	91
7.2.1.16	S.M.S.' slutningsmekanisme	91
7.2.1.17	Tilpasset presentasjon av informasjon	91
7.2.1.18	Tjenester markert som relevante/ikke-relevante	91
7.2.1.19	Brukeren endrer/oppdaterer opplysninger.....	92
7.2.1.20	S.M.S. tilpasser informasjon på nytt	92
7.2.1.21	S.M.S. markerer hva som er endret.....	92
7.2.1.22	Brukeren endrer/utvider regelverksgrunnlaget.....	92
7.2.1.23	Utføre en tjeneste	93
7.2.1.24	Logg ut	93
7.2.1.25	Arbeidsminnet tømmes	93
7.2.2	Utveksling av personopplysninger og regelverk mellom S.M.S. og andre aktører	93
7.3	S.M.S.-prototyp basert på CWM – oversikt.....	94
7.4	Eksempel – automatisert veiledning basert på Lånekassens regler for rentefritak ..	95
7.4.1	Sortering av regelverket	95
7.4.2	Etablering av rettslig informasjonsmodell	97
7.4.3	Etablering av rettslig prosessmodell	99
7.4.4	Eksempel på kjøring	100
7.4.4.1	Opplysninger om brukeren 'Peder Ås' i S.M.S.	101
7.4.4.2	Regel M1: Låntaker som dokumenterer fødsel	101
7.4.4.3	Regel M2: Den av foreldrene som har omsorgen for barn.....	103
7.4.4.4	Felles regel PK1-FM1: Inntektsgrense som økes pr barn	104
7.4.5	Konklusjon	105

7.5	Eksempel – automatisert veiledning basert på Husbankens regler for bostøtte.....	105
7.5.1	Konklusjon	108
7.6	Vurdering – automatisert rettslig veiledning.....	108
7.6.1	Konklusjon	109
7.7	Nærmere om S.M.S. kontrollstruktur.....	110
7.7.1	Konklusjon	114
7.8	Kravtabell og konklusjon	115
8	Oppsummering	117
8.1	Kan Semantic Web gi bedre rettighetsinformasjon?.....	117
8.1.1	Schartums transformasjonsmetode som grunnlag for veiledning	118
8.2	Våre erfaringer med systemutvikling som forskningsmetode.....	118
9	Litteraturliste	120
10	Vedlegg 1: Oppsummering av økonomien til Peder og Marte.....	125
11	Vedlegg 2: Oppsummering av økonomien dersom Marte og Peder hadde visst	126
12	Vedlegg 3: Beskrivelse av eksempler i kapittel 3	127
12.1	Eksempel – kapittel 3.2.2 – litt om Peder	127
12.1.1	N3-notasjon	127
12.1.2	Filnavn.....	127
12.2	Eksempel – kapittel 3.2.3 – opplysningstyper vs opplysninger	127
12.2.1	Filnavn.....	127
12.3	Eksempel – kapittel 3.3.5 – spørring med SPARQL.....	127
12.3.1	Filnavn.....	127
12.3.2	Kjøring	128
12.3.3	Resultat.....	128
12.4	Eksempel – kapittel 3.3.6.1 – regler og slutningsmekanismer.....	128
12.4.1	Filnavn.....	128
12.4.2	CWM-kommando.....	128
12.4.3	Resultat av kjøring i CWM	128
12.4.3.1	Kommentar	129
13	Vedlegg 4: Kildekode.....	130
13.1	Kapittel 3 – Eksempeldata.....	130
13.1.1	barepeder.n3	130
13.1.2	Eksempel2-opplysningstyper-bolig.n3.....	130
13.1.3	Eksempel4-sparql-query-peder.sparql.....	131
13.1.4	Eksempel5-Peder-Husstandsmedlem.n3	131
13.2	Kapittel 8 -- Registeropplysninger-filer	132
13.2.1	Registeropplysninger-Peder-skdf.data	132
13.2.2	Registeropplysninger-Peder-skds.data	132
13.2.3	Registeropplysninger-Peder-lanekassen.data	133
13.2.4	Registeropplysninger-Peder-egne.data.....	133
13.3	Regel-filer.....	134
13.3.1	Rentefritaksregel-M1.regel	134
13.3.2	Rentefritaksregel-M2.regel	134
13.3.3	Rentefritaksregel-PK1-FM1.regel.....	135
13.4	Rapportregler.....	136
13.4.1	Rapportregel-rentefritak.rapp.....	136
13.4.2	Rapportregler-transformasjonsmetoden.rapp.....	136
13.5	RDF-data på XML-format	137
13.5.1	Transformasjonsmetoden	137
13.6	Egne erfaringer	146

13.7	Eksempler.....	146
13.8	Andre eksempler	147
13.9	Skjermbilder fra kjøring av CWM	147
13.10	Noen referanser	150

Figurliste

Figur 1: Schartums modell som illustrerer rettslig informasjonsmodell, prosessmodell og resultater	23
Figur 2 -- Eksempel på RDF-data: To RDF-tripler om Peder Ås (navn og barn).....	27
Figur 3 -- RDF-data som viser opplysningstyper	30
Figur 4 Sirkelen viser RDF-datamodellens anvendelsesområde i Schartums modell	31
Figur 5 -- Semantic Web Architecture -- Tim Berners-Lee 2006 -- http://www.w3.org/2006/Talks/0718-aaai-tbl/Overview.html#(14)	34
Figur 6 Sirkelen illustrer at SPARQL, regler og logikk understøtter prosessmodellen i Schartums modell.....	40
Figur 7 - fra Ski kommunes hjemmeside [6/2-2007]	54
Figur 8 – Skjerm bilde etter å ha klikket på ”Foreldre og barn” under overskriften ”Familie” [6/2-2007].....	55
Figur 9 -- Eksempel på request XML fra Minside til tjenesteleverandør	62
Figur 10 Eksempel på XML response fra tjenesteleverandør (NAV).....	63
Figur 11 Eksempel på utdrag fra XSL-stilark	64
Figur 12 Eksempel på kode transformert fra XML til HTML	65
Figur 13 Eksempel på visning av registerdata i nettleser.....	65
Figur 14 Eksempel på visning av registertjeneste i Minside.....	66
Figur 15 - Overordnet arkitektur -- S.M.S. og aktører	86
Figur 16 -- Detaljering av utveksling av personopplysninger og/eller regelverk mellom S.M.S. og tjenesteleverandører/tredjeparter	94
Figur 17 -- Flytskjema som illustrerer S.M.S. kontrollstruktur ved håndtering av regelverk	111
Figur 18 -- Utkast til ER-modell for begrepene i Schartums transformasjonsmetode.....	112
Figur 19 -- Utkast til relasjoner og attributter for entiteten "regel".....	113

1 Innledning

Peder Ås og Marte Kirkerud har nettopp blitt foreldre. Sykehuset har sendt varsel til folkeregisteret om den nye innbyggeren, og om et par uker vil Marte få et brev fra NAV¹ som forteller at de har fattet vedtak om tildeling av barnetrygd, kr 970,- per måned fra og med neste måned. Fordi Marte ikke har jobbet minst seks av de siste ti månedene før fødselen, har de ikke krav på "foreldrepenger ved fødsel".² Men Peder og Marte har undersøkt på nettsidene til NAV og kommunen de bor i, samt på Minside, og funnet ut at de kan få engangsstønad, som utgjør kr 33.358,-.

Det de ikke vet er at de i den samme perioden også har rett til bostøtte, kr 2.057,- per måned, samt rentefritak på studielånet – en ordning som gir kostnadsfri utsettelse av avdragene. Til sammen ville dette gitt dem en økning i disponibel inntekt på nesten 25 %, nærmere bestemt kr 24.324,-, i det første halvåret etter fødselen.³

Puster velferdsstaten Norge lettet ut over nok et tilfelle av innbyggere som ikke vet om, og dermed ikke får gjort krav på rettigheter de har – eller gremmes den over at viktig informasjon ikke har nådd frem til målgruppen? Ifølge den statlige informasjonspolitikken er et av målene at "befolkningen skal [...] få informasjon om sine rettigheter, plikter og muligheter". Rettssikkerhetsgarantien som er nedfelt i forvaltningslovens § 11 om forvaltningens veiledningsplikt, har som formål "å gi parter og andre interesserte adgang til å vareta sitt tarv i bestemte saker på best mulig måte". Hvordan står dette i forhold til det dokumenterte underforbruket av velferdsordninger?⁴

Scenarioet over, som er detaljert i kapittel 2.4 "Mer om Peder og Marte – detaljering av scenario", viser at informasjon som er viktig for Peder og Marte ikke er enkelt tilgjengelig, selv om de leter der man kan forvente å finne den. Dette til tross for at offentlig sektor har som uttalt målsetning at innbyggerne skal få informasjon om sine rettigheter.⁵ Dette danner utgangspunkt for den overordnede problemstillingen i oppgaven vår; folk får ikke den informasjonen de trenger på en enkel måte.

Scenarioet illustrerer hvordan offentlig informasjon i 2007 er tilgjengelig via Internett.⁷ Siden de første offentlige nettstedene ble etablert på midten av 90-tallet, har det blitt lagt ned mye arbeid i å gjøre nettstedene mer brukervennlige. Dette er imidlertid i hovedsak blitt gjort ved hjelp av de samme standardene for nettsider som ble utviklet på begynnelsen av 90-tallet,

¹ Arbeids- og velferdsdirektoratet <http://www.nav.no/> [30/4-2007]

² NAVs informasjonsside om foreldrepenger ved fødsel, se: <http://www.nav.no/805330375.cms> [30/4-2007]

³ Se kapittel 2.4 for detaljer om Peder og Martes situasjon.

⁴ Se for eksempel St.meld nr 25 (1999-2000) Om fri rettshjelp, se:

<http://www.regjeringen.no/nb/dep/jd/dok/regpubl/stmeld/19992000/Stmeld-nr-25-1999-2000-.html?id=331821> [30/4-2007]

⁵ "Informasjonspolitik for statsforvaltningen", 2. målsetning: "Befolkning [...] skal få informasjon om sine rettigheter, plikter og muligheter". Se:

<http://www.regjeringen.no/en/dep/fad/Documents/rapporter/planer/Rapporter/2001/Informasjonspolitik-for-statsforvaltningen.html?id=438513> [29/4-2007]

⁶ I forordet til "Bostøtte 2007 - Håndbok i regelverk", februar 2007 står det "Bostøtte er en rettighetsordning og det er viktig at kunnskapen om ordningen når fram til målgruppene. Vi håper at den enkelte kommune og bydel aktivt vil bidra til å gjøre ordningen kjent for den det måtte angå."

⁷ I denne fremstillingen vil vi benytte Internett og nettet synonymt med World Wide Web, selv om sistnevnte bare er en av flere mulige måter å bruke Internett på.

nemlig HTML og CSS.⁸ Samtidig har de som utviklet disse standardene, jobbet videre. En retning i denne videreutviklingen er *Semantic Web*.

Semantic web omtales nå stadig oftere som “web of data”, og bygger på et sett nye standarder fra World Wide Web Consortium (W3C).⁹ Som all ny teknologi følger det en flom av løfter med Semantic Web, og blant disse finner vi løftene om at det skal bli lettere å finne frem til relevant informasjon. Kanskje kan den nye teknologien også hjelpe Peder og Marte? Dette fører oss over til den konkrete problemstillingen i denne oppgaven:

Problemstilling: *Kan Semantic Web bidra til å gjøre rettighetsinformasjon tilgjengelig på en bedre måte?*

1.1 Nærmere om problemstillingen

1.1.1 Offentlig informasjon

Offentlig sektor har et stort behov for å informere om seg selv og sine tjenester. Det holder ikke å ha gode velferdsordninger hvis innbyggerne ikke vet at disse finnes. Det å informere innbyggerne om rettigheter og plikter er en omfattende prosess og offentlig sektor har gjennomført mye arbeid på dette området. Internett åpnet en ny mulighet for å gjøre offentlig informasjon tilgjengelig. Dette har resultert i at de aller fleste offentlige virksomheter i dag har egne hjemmesider.

Som en illustrasjon på mangfoldet av nettsteder med offentlig informasjon kan vi nevne at Norge.no¹⁰ høsten 2006 gjennomførte en kvalitetsvurdering av 691 forskjellige offentlige nettsteder.¹¹ Dette er i hovedsak etaters og kommuners ”hjemmesider”. Flere av etatene opererer i tillegg til dette med egne spesialiserte nettsteder, men det bare er etatens hjemmeside som er med i vurderingen til Norge.no.¹² Andre eksempler på offentlige nettsteder som ikke regnes med er nettsidene til sykehus og skoler.¹³ Antallet offentlige nettsteder er derfor i realiteten mye høyere.

Problemet offentlig sektor står overfor er av samme art som det som gjelder for de øvrige aktørene som ønsker å gjøre informasjon kjent gjennom Internett. Det er *for mye* informasjon tilgjengelig. Det offentlige informerer på et generelt grunnlag, mens det man som innbygger ofte ønsker er å få informasjonen som gjelder ens egen situasjon.

For å sørge for at innbyggerne skal få den informasjonen de trenger har offentlig sektor lansert ulike løsninger for å hjelpe innbyggeren. Et eksempel er det såkalte LivsIT-prosjektet¹⁴ hvor utgangspunktet var at innbyggeren skulle finne frem til relevant informasjon basert på sin ”livssituasjon”. Man får ikke opp all informasjon som er tilgjengelig om et område, men informasjonen er ”foredlet” til å passe bedre med din livssituasjon.

⁸ HyperText Markup Language (HTML) og Cascading Style Sheets (CSS) er standardisert gjennom World Wide Web Consortium (W3C), som også jobber med standardene for Semantic Web. Se oversikt over W3Cs aktiviteter på deres hjemmeside <http://www.w3.org/> [30/4-2007]

⁹ Standarder fra W3C kalles ”Recommendations”

¹⁰ <http://www.norge.no> [30/4-2007]

¹¹ Fra artikkelen ”Ikke helt på nett”, Aftenposten 11. april 2007, side 17.

¹² Et eksempel på dette er Post- og teletilsynet, som i tillegg til egen hjemmeside, www.npt.no, har portalene www.telepriser.no, www.nettvett.no, www.frekvensportalen.no og www.bredbandsporten.no.

¹³ For sykehus vurderes bare helseregionenes hjemmesider.

¹⁴ Mer om LivsIT kommer i kapittel 5.1 LivsIT.

Et nyere eksempel er Minside. På samme måte som LivsIT skal Minside hjelpe innbyggerne med å finne frem til offentlige elektroniske tjenester uten å måtte vite hvilken etat som er ansvarlig for tjenesten. Det holder å logge inn ett sted, og derfra skal du kunne benytte deg av de ulike tjenestene offentlig sektor tilbyr. Gjenfinningsmekanismen er temaord, som er basert på videreutvikling av "livssituasjonene" i LivsIT. Men i motsetning til LivsIT, der brukeren var anonym, er Minside en tjeneste med tilgang til opplysninger om brukeren.

Det ønskelige for både offentlig sektor og innbyggerne er at innbyggeren skal få den informasjonen han eller hun trenger i en gitt situasjon. Informasjonen skal være relevant for den enkelte. Den statlige informasjonspolitikken hevder at offentlig sektor skal gi innbyggerne informasjon om deres plikter og rettigheter. Samtidig illustrerer scenarioet vårt at den statlige informasjonspolitikken ikke gjennomføres på en så god måte som ønskelig. Peder og Marte går glipp av flere tusen kroner i måneden på grunn av vanskelig tilgjengelig informasjon fra det offentlige. Selv om både LivsIT og Minside er tiltak som er ment å bedre på situasjonen, ser vi at disse tiltakene ikke nødvendigvis er tilstrekkelige.

1.1.2 Offentlig informasjon på nye måter?

Som tidligere nevnt har den statlige informasjonspolitikken som en av sine målsetninger at den skal informere om **dine** rettigheter. Ville det ikke da vært elegant med en nettside som samlet all informasjon om dine rettigheter på ett sted? Slik at du slipper å lete etter informasjonen du trenger, den ligger tilgjengelig for deg når du trenger den.

Venter du barn får du informasjon som er relevant for de som er gravide, etter nedkomsten endrer informasjonen seg til relevant informasjon for de med spedbarn¹⁵. Når det nærmer seg tid for barnehave- eller kontantstøttesøknad kommer relevant informasjon om dette emnet opp. Og kanskje like viktig – du slipper å få all denne informasjonen hvis du ikke har barn eller planer om å få det. Er du 70 år og/eller barnløs vil du ikke få informasjon om barnehavetilbudet i din kommune.

Informasjonssamfunnet og Internett har gjort offentlig informasjon enklere tilgjengelig enn noen gang tidligere, men det fører også med seg noen nye utfordringer. Den største utfordringen er kanskje det å finne frem i all informasjonen? Med en datamaskin koblet til Internett har man tilgang til ufattelige mengder informasjon.¹⁶

Problemet nå er ikke nødvendigvis at informasjon ikke finnes, men at den drukner i all den andre informasjonen som også ligger der. Dette har skapt et krav om at man må lage informasjon slik at den er enkel gjenfinnbar¹⁷. Det gjelder også informasjon fra det offentlige. Mens noen vil mene at informasjonsoverfloden på Internett vil gjøre Internett mer eller mindre ubrukelig som ressurs, mener andre at problemet kan løses på den samme måte som vi løser alle problemer med datamaskiner; nemlig ved å "oppgradere til neste versjon", altså Semantic Web.¹⁸

¹⁵ Dette samsvarer med "folkets ønskeliste" i Teknologirådets rapport "offentlige tjenester på Internett". For rapporten, se: http://www.teknologiradet.no/rapport_endelig_isxfU.pdf.file, for ønskelisten, se: <http://www.teknologiradet.no/FullStory.aspx?m=74&amid=3121> [30/4-2007]

¹⁶ Ifølge David Norheim, på Norstellas seminar om semantisk interoperabilitet 14. mars, indekserte Google 8,5 milliarder nettsider for ett år siden, mens de i år passerer 100 milliarder nettsider.

¹⁷ Informasjonsoverflod og gjenfinnbar blir her brukt som oversettelse av det som i den engelske litteraturen omtales som "information overflow" og "findability".

¹⁸ Semantic Web innebærer vel og merke ikke en ny versjon av World Wide Web, i den forstand at vi erstatter World Wide Web med noe nytt. Semantic Web er en utvidelse av de standardene som ligger til grunn for dagens World Wide Web, og som gir nye muligheter som kan utnyttes i parallell med de vi har idag.

Dette er temaet i denne oppgaven: Er det mulig å gi innbyggerne den informasjonen de trenger, når de trenger det, på en enkel måte? Vi ønsker å se om dette lar seg gjøre ved hjelp av den siste utviklingen innen internett-teknologi, Semantic Web. Vi vil fokusere på den delen av offentlig informasjon som gjelder informasjon om innbyggernes *rettigheter*.

1.2 Oversikt over fremstillingens struktur

I kapittel 2 presenterer vi forskningsmetoden vi har valgt – systemutvikling som forskningsmetode. Deretter gir vi en kort oversikt over teoribruken i oppgaven. Kapitlet inneholder videre en nærmere gjennomgang av de øvrige metodene vi gjør bruk av, før vi til slutt går igjennom det fullstendige scenarioet som beskriver Peder og Martes situasjon i detalj.

Kapittel 3 er en gjennomgang av Semantic Web. Vi ser først på begrensningene i dagens web og hvordan Semantic Web kan være en løsning på dette. Deretter går vi nærmere inn på det som er den viktigste byggestenen i Semantic Web, nemlig datamodellen som betegnes som ”Resource Description Framework” (RDF). Til slutt går vi igjennom den ”stabelen” av teknologier som utgjør Semantic Web.

Kapittel 4 og 5 utgjør til sammen en behovsanalyse. Første del av behovsanalysen (kapittel 4) er å utlede hvilke behov den statlige informasjonspolitikken stiller til en tjeneste som skal gi rettighetsinformasjon. Resultatet – et sett av behov – tar vi med videre til den andre delen av behovsanalysen, kapittel 5. Her går vi igjennom to eksisterende tjenester, LivsIT og Minside, og ser hvilke av behovene som er dekket av disse to tjenestene. De udekkete behovene – gapet – utgjør resultatet av behovsanalysen.

Resultatet av behovsanalysen er grunnlaget for å etablere en funksjonell kravspesifikasjon, som vi gjør i kapittel 6. Denne kravspesifikasjonen lages som en tenkt videreutvikling av Minside.

Med utgangspunkt i kravspesifikasjonen fra kapittel 6, ser vi i kapittel 7 på hvordan Semantic Web-teknologi kan brukes i en løsning som dekker kravene i kravspesifikasjonen. Dette gjør vi i form av en løsningsbeskrivelse for det vi har kalt ”Semantisk Minside”, forkortet S.M.S.¹⁹ Vi foretar samtidig en vurdering av løsningsforslagets viktigste deler.

Kapittel 8 er en oppsummering av oppgaven med de konklusjonene og erfaringene vi kan trekke av arbeidet.

¹⁹ Vi har valgt å skrive forkortelsen med punktum, for å skille den fra forkortelsen for ”Short Message System” som er tekstmeldinger på mobiltelefoner.

2 Metode, teori og scenario

For å besvare problemstillingen har vi valgt en ”informatisk” tilnærming, i den forstand at vi selv ønsker å prøve ut teknologien det er snakk om – Semantic Web – på det aktuelle problemområdet – tilgang til rettsinformasjon. Konkret vil vi gjøre dette ved å designe en løsning som bruker Semantic Web-teknologi, som vi deretter gir vår vurdering av med tanke på om den bidrar til forbedret tilgang til informasjon.

Semantic Web-teknologien er en utvidelse av dagens Internett-teknologi. Det gjør det mulig å designe en løsning som baserer seg på dagens løsninger for offentlig informasjon på Internett. Vi har valgt å ta utgangspunkt i Minside, og løsningsforslaget kaller vi derfor ”Semantisk Minside”, heretter forkortet til S.M.S.

Å designe og utvikle et informasjonssystem er et arbeid det finnes etablerte metodologier for, *systemutviklingsmetoder*. Felles for de fleste av metodene er at de deler arbeidet inn i faser. Vi vil ta utgangspunkt i en enkel fossefallsmodell, men vi vil bare gjennomføre de første fasene (behovsanalyse, kravspesifikasjon, design), samt kombinere det med prototyping for å kunne gi en vurdering av resultatet, uten at vi trenger å implementere og produksjonssette løsningsforslaget. I kapittel 2.1 ”Systemutvikling som forskningsmetode” nedenfor, ser vi nærmere på hvordan systemutvikling kan brukes som forskningsmetode. Senere, i kapittel 2.3.1, ser vi nærmere på hvilke konkrete valg vi har gjort i systemutviklingsarbeidet.

2.1 Systemutvikling som forskningsmetode

Vårt undersøkelsesopplegg er spesielt i den forstand at det ikke har som mål å få kunnskap *om et eksisterende fenomen*, der metodevalget styres av hva som er en hensiktsmessig måte å belyse fenomenet på. Eksempelvis vil det i samfunnsvitenskaplige undersøkelser være en vurdering om man bør belyse et fenomen gjennom kvalitative eller kvantitative undersøkelser. Valget vil også ha en betydning for i hvilken grad det er mulig å generalisere resultatene. Normalt vil det f.eks. være lettere å generalisere resultater basert på kvantitative data fra mange undersøkelsesobjekter, sammenlignet med kvalitative data fra færre objekter (typisk casestudier).

I vårt tilfelle derimot, lager vi noe nytt. I stedet for utvidet kunnskap om et fenomen, er målet for vårt arbeid å foreslå en løsning – i form av et informasjonssystem – på noe vi oppfatter som et problem. Vi må *designe* denne løsningen. Artikkelen ”Design Research in Information Systems”²⁰ argumenterer for hvorfor design av informasjonssystem også er en aktuell forskningsmetode. Poenget er at design og utvikling av informasjonssystemer bør anerkjennes som forskningsmetode fordi det *produserer verdifulle resultater*. Blant annet sammenlignes design av informasjonssystemer med forskning i form av konstruksjon og testing (i full skala) av bl.a. fly; utviklingen innen luftfart ville ikke vært mulig uten at noen faktisk bygget (antatt) flyvedyktige konstruksjoner, som prototyper og senere pilotversjoner, for så å prøve dem.

Design av informasjonssystemer som forskningsmetode er en tilnæringsmåte som normalt vil involvere bruk av *flere* konkrete, praktiske metoder. For eksempel trengs det metoder for å sette seg inn i problemområdet man ønsker å designe en løsning for. Dette kan blant annet gjøres ved hjelp av dokumentstudier eller intervjuundersøkelser. Videre trengs det teknikker for selve designet (f.eks. analyse- og modelleringsmetodikk fra systemutvikling), og det trengs metoder for å validere løsningen som er designet. Mer om de konkrete metodevalgene

²⁰ Se: <http://www.isworld.org/Researchdesign/drisISworld.htm> [30/4-2007]

knyttet til systemutviklingen i denne oppgaven finnes i kapittel 2.3.1. Systemutviklingsmetode.

2.1.1 utfordringer ved vårt metodevalg

En åpenbar svakhet når vi bruker systemutvikling som undersøkelsesmetode, er at vi har alle rollene i arbeidet selv; når vi både definerer behovene, lager løsningen og vurderer den etterpå, er det fare for at vi ender opp med en løsning som kun tilfredsstiller *våre* behov, og på en måte som kun *vi* er fornøyde med. Det gir med andre ord ingen sikkerhet for at vi 1) lager en løsning som dekker *faktiske* behov, og det er heller ingen sikkerhet for at løsningen 2) tilfredsstiller disse behovene på en måte som *andre* mener er god.

For å kompensere for denne svakheten har vi brukt to strategier. For det første baserer vi behovsanalysen vår på den statlige informasjonspolitikken. Informasjonspolitikken er et uttrykk for hvordan det offentlige selv mener utfordringer knyttet til informasjon bør løses. I kapittel 4.1.1 ser vi dessuten hvordan informasjonspolitikken kan knyttes til styringsidealet rettssikkerhet, og på den måten er uttrykk for en helt sentral verdi i vårt samfunn. Ved å basere behovsanalysen på den statlige informasjonspolitikken mener vi at vi øker sannsynligheten for å komme frem til *faktiske* behov. For det andre bruker vi et scenario som teknikk i analysefasen. Og lar scenarioet med Peder og Marte følge oss i arbeidet, spesielt i forbindelse med vurderingen av den løsningen vi designer. Siden scenarioet kun representerer én konkret situasjon, gir den ikke mulighet til å trekke generelle konklusjoner direkte. Men vi mener scenarioet er relevant for en viktig utfordring knyttet til rettighetsinformasjon, og det gir oss grunnlag for å argumentere for at en løsning som ville fungere i Peder og Martes situasjon, også vil fungere for andre i lignende situasjoner²¹.

Et annet tiltak vi gjør, for å unngå at vi kun designer en løsning ”på papiret” som vi så vurderer i forhold til en ”papirsituasjon”, er at vi lager prototyper på sentrale deler av funksjonaliteten i løsningen. På den måten får vi testet hvordan funksjonaliteten i løsningen vår vil fungere i Peder Ås sin situasjon. Her får vi også effekt av at vi har et såpass detaljert scenario – som er laget *før* løsningsforslaget – slik at vi kan teste funksjonaliteten vi prototyper med scenarioet som ”testcase”.²²

Et spesielt forhold ved arbeidet med utviklingen av S.M.S. er at det vil være et system ”med juss i”. Som vi vil se bør S.M.S. kunne gi en form for rettslig basert *veiledning* basert på noe S.M.S. vet om brukeren. Rettssikkerhetsgarantier som er relevante for utvikling av rettslige beslutningssystemer er foreslått i Schartum (2005) gjennom det vi vil referere til som ”Schartums transformasjonsmetode” – hvordan transformere lovregler til programkode på en måte som er i tråd med rettssikkerhetsidealet. Selv om vi i vårt tilfelle ikke skal utvikle et beslutningssystem, vil en form for automatisert veiledning på samme måte som i rettslige beslutningssystemer forutsette at rettsregler transformeres på korrekt måte over i veiledningsalgoritmene. Det er derfor nærliggende å anta at i hvert fall deler av Schartums transformasjonsmetode er anvendelig i vårt tilfelle. I kapittel 2.3.4 ser vi nærmere på Schartums transformasjonsmetode og bruken av den i denne oppgaven.

²¹ Det å bruke scenario er en vanlig teknikk i systemutviklingsarbeid, for eksempel ved å lage use-cases, se for eksempel G.Schneider og J. Winters "Applying use cases" (1998)

²² Dersom testcases lages *etter* at funksjonaliteten er implementert i en prototyp, er det fare for at man (ubevisst) velger et testcase som man vet vil virke.

2.1.2 Nærmere om utviklingsarbeidets faser

For å forsøke å unngå at vi designer en løsning der teknologien vi har valgt blir et mål i seg selv, vil vi gjøre de første fasene av utviklingsarbeidet så teknologiavhengig som mulig. Dette søker vi å oppnå ved først å foreta en *behovsanalyse* – det vil si et forsøk på å identifisere de behovene løsningen skal dekke, uten samtidig å si *hvordan* behovene skal dekket. De identifiserte behovene holder vi deretter opp mot de to eksisterende løsningene LivsIT og Minside, for å sortere vekk de behovene vi mener allerede er dekket. Det *gapet* som gjenstår blir deretter grunnlag for en *funksjonell spesifisering* for videreutvikling av Minside – ettersom det er denne løsningen vi bygger videre på. Først når det er gjort, bringer vi inn den semantiske teknologien og beskriver konkrete løsningsforslag til de funksjonelle kravene vi stiller til SMS. Samtidig som vi beskriver Semantic Web-baserte løsninger på de forskjellige funksjonelle kravene, forsøker vi å evaluere løsningsforslagene både gjennom å sammenholde dem direkte med den statlige informasjonspolitikken, og ved å se på scenariot vårt; hva vil løsningsforslaget innebære for situasjonen til Peder og Marte? Nedenfor følger en gjennomgang av de fasene vi går igjennom.

2.1.2.1 Første fase: Behovsanalysen

Utgangspunktet for behovsanalysen er den statlige informasjonspolitikken. Når vi går igjennom denne er vårt perspektiv knyttet til informasjon om *rettigheter*. Målgruppen for informasjonen er enkeltpersoner på jakt etter informasjon om sine rettigheter, som for eksempel Peder og Marte. Vi bruker dette perspektivet når vi gjennomfører en tekstlig fortolkning av dokumentet som beskriver informasjonspolitikken, og utleder behovene fra dette.

Vi er på jakt etter løsninger på *udekkete* behov. Vi har derfor valgt å sammenholde behovene vi utleder fra informasjonspolitikken med funksjonalitet som allerede finnes i LivsIT og Minside. Begrunnelsen for å velge disse to løsningene er at de etter vår mening gir løfter om å forbedre tilgangen til informasjon om rettigheter.²³ LivsIT samler rettighetsinformasjon knyttet til livssituasjoner. Minside samler på samme måte nettbaserte tjenester fra det offentlige etter den samme temainndelingen som LivsIT. I tillegg har Minside tilgang til opplysninger som er registrert om innbyggeren i flere forskjellige etater. Det er nærliggende å tenke seg at denne kunnskapen om innbyggeren burde kunne brukes til å gi bedre informasjon. Som en del av behovsanalysen inngår derfor også en presentasjon av de to løsningene.

For å fullføre behovsanalysen holder vi altså behovene vi har formulert med utgangspunkt i den statlige informasjonspolitikken opp mot de to løsningene LivsIT og Minside. Her finner vi at mange av behovene allerede er dekket, men vi får også frem et *gap* i form av udekkete behov. Siden vi tar utgangspunkt i Minside, er det bare dette gapet vi trenger å gå videre med i utviklingsarbeidet. Det udekkete behovet utgjør dermed input til neste fase, funksjonell spesifisering.

2.1.2.2 Andre fase: Funksjonell kravspesifisering

I denne fasen forsøker vi å formulere hvert av behovene i ett eller flere konkrete krav til funksjonalitet for S.M.S. Også disse funksjonelle kravene skal fortrinnsvis være så

²³ Ideen bak LivsIT finner vi formulert i NOUen som ligger til grunn for den statlige informasjonspolitikken (Viksveen-utvalget). Der fremgår det at de mener problemet med manglende kjennskap til rettigheter skyldes at "regelverket ikke er organisert i forhold til befolkningens livssituasjon". (NOU 1992:21 s. 121, vår understrekning)

uavhengige som mulig av valg i forhold til konkret teknisk implementasjon. Etterhvert som kravene konkretiseres er det riktignok urimelig å tro at vi ikke er påvirket av den teknologien vi har ”hodet fullt av”, slik at det er sannsynlig at teknologiske valg ”skinner igjennom” i våre kravformuleringer.²⁴

I arbeidet med å formulere de funksjonelle kravene bruker vi scenarioet vårt som et mentalt verktøy for å analysere behovene og komme frem til et funksjonelt krav. I noen tilfeller gjør vi dette eksplisitt ved å sette opp konkrete bruksscenarier (”use case”). Resultatet av arbeidet med å formulere konkrete krav til S.M.S. er en funksjonell kravspesifikasjon, som utgjør input til neste fase, design av løsningsforslagene.

2.1.2.3 Tredje fase: Løsningsforslagene og evalueringen

I denne fasen designer vi en løsning som gjør bruk av Semantic Web for å oppfylle den funksjonelle kravspesifikasjonen. Dette formulerer vi som en løsningsbeskrivelse. Vi har også valgt å vurdere løsningsforslaget fortløpende i denne fasen. For å konkretisere løsningsforslaget prototyper vi deler av funksjonaliteten. Dette er til hjelp ved vurderingen av forslaget ettersom det gir oss mulighet for å se hvordan løsningsforslaget ville fungert for Peder og Marte.

En slik ”case-basert” vurdering gir oss ikke grunnlag for å slutte at resultatet vil være allmenngyldig, men det vil kunne styrke et argument for at det er sannsynlig at dersom S.M.S. forbedrer tilgangen til rettighetsinformasjon for Marte og Peder, så vil også flere ha nytte av en slik løsning.

Prototypingen gjør vi ved å se på de to konkrete regelverkene som er aktuelle for Peder og Marte. Vi transformerer disse reglene til et Semantic Web-regelverksformat²⁵, og bruker et verktøy – CWM – for å anvende disse reglene på opplysninger om Peder og Marte.

2.2 Kort anvisning over oppgavens teoribruk²⁶

I dette kapittelet gir vi en kort oversikt over denne oppgavens teoribruk. Hovedfokus for oppgaven er en praktisk tilnærming – å utvikle en løsning. Oppgaven har derfor ikke en analyse av teori. Til gjengjeld gjør vi bruk av teoretisk materiale i forbindelse med systemutviklingsarbeidet.

For det første går vi gjennom Semantic Web og ser denne i forhold til Schartums transformasjonsmetode for å kunne gi en foreløpig vurdering av om Semantic Web kan understøtte Schartums transformasjonsmetode. Schartums transformasjonsmetode presenterer vi i metodegjennomgangen nedenfor, i kapittel 2.3.4 ”Transformasjon av rettsregler”. Semantic web og den foreløpige vurderingen i forhold til Schartums transformasjonsmetode, presenterer vi i kapittel 3 ”Hva er Semantic Web?”.

I tillegg er behovsanalysen vår basert på en teoretisk gjennomgang av temaet offentlig informasjon, representert ved den statlige informasjonspolitikken. Dessuten ser vi på juridisk

²⁴ Dette blir forsterket av at alle fasene i vårt systemutviklingsarbeid gjøres av de samme to personene. I systemutviklingsmetode er det vanlig at brukerrepresentanter bidrar med behovskartleggingen og den funksjonelle spesifikasjonen.

²⁵ Regelverksformat for Semantic Web er ikke standardisert ennå. Vi benytter et forslag til format, som verktøyet vi bruker, CWM, forstår.

²⁶ Vi understreker at vi med teori her ikke sikter til den type vitenskapelige teorier som det skal være mulig å utlede testbare hypoteser fra, for eksempel innenfor naturvitenskapen.

teori for å forankre informasjonspolitikken i styringsidealet rettssikkerhet. Dette gjør vi i kapittel 4 ”Behovsanalyse del 1 – offentlig informasjon”.

2.3 Oversikt over oppgavens øvrige metodebruk

I tillegg til metoden vi har valgt for å gi svar på problemstillingen vår, gjør vi i de forskjellige delene av arbeidet bruk av metoder fra ulike fagområder.²⁷ Nedenfor følger en gjennomgang av disse, der vi kort beskriver metoden, hvor den brukes, og hvilke valg vi har gjort. I kapittel 2.1. over har vi beskrevet valget av systemutvikling som *forskningsmetode*. Videre i dette kapitlet går vi nærmere inn på de konkrete valgene for systemutviklingen vi har gjort i denne oppgaven.

2.3.1 Systemutviklingsmetode

Systemutviklingsmetode betegner en metode for å lage et informasjonssystem. Det finnes forskjellige modeller for systemutvikling. Noen modeller beskriver arbeidets faser fra oppstart til ferdig system, eventuelt til systemet fases ut.²⁸ Andre modeller gir anvisning for hvordan deler av arbeidet skal gjøres.²⁹ Grensene mellom de forskjellige typer modeller er flytende. Våre viktigste valg har vært faseorientert utvikling i form av en (delvis) fossefallsmodell, bruk av scenario for å levendegjøre problemstillingen, og bruk av prototyping for å skape forståelse av teknologien og for å kunne vurdere resultatet.

I vårt arbeid har vi valgt en fossefallsmodell for å strukturere prosessen frem til løsningsforslagene. Samtidig er det ikke et arbeid frem mot en endelig løsning. Etersom vi også baserer oss på eksisterende løsninger, kan vårt arbeid sees på som én iterasjon i en spiralmodell. Formålet med en slik iterasjon er å gjøre erfaringer med ukjent teknologi for å kunne vurdere denne. Slik erfaringsbygging kan bidra til å redusere risikoen i et eventuelt videreutviklingsprosjekt for Minside. Et annet forhold er at vi utgjør hele ”prosjektgruppen” selv, og dermed selv gjennomfører de forskjellige fasene. Dette gjør at vi kan bruke erfaringer fra arbeidet i en fase, til å gå tilbake og gjøre justeringer i det som er output fra den foregående fasen. Dette understreker det iterative i arbeidet vårt.

²⁷ Bruk av forskjellige metoder er uunngåelig i tverrfaglige forskningsområde som forvaltningsinformatikk. Dette harmonerer etter vår mening også med hvordan artikkelen ”Design Research in Information Systems” deler forskningsområder i to grupper; ”paradigmatic” og ”pre-paradigmatic” eller ”multi-paradigmatic”: ”Historically, some research communities have been observed to have nearly universal agreement on the phenomenon of interest and the research methods for investigating it; in this page we term these *paradigmatic* communities. Other research communities are bound into a nominal community by overlap in sets of phenomena of interest and/or overlap in methods of investigation. We term these *pre-paradigmatic* or *multi-paradigmatic* research communities.” Se: <http://www.isworld.org/Researchdesign/drisISworld.htm> [30/4-2007]

²⁸ Eksempler på dette er fossefallsmodell og spiralmodell.

²⁹ Bruk av scenario og use-case er en måte å sikre at løsningen tilpasses bruk i konkrete situasjoner. ”Participatory design” beskriver hvordan brukerrepresentanter bør involveres i arbeidet for å sikre at systemet faktisk kan tas i bruk og være til nytte for de som skal bruke det. ”Objektorientert analyse” gir en metode for å transformere representasjonen av problemområdet i den virkelige verden, til datamaskinen. Slike modeller er ofte knyttet til konkrete modelleringsspråk. Bruk av slike analyseteknikker kan bidra til å ”tvinge” de som skal lage en løsning til å konkretisere forhold som er underforståtte i det daglige, men som krever en eksplisitt formulering for å kunne transformeres til et datamaskinsystem. Et typisk eksempel på dette er dataorientert analyse – der man fokuserer på hvilke data et system skal ha – som for eksempel entitets-relasjons-modellering (forkortet ER-modellering, se http://en.wikipedia.org/wiki/Entity-relationship_model). Modelleringsspråket tvinger en blant annet til å ta stilling til *kardinaliteten* i relasjonene (en avdeling *må* ha minst en ansatt, *kan* ha mange). Modelleringsspråkene gir også et verktøy for å kommunisere spørsmål og resultater mellom deltakere i arbeidet. Dersom den samme informasjonen skulle formidles i naturlig språk, vil kravet til presisjon ofte gjøre tekstene tungt tilgjengelige. Schartums transformasjonsmetode er et eksempel på en analyseteknikk for et spesielt problemområde, rettsregler.

Vi bruker scenarioet med Peder og Marte, som en teknikk i flere faser av systemutviklingsarbeidet. For det første, ettersom scenarioet er en beskrivelse av en situasjon som er realistisk, bidrar det til å tydeliggjøre at det er et reelt behov for å finne en god løsning. Slik sett bidrar scenarioet med motivasjon. For det andre er scenarioet egnet til å konkretisere en brukssituasjon, noe som er en hjelp i arbeidet med å konkretisere behovene og de funksjonelle kravene (behovsanalysen og funksjonell spesifisering). For det tredje kan scenarioet være til hjelp for å vurdere løsningen underveis, som et "testcase".

Muligheten til å vurdere resultatet av arbeidet er viktig dersom det skal være noen hensikt i en iterativ prosess. Det er en utfordring å kunne foreta en vurdering kun basert på beskrivelser i form av tekst og tegning. Dessuten er det alltid fare for at ens oppfatning av en teknologi basert på *beskrivelser* av teknologien, ikke harmonerer med hvordan teknologien faktisk fungerer. En viktig del av arbeidet vårt er derfor forsøkene på å "teste" løsningsforslagene i form av delvise prototyper. Prototyping i systemutviklingsarbeid kan ha forskjellige formål. I vår sammenheng er formålet både å gi oss en større forståelse for teknologien – "hands on"-erfaring gir en annen form for læring enn en ren teoretisk tilnærming – og å gi oss et bedre grunnlag til å vurdere resultatet enn hvis vi bare hadde foretatt en beskrivelse.

2.3.2 Analyse av LivsIT og Minside

Siden vi har valgt å la S.M.S. være en videreutvikling av LivsIT og Minside – for å unngå å duplisere funksjonalitet som allerede finnes – trenger vi kunnskap om disse to løsningene. I vår oppgave har undersøkelsen av disse to løsningene og vurderingen av dem i forhold til de behovene vi utleder fra den statlige informasjonspolitikken, først og fremst den funksjonen å fungere som en justeringsmekanisme, et "filter" for behovene vi utleder. Samtidig er det en forutsetning med tanke på at S.M.S. er tenkt som en videreutvikling av Minside at vi kjenner løsningen godt, dersom våre forslag til videreutvikling skal ha noen mulighet til å være reelle.

Her er vi i en spesiell situasjon siden vi begge, gjennom jobbene våre, har arbeidet med både LivsIT og Minside og derfor kjenner godt til disse løsningene fra tidligere.³⁰ Vi har vurdert denne forkunnskapen til å være tilstrekkelig for vårt formål, og vi har derfor ikke laget egne undersøkelsesopplegg for å sette oss inn i disse løsningene. Samtidig har arbeidet med å beskrive og vurdere LivsIT og Minside, i kapittel 5, bidratt til å øke vår kunnskap om disse to løsningene.

2.3.3 Juridisk metode

Problemstillingen vår er knyttet til informasjon om rettigheter. I scenarioet vårt er dette konkretisert gjennom to ordninger, bostøtte og rentefritak. Å fastslå innholdet i disse rettighetene, og å vurdere om og hvor mye Peder og Marte har krav på, er rettslige spørsmål. Juridisk metode gir anvisning på hvordan rettslige spørsmål skal besvares. I motsetning til forskningsmetoder der valg av fremgangsmåte gjøres ut fra hva man mener gir det beste resultatet, er juridisk metode styrt av rettslige normer som f.eks. angir hva man skal, kan og ikke skal eller kan ta hensyn til.

Vi har behov for å kunne fastslå Peder og Martes rettigheter. Vi har i den sammenheng brukt en "modifisert rettsdogmatisk metode"³¹, og kun sett på lover (og stortingsvedtak) og

³⁰ Steinar Skagemo er systemadministrator i Minside og Kjetil Helberg deltok i en arbeidsgruppe ledet av Norge.no for etablering av ny arkitektur for LivsIT (2005-2006).

³¹ Begrepet er fra Schartums presentasjon "Juridisk metode innen forvaltningsinformatikken" og betegner en forenklet fremgangsmåte for å få et resultat som er tilstrekkelig nøyaktig, men som ikke tilfredsstillende metodekravene for å avgjøre spørsmålet i en reell sak. I vår sammenheng er det tilstrekkelig å sannsynliggjøre at

forskrifter, samt håndbøker og informasjon på etatenes hjemmesider. Vi har dessuten brukt Husbankens bostøttekalkulator for hjelp til å tallfeste hvor mye bostøtte Peder og Marte har rett til. Det er viktig å understreke at vi ikke er jurister, og at det i et reelt systemutviklingsprosjekt er nødvendig å bringe inn juridisk ekspertise både for å sikre at juridiske problemstillinger blir identifisert, og besvart i henhold til en fullstendig rettsdogmatisk metode.

I kapittel 4.1 har vi sett nærmere på bakgrunnen for den statlige informasjonspolitikken, og tilknytningen til rettssikkerhetsidealene og forvaltningens veiledningsplikt. På samme måte som ved avgjørelsen av Peder og Martes situasjon, vil resonnementet her være underlagt krav til juridisk metode. I vårt tilfelle er det tilstrekkelig å få påpekt rekkevidden av rettssikkerhetsidealet og at det eksisterer en kobling mellom rettssikkerhetsidealet og den statlige informasjonspolitikken. Her har vi brukt juridisk litteratur, som på sin side bygger argumentasjon i tråd med juridisk metode.

2.3.4 Transformasjon av rettsregler

Schartums transformasjonsmetode, som vi presenterer i dette avsnittet, er en mindre etablert metode enn de øvrige metodene vi har brukt. Dessuten bruker vi denne metoden på et annet område enn den opprinnelig er tiltenkt, og bare utvalgte deler av den. Derfor har vi valgt å gå noe mer i detalj i gjennomgangen av denne metoden, enn vi har gjort for metodene ovenfor.

Som vi har vært inne på vil en form for automatisert veiledning av en bruker i forhold til rettighetsinformasjon, nødvendigvis innebære et informasjonssystem ”med juss i”.³² Det vil si at juss må transformeres over i programkode. Schartum (1993) viser eksempler på at transformasjon av juss til programkode i forbindelse med utvikling av beslutningssystemer i forvaltningen, gjøres på måter som truer rettssikkerheten. I Schartum (2005) gis det forslag til hvordan systemutviklingsarbeidet bør organiseres, og konkrete forslag til hvordan regelverket som skal transformeres kan analyseres. Målet med disse forslagene er å unngå beslutningssystemer som har elementer som enten står i direkte strid med gjeldende rett, eller som ”skaper” rett ved at programkoden uttrykker regler det ikke finnes grunnlag for i de opprinnelige regelverkene. Schartum (2005) påpeker hvordan utvikling av beslutningssystemer, som har trekk til felles både med enkeltsaksbehandling og med forskriftsarbeid, og som kan oppsummeres som forhåndsavgjørelse av enkeltsaker for en ubestemt krets av personer, bør underlegges tiltak som sikrer rettsriktighet og forutberegnlighet.³³

Peder og Marte går glipp av rettigheter ved å vise til den sentrale loven og forskriftene – så lenge de er fiktive personer risikerer de tross alt ingenting på at vi ikke har gått grundigere til verks og undersøkt f.eks. forarbeider og dommer. Foilene fra presentasjonen er tilgjengelige på <http://www.uio.no/studier/emner/jus/afin/FINF5001/v05/undervisningsmateriale/Juridisk%20metode%20innen%20forvaltningsinformatikken.ppt> [14/4-2007]

³² Schartum (2005:6) bruker dette uttrykket for å identifisere ”rettslige informasjonssystemer”. Hans fremstilling dreier seg om en undergruppe av disse, nærmere bestemt informasjonssystemer (som inngår i systemer) for å treffe rettslige beslutninger – ”rettslige beslutningssystemer”. Den andre gruppen rettslige informasjonssystemer er ”rettslige tekstsystemer”. Det systemet vi ønsker å utvikle kan kanskje sees på som en krysning – et rettslig veiledningssystem.

³³ I Schartum (2005) er det disse to rettssikkerhetskravene som er tema. Schartum plasserer sine anbefalinger til tiltak i kategorien ”andre rettssikkerhetstiltak” (Schartum 2005:13), det vil si tiltak som forvaltningen iverksetter for å gå utover rettssikkerhetsgarantiens minstekrav. Hvis man velger å se på utvikling av rettslige beslutningssystemer som en selvstendig form for rettslig beslutningsprosess, som er forskjellig fra både enkeltsaksbehandling og forskriftsarbeid, er det et spørsmål om ikke tiltakene fortjener samme status som rettssikkerhetsgarantiene for de eksisterende to beslutningsprosessene, f.eks. i form av egne regler i forvaltningsloven.

Dersom S.M.S. skal kunne gi automatisert veiledning om rettigheter, vil det i praksis si at S.M.S. må kunne vurdere om et regelverk kommer til anvendelse eller ikke for den personen som skal veiledes. Selv om det ikke dreier seg om et beslutningssystem, mener vi at Scharthums transformasjonsmetode har forslag til analyseteknikker som er egnet for vårt formål. Analyseteknikken han foreslår tas i bruk når det relevante regelverket for et beslutningssystem er kartlagt,³⁴ og gir anvisninger på hvordan reglene i regelverket kan struktureres til en *rettslig informasjonsmodell* og en *rettslig prosessmodell*.

Regelverket skal først sorteres etter *partskategorier* og deretter i henhold til *type regel*. En samlet fremstilling av modellen kan deretter f.eks. gis i en matrise.

Partskategorier er en kategorisering av partene regelverket selv omtaler og forholder seg til. Etter en slik sortering vil det fremgå at noen av reglene i regelverket kun gjelder for noen partskategorier, mens andre regler er felles for flere partskategorier. Regeltypene deles i tre, 1) formelle inngangskriterier, 2) materielle inngangskriterier og regler knyttet til 3) avgjørelsens nærmere innhold. I noen tilfeller er det et pragmatisk valg om det er ønskelig å operere med flere partskategorier, eller å ha færre partskategorier og flere inngangskriterier. Dette har metoden til felles med andre analyseteknikker, som f.eks. ER-modellering, der det må vurderes konkret om en egenskap skal modelleres som en egen entitet som relateres til en annen entitet, eller om det er tilstrekkelig at det settes som et attributt på en eksisterende entitet.³⁵

Slik vi forstår analysemetoden gir den en oppskrift på å finne frem til de alternativene ”stiene” som det er mulig å forfølge gjennom et – for mange – labyrintaktig regelverk. Partskategorier indikerer de forskjellige inngangsportene til labyrinten, og ved hjelp av formelle og materielle inngangskriterier – som er ”sjekkpunkter” underveis – blir personen enten stoppet eller sluppet videre inn til dit hvor reglene om ”avgjørelsens nærmere innhold” trer i kraft. Dermed virker det svært sannsynlig at denne tilnærmingen bør kunne anvendes med tanke på veiledning. Dersom det f.eks. er mulig å fastslå at en person tilhører en partskategori, er vedkommende allerede gitt veiledning til et relevant startpunkt i regelverket. Kan det videre fastslås at vedkommende tilfredsstillter – eller *ikke* tilfredsstillter – de inngangskriteriene som vil møte henne eller ham på veien gjennom regelverket, er det enda mer verdifullt.

Det er viktig å få frem verdien av å kunne *avvise* et regelverk som en del av veiledningen. Dersom det er umulig å automatisere veiledningen i forhold til hvilke regelverk som med sikkerhet er relevante for en person i den forstand at vedkommende vil komme helt frem til reglene om avgjørelsens innhold, vil det gjøre letingen etter slike regler enklere dersom (en del av) reglene som *ikke* er relevante kan markeres (eller skjules). Fra det sistnevnte kan vedkommende også få informasjon om hvilke forhold som gjør at hun eller han ikke tilfredsstillter kravene. Spesielt dersom dette er forhold vedkommende kan gjøre noe med, kan slik informasjon åpenbart være nyttig.

³⁴ Som nevnt i kapittel 2.3.3 forholder vi oss til forenklet rettskildebruk slik at det kartlagte regelverket er mindre omfattende i vårt tilfelle enn det som er nødvendig ved utvikling av et reellt beslutningssystem.

³⁵ Et eksempel på en type egenskap som i noen tilfeller er et attributt og i andre tilfeller en egen entitet, er *adresse*. Ofte er det nok å la f.eks. en person ha et par attributter (adresse, postnummer, poststed) som tilsammen utgjør en adresse. I andre tilfeller ønsker man kanskje å kunne operere med flere adresser som spiller forskjellige roller (besøksadresse, postadresse, fakturaadresse etc), og adressen blir dermed en egen entitet.

Vår løsningsbeskrivelse for S.M.S. vil være en generell løsning, som ikke er tenkt knyttet til et konkret regelverk, men som forhåpentligvis kan brukes som verktøy for å gi veiledning i alle regelverk som på en eller annen måte gjøres tilgjengelige for systemet. Det er m.a.o. ikke behov for å anvende transformasjonsmetoden for et konkret regelverk i forbindelse med utarbeidelsen av S.M.S. som sådan. Men for å kunne gi en vurdering av løsningsforslaget, vil vi bruke vårt scenario, og for å etablere et testcase for dette, trenger vi å transformere regelverket for bostøtte og rentefritak. Vi har derfor valgt å gjøre det i samme kapittel som løsningsforslaget, kapittel 7.

2.4 Mer om Peder og Marte – detaljering av scenario

Som en hjelp til å kunne vurdere om det offentlige lykkes med sin informasjon har vi valgt å lage et scenario, der vi ser på en person i en gitt situasjon, og hva slags informasjon om rettigheter (og plikter) denne personen vil motta. Å bruke beskrivelser av ”typer” på denne måten er en metodologi som ofte brukes i forbindelse med utvikling av nettsteder, og da gjerne under navnet ”Personas”.

Vår ”persona” er Peder Ås. I vårt scenario er han samboer med Marte Kirkerud, og sammen har de nettopp (15. januar) fått en datter som de har gitt navnet Synnøve. Egentlig er det hun som er den viktigste personen; spørsmålet til Peder Ås og Marte Kirkerud dreier seg konkret om hvordan økonomien deres blir når de nå har gått fra å være samboere til å være nybakte foreldre.

Før vi kommer til det spørsmålet, må vi nevne at Peder og Marte møttes på lærerskolen, og at de ble ferdige samtidig, våren 2006. Begge var heldige, og fikk jobb som lærere på hver sin skole i Oslo, der de bor, allerede fra august samme år. Peder har i første omgang riktignok ikke en full stilling, men er ansatt 60 %. Begge har samme lønnstrinn, lønnstrinn 34. For Marte betyr det en månedslønn på 23.341,-³⁶, mens Peder har 14.005,-.

Etter at de begynte å jobbe fikk Marte og Peder råd til å flytte ut av sin lett kummerlige kollektiv-tilværelse, og de har siden september bodd i en to-roms leilighet på 40 m² til 7000,- per måned (ekskl. strøm). Leiligheten er i et borettslag, og er eid av et eldre ektepar som har trukket seg tilbake til en villa i sjøkanten på Sørlandet.

Nå har de nettopp kommet hjem fra sykehuset med sin datter – en litt utålmodig jente som faktisk valgte å komme tre uker før termin. Hun sover nå, og Marte og Peder sitter hjemme foran PC-en for å se hva de har i vente økonomisk. Det eneste de er sikre på er at de vil få hver sin hilsen fra Lånekassen snart. Begge har 300.000,- i studielån, og 15. februar forfaller den første regningen. Antagelig må de ut med nesten 6.000,- kroner hver.

Da de oppdaget at Marte var blitt gravid i juni, hadde de såvidt sjekket hva de hadde i vente av støtteordninger – på www.trygdeetaten.no hadde de funnet en link på forsiden som het ”Jeg skal ha barn” der det stod en del (som de ikke orket å lese akkurat da).

Peder sender derfor nettleseren til www.trygdeetaten.no. De får opp en overraskende kortfattet side som forteller dem at de blir sendt videre til det nye nettstedet til NAV – og begge blir synkront flau for at de allerede hadde glemt at Trygdeetaten (selvsagt) er blitt til

³⁶ Begynnerlønn for lærere i Orkdal kommune: 278.000,-, se: http://orkdalinternett.adcomdata.no/Arkiv//Dokumenter/Saklister%20og%20protokoller/FSSL_20062006.pdf#search=%22begynnerl%C3%B8nn%20%C3%A6rere%22 [28/9-2006].

NAV. Men så lenge Internett husker det for dem, så er jo ingen skade skjedd. På forsiden til NAV, på nav.no, er det riktignok litt trangere om saligheta og nybakte og vordende foreldre er ikke begunstiget med en egen link med så klar tale som på de gamle sidene. Men de finner en "arkfane" med tittelen "Familie og omsorg", og i venstremenyen på den siden de kommer til da, finner de "Svangerskap, fødsel og adopsjon". Peder klikker på det menyvalget, og dermed får de listet opp "Ytelser ved svangerskap, adopsjon og fødsel".³⁷

2.4.1 Stønader for nybakte foreldre

På denne siden nevnes det fem forskjellige stønadsformer, hvorav én, "foreldrepenger ved adopsjon" er åpenbart irrelevant. "Svangerskapsenger" virker også å være noe på siden, all den tid svangerskapet nå er over. De går inn på siden, og konstaterer at intuisjonen var korrekt, dette er en ordning for de som eventuelt må slutte i jobben mens de er gravide. Dermed gjenstår det tre ordninger.

Engangsstønad viser seg å være et tilbud til de som ikke har rett på foreldrepenger. Det er m.a.o. rimelig å sjekke ut foreldrepenger først. **Foreldrepenger** ved fødsel gis til de som "har hatt pensjonsgivende inntekt i minst seks av de ti siste månedene før stønadsperioden tar til".³⁸ August, september, oktober, november, desember ... Ai ... Selv om de tar med halve januar, kommer de ikke lenger enn til fem og en halv måned ... Kan "gradert uttak" redde dem fra en snikende katastrofe? Dessverre, gradert uttak er bare for de som allerede har rett til foreldrepenger. Tilbake til engangsstønaden. Jo, her er det *litt* trøst å få. For foreldre uten krav på foreldrepenger, gis det engangsstønad på 33.358,- kr per barn.³⁹

Marte og Peder leser over de samme sidene en gang til. Og enda en gang. Men de finner ikke noe mer penger, og på et ark skisserer Marte det de vet⁴⁰. Det første halvåret, som hun ønsker å være hjemme med Synnøve, må de klare seg uten inntekten hennes, og engangsstønaden dekker knapt opp for de to første månedene. (Og det er før de har skaffet seg vogn og vaskemaskin, og før Lånekassen har fått pengene de hittil bare har etterlyst).

2.4.2 Hva Peder og Marte ikke vet

Som vi har sett over reduseres inntekten til Peder og Marte betydelig etter at Synnøve er født, ettersom verken Peder eller Marte har opptjent rett til foreldrepenger. (Dette skyldes egentlig Synnøves utålmodighet etter å komme til verden, men det skal vi ikke holde mot henne.) Mye tyder på at det kan bli et stramt år for den lille familien, selv om engangsstønaden på drøyt 33 tusen selvsagt gir en viktig støtte (men ikke mer enn at mer enn to tredeler spises opp av de to forfallene for studielånene deres i februar og mai).

Det vi vet, som Peder og Marte gjerne skulle ha visst, er at deres samlede inntekt nå er så lav, at de har rett på rentefritak hos Lånekassen, slik at de med god samvittighet kan utsette innbetalingen på studielånet for den som er hjemme med Synnøve. Det vil ikke påløpe ekstra renter som følge av denne utsettelsen. Enda viktigere er det at de har rett til bostøtte fra Husbanken. Bostøtte er en ytelse som kan komme opp i flere tusen kroner per måned.

³⁷ Menyvalgene er hentet fra NAVs nettsted <http://www.nav.no> [16/1-2007].

³⁸ Fra informasjonen om Foreldrepenger ved fødsel, <http://www.nav.no/page?id=328> [16/1-2007]

³⁹ Det er neppe helt sannsynlig at dette først blir klart for Peder og Marte *etter* at de kommer hjem fra sykehuset, men det ville ikke påvirket saken mye dersom de hadde visst om det tidligere. Det er kanskje også sannsynlig at de tar en telefon til NAV for å forsikre seg om at det faktisk er slik at de ikke har krav på foreldrepenger fordi de mangler to uker med jobbing. Men det er sent på kvelden, og ingen kontorer holder åpent, og vi har ikke tid til å vente til neste dag, så vi skriver videre.

⁴⁰ Se vedlegg 1 for en oppsummert versjon av økonomien til Peder og Marte.

Vi vil se nærmere på hvorfor vi mener Peder og Marte er omfattet av disse ordningene, og beregne hvor stor påvirkning det ville hatt på deres økonomi. For å avgjøre dette må man se på regelverket for de to ordningene og anvende juridisk metode, nærmere bestemt rettsdogmatisk metode. Som vi har påpekt i kapittel 2.3.3, bruker vi en forenklet utgave av juridisk metode.

2.4.2.1 Lånekassens ordning for rentefritak

Vi har for enkelhets skyld satt terminbeløpet til 6000,- for både Peder og Marte.

Hovedregelen for rentefritaksordningen finnes Lov om utdanningsstøtte, § 8 fjerde ledd, andre punktum:

”Det kan gis rentefritak i forbindelse med videre utdanning, militærtjeneste, sykdom, arbeidsledighet eller omsorgsforpliktelser når låntakers inntekt er under fastsatte grenser, eller hvor andre forhold gjør at innkreving av renter vil virke urimelig”.⁴¹

Rentefritaksordningen henger sammen med ordningen for betalingsutsettelse, § 8 fjerde ledd, første punktum;

”Det kan gis utsettelse med betaling av hele terminbeløp i inntil 3 år eller inntil 6 år ved utsettelse av halve terminbeløp”.

Normalt vil slik utsettelse føre til at det påløper ekstra renter ettersom lånets totale tilbakebetalingstid blir forlenget. Dersom lånetakeren som får betalingsutsettelse også får rentefritak, derimot, unngår man denne ekstra kostnaden. Andre punktum nevner ”omsorgsforpliktelser” i kombinasjon med lav inntekt som kriterium for å kunne få rentefritak. For en nærmere angivelse av disse, ser vi på forskriften om tilbakebetaling som er gitt med hjemmel i lovens § 8, femte ledd, nærmere bestemt ”Forskrift om forrentning og tilbakebetaling av utdanningslån og tap av rettigheter 2007”⁴². Den aktuelle paragrafen her er § 12-1, om rentefritak av sosiale og økonomiske årsaker, der fødsel nevnes som en av flere alternative grunner, og inntektsgrensen spesifiseres. Paragrafen som helhet lyder:

12-1. Rett til rentefritak av sosiale og økonomiske årsaker

En låntaker som oppfyller ett av kravene i bokstav a-d nedenfor og som samtidig har en brutto månedsinntekt på kr 14.980 eller lavere, har rett til rentefritak. Inntektsgrensen blir økt med kr 1.500 for hvert barn under 16 år som låntakeren forsørger. Låntakeren må dokumentere grunnlaget for søknaden om rentefritak.

a) Forhold knyttet til sykdom og arbeidsledighet

Låntaker som dokumenterer

- 1) sykdom,
- 2) medisinsk rehabilitering,
- 3) yrkesrettet attføring,
- 4) tidsbegrenset uførestønad eller
- 5) registrert arbeidsledighet.

⁴¹ § 8, fjerde ledd, 2. punktum, se: <http://www.lovdata.no/all/hl-20050603-037.html#8> [30/4-2007]

⁴² Se: <http://www.lovdata.no/for/sf/kd/kd-20061220-1645.html> [11.2.2007] Vi fant forskriften ved hjelp av Lovdatas hjemmelsregister, se <http://www.lovdata.no/for/sf/sf-20050603-037.html> [11.2.2007]

Det er en forutsetning at forholdet varer i minst tre måneder sammenhengende.

b) *Verneplikt eller elev ved militær skole*

Låntaker som avtjener militær eller sivil verneplikt i Norge, låntaker som er elev ved en militær skole og borgere fra EØS-land som avtjener verneplikt i hjemlandet.

Det er en forutsetning at forholdet varer i minst tre måneder sammenhengende.

c) *Fødsel eller adopsjon*

Låntaker som dokumenterer fødsel og omsorg etter fødsel. Den av foreldrene som har omsorgen for barnet, får rentefritak i inntil tolv måneder fra fødselen. Ved adopsjon av barn under ti år gis det rentefritak etter samme regler i inntil tolv måneder fra adopsjonsdato. I perioder der foreldrene deler omsorgen, kan rentefritaket bli delt.

d) *Omsorgsarbeid*

Låntaker som har omsorgsarbeid som varer i minst tre måneder sammenhengende. Som omsorgsarbeid regnes omsorg for barn under ti år eller omsorg for uføre, funksjonshemmede eller eldre med stort pleiebehov i den nærmeste familien. Se likevel § 12-5.

Ut fra forskriftens § 12-1, kan vi konkludere med at den som er hjemme med barnet, kan få rentefritak ettersom den som er hjemme, Marte, har inntekt lik 0,- (under inntektsgrensen i første ledd), og har omsorg for barnet etter fødselen (punkt c). Med fire forfall i året, og permisjonen fordelt slik at begge er et halvt år hjemme, vil begge to kunne utsette to forfall, altså til sammen fire forfall à 6.000,-, som totalt blir 24.000,- kroner. Dette er ikke penger de får ekstra, men i praksis vil det antagelig være til stor hjelp å slippe å betale disse 24.000,- dette året, og isteden betale dem det året de egentlig skulle være ferdige med studielånet.

2.4.2.2 Husbankens bostøtteordning

Hovedregelen for bostøtte finnes i Lov om Den Norske Stats Husbank, § 16 (1), som lyder:

Etter retningslinjer som fastsettes av Stortinget, skal det gis bostøtte til personer som bor i bestemte grupper av boliger og som boutgiftene virker særlig tyngende for.⁴³

Det er nærmere regler for bostøtte-ordningen i et stortingsvedtak.⁴⁴ Dette vedtaket gir samtidig departementet hjemmel til å bestemme at bostøtte kan gis til andre personer enn de som nevnes i stortingsvedtaket (§ 1, 2. ledd), samt hjemmel til å gi forskrifter for beregning av bostøtte (§ 3). Disse reglene finner vi i "Forskrift om bostøtte fra Den Norske Stats Husbank".⁴⁵ Siden forskriften gjentar stortingsvedtaket med tanke på hvem som kan få bostøtte, har vi konsentrert oss om denne.

Forskriften deler spørsmålet om bostøtte i tre deler. For det første er det spørsmål om hvilke personer som kan få bostøtte (krav til husstanden), for det andre er det spørsmål om hva slags bolig husstanden bor i og krav til denne (krav til boligen) og for det tredje er det utmåling av

⁴³ LOV 1946-03-01 nr 03 om Den Norske Stats Husbank, se: <http://www.lovdata.no/all/hl-19460301-003.html> [28/1-2007]

⁴⁴ Stortingets vedtak av 12. juni 1972 med senere endringer, senest ved vedtak av 23. januar 1997. Stortingsvedtaket er tilgjengelig på Husbankens nettsted for regelverk, se: <http://www.husbanken.no/regelvrk/hb-nr.nsf/webRegelverk/47A783993F270A91C1256488002BA500?OpenDocument> [28/1-2007]

⁴⁵ Forskrift 2004-12-22 nr 1755 fra Kommunal og regionaldepartementet, se: <http://www.lovdata.no/for/sf/kr/xr-20041222-1755.html> [28.01.2007]. I Husbankens egne retningslinjer for bostøtte (se nedenfor) er det forøvrig henvisning til en forskrift gitt av samme departement og med samme navn, men med en annen dato: "Gitt av Hovedstyret i Husbanken med hjemmel i Forskrift om bostøtte fra Den Norske Stats Husbank gitt av Kommunal- og regionaldepartementet 21.3.2005 § 10." Vi klarte ikke å finne denne forskriften på Lovdata.no, hverken ved søk etter forskrifter med ordet "bostøtte" i tittelen, eller når vi gikk via hjemmelsregisteret og fant forskrifter gitt med hjemmel i lov om den Norske Stats Husbank.

bostøtte (inkludert behovsprøving). For det andre spørsmålet finner vi i Peder og Martes tilfelle svaret i forskriften, men for det første og det tredje spørsmålet må vi i tillegg se i Husbankens ”Retningslinjer for bostøtte”,⁴⁶ som er gitt med hjemmel i forskriften.

Spørsmål 1: Hvem kan få bostøtte? For det første må personene som skal få bostøtte være ”folkeregistrert og lovlig bosatt i Norge”, jfr § 1, 2. ledd. Dette er sant for alle i Peder og Martes husstand. Forskriftens § 2, 1. ledd stiller opp alternative krav til hvem som må være medlem av en husstand for at husstanden skal ha krav på bostøtte. Her er det først spørsmål om hvorvidt Peder, Marte og Synnøve er medlemmer av husstanden. I retningslinjene finnes det en legaldefinisjon av husstand, som definerer de som bor i boligen og de som har folkeregistrert adresse i boligen, som medlemmer av husstanden, så sant de ikke bare leier en del av boligen. Med andre ord er alle tre medlemmer av husstanden. Det neste spørsmålet er om et av de alternative kravene til hvem som må være medlem av husstanden, gjelder for dem. Et av disse alternativene er krav om at husstanden må ha barn under 18 år (§ 2, 1. ledd, 1. punkt). Konklusjonen her er at siden Synnøve er medlem av husstanden og under 18 år, kan Peder og Marte få bostøtte.

Spørsmål 2: Krav til boligen? Dette spørsmålet er todelt. Det første spørsmålet gjelder finansiering. Forskriftens § 3 har en hovedregel som tilsier at en bolig må være finansiert med en av fire alternative typer støtteordninger for å kunne få bostøtte, for eksempel oppstartstilskudd fra Husbanken til omsorgsboliger (§ 3, 1. ledd, punkt 4). Hvorvidt det er tilfelle for boligen til Peder og Marte vet vi ikke. Men det er to unntakshjemler som sier at det ikke er krav til finansieringen. Den ene gjelder for borettslagsboliger (§ 3, 2. ledd, punkt 3), den andre for husstander med barn under 18 år (§ 3, 3. ledd, punkt 1). Begge disse unntakshjemplene gjelder for Peder og Marte, finansieringsformen for boligen deres er med andre ord ikke til hinder for å få bostøtte.

Det andre spørsmålet er konkret om boligens størrelse og innhold. Forskriftens § 4, første ledd, første setning stiller krav om at boligen skal ha ”kjøkken, bad, oppholdsrom, soverom og egen inngang”. En to-roms borettslagsleilighet i Oslo vil normalt ha alt dette, så her konkluderer vi positivt. I andre setning stilles det et krav om at boligen må være over 40 m², men andre ledd gir et unntak fra dette kravet for husstander med barn under 18 år (§ 4, 2. ledd, punkt 1). Peder og Marte faller innunder denne unntaksregelen, ettersom deres leilighet ikke er *over* 40 m². Til slutt er det et krav om at boligen må være en helårsbolig (§ 4, 3. ledd, første setning). Dette må vi anta at Peder og Martes borettslagsleilighet er. Vi kan derfor konkludere at med den boligen Peder og Marte har, kan de få bostøtte.

Spørsmål 3: Utmåling: I forskriftens § 5 står regler for utmåling av bostøtte. Det er tre regler. Første ledd angir hva som skal regnes som husstandens *boutgifter*, inkludert en hjemmel for Husbanken til å fastsette utfyllende regler. Andre ledd angir mulighet for tillegg for alders-, uføre og etterlattepensjonister – og er med andre ord ikke relevant for Peder og Marte. Tredje ledd angir beregningsmåten for bostøtte, som er differansen mellom husstandens boutgift og såkalt ”rimelig boutgift”, multiplisert med dekningsprosenten. Her er det for det første nødvendig å fastslå 1) Peder og Martes boutgifter, deretter fastslå 2) ”rimelig boutgift”, og 3) regne ut differansen mellom disse, før vi så fastslår 4) ”dekningsprosenten”, og til slutt 5) multipliserer ”dekningsprosenten” med differansen.

⁴⁶ Se: <http://www.husbanken.as/regelvrk/hb-nr.nsf/webRegelverk/BD7BF0587C7C9CFCC1257134002BC440?OpenDocument> [14.4.2007]

- 1) Boutgifter: Husbankens retningslinjer § 5 litra C gjelder fastsettelse av boutgifter for ”leid bolig”: ”Her medregnes: 1. Søkerens månedlige husleie eksklusive utgifter til lys og oppvarming.” Dette er med andre ord hele husleien på 7.000,- pr måned.
- 2) ”Rimelig boutgift”: Det er ingen eksplisitt definisjon av hva som menes med dette, men i Husbankens retningslinjer § 5a, andre ledd, fremgår det at ”rimelig boutgift” avhenger av ”husstandens samlede inntekter”:
”Ved behovsprøving blir godkjente boutgifter sammenholdt med de boutgifter som det ansees rimelig at husstanden selv er i stand til å betale, når det sees hen til husstandens samlede inntekter, jf § 6, 4. ledd og § 7, 5. ledd.”

De paragrafene det henvises til i retningslinjene er så omfattende at vi ikke lykkes i å finne ut av hva som er Peder og Martes ”rimelige boutgift”, og vi kan dermed heller ikke å beregne hva bostøtten vil bli. Vi har til gjengjeld fastslått med stor grad av sannsynlighet at Peder og Marte *har krav på* bostøtte. For å beregne beløpet velger vi å bruke bostøtte-kalkulatoren på Husbankens hjemmesider.⁴⁷

Beregning av bostøtte med Husbankens bostøtte-kalkulator

Vi har gjort to beregninger – vi forutsetter at inntekten fra den som er hjemme med Synnøve er 0,- i den perioden vedkommende er hjemme. Vi antar videre at bostøtten skal beregnes to ganger, en gang for når Marte er hjemme og Peder jobber, og en gang når de har byttet.

Opplysninger oppgitt i simulatoren: Ingen formue, 40 m², 7000,- i husleie, fremleiet bolig i borettslag som ikke har mottatt støtte fra Husbanken.

Periode 1: Mens Marte er hjemme med barnet:

Alminnelig inntekt Peder: 168.060,- pr år

Personinntekt Peder: 130.000 pr år (*estimert*)

Resultat av beregningen

Din årlige bostøtte er beregnet til kr 24 684

Månedsbetøp kr **2 057**

Periode 2: Mens Peder er hjemme med barnet

Alminnelig inntekt Marte: 280.100,- pr år

Personinntekt Marte: 220.000,- pr år (*estimert*)

Resultatet av beregningen

”Husstandens samlede inntekter og boutgifter gir etter behovsprøving ikke grunnlag for bostøtte.”

Månedsbetøp kr **0,-**

2.4.3 Konklusjon

Peder og Marte går glipp av inntil 2057,- kr i bostøtte i måneden samt mulighet til å utsette innbetalinger på studielånet fordi de manglet informasjon. Til sammen utgjør dette 24.342,- kr i tilgjengelige midler bare i de første seks månedene etter at Synnøve er født⁴⁸. Med andre ord, *Peder og Marte får ikke den rettighetsinformasjon de trenger på en enkel måte!*

⁴⁷ Kalkulatoren er tilgjengelig på <https://tjenester.husbanken.no/bostottesimulator> [30/4-2007]

⁴⁸ Se vedlegg 2 for en oppsummert versjon av Peder og Martes økonomi med bostøtte og rentefritak.

3 Hva er Semantic Web? ⁴⁹

Den overordnede problemstillingen i dette arbeidet er at vi mener innbyggerne ikke får informasjon om sine rettigheter på en enkel nok måte, og vi ønsker å se om Semantic Web kan forbedre dette. For å gå videre med det spørsmålet trenger vi å etablere en forståelse av hva som menes med Semantic Web og hva det mener å muliggjøre.

Ordet ”semantikk” kommer fra gresk og betyr ”vitenskapen om ordenes betydning”.

semantikk~tik'k -en (fra gr 'betydningsvitenskap', av *sema* 'tegn') vitenskapen om ordenes betydning⁵⁰

Så det korte svaret på hva Semantic Web er, er at det dreier seg om en måte for datamaskiner å utveksle dataenes *betydning* på.

Fra før er det definert felles format og syntaks for utveksling av data mellom datamaskiner, for eksempel tegnsettene ASCII og Unicode, og SGML/XML – en standardisert syntaks som brukes til å definere en rekke andre standardiserte språk, hvorav HTML⁵¹ er den viktigste av disse på World Wide Web (WWW). Semantic Web er en videreutvikling av dagens WWW og forutsetter – som navnet antyder – definisjoner av dataenes betydning, på tvers av Internett.

Dette kapitlet har tre hoveddeler. I den første delen (3.1) vil vi starte med å se nærmere på hvorfor de nevnte standardene som brukes på dagens World Wide Web ikke kan løse utfordringene med å definere dataenes betydning. Vi gjør dette gjennom først å se på begrensningene i HTML, før vi forsøker å identifisere et behov for metadata, som en mekanisme til å angi dataenes *opplysningstyper*, ettersom vi har behov for å håndtere dette i henhold til transformasjonsmetoden til Schartum (2005). I neste skritt ser vi om dette isteden kan håndteres med XML, før vi nærmer oss en mulig løsning ved å se på et eksempel på en fremvoksende mekanisme for å gjøre data ”smartere”.

”Smarte data” kan bidra til å forenkle mange av de trivielle tingene vi gjør til daglig. Men den underliggende ideen har et større potensiale. I den andre hoveddelen i dette kapitlet (3.2) vil vi derfor gå i dybden på det som etter vår mening er hovedteknologien, eller *byggestenen*, i Semantic Web, nemlig den underliggende datamodellen, RDF.⁵² Ideen er at alle data skal kunne gjøres tilgjengelig som RDF-data, noe som åpner for ”web of data”. I dette ”dypdykket” i RDF bruker vi blant annet den semiotiske triangelen, som ofte trekkes inn når semantiske teknologier skal forklares. Vi gir videre eksempler på hvordan RDF kan representere både konkrete opplysninger og opplysningstyper, og tilbyr et sett av uttrykk som muliggjør datamodellering slik vi er vant til fra utvikling av informasjonssystemer med

⁴⁹ ”Semantic web” refererer noen ganger til en visjon som kan realiseres ved hjelp av flere alternative (dels konkurrerende) teknologier, som for eksempel Topic Maps, Core Components eller standardene fra W3C. Andre ganger refererer det kun til den sistnevnte teknologien. Det er denne betydningen vi bruker i vår fremstilling.

⁵⁰ Fra Dokumentasjonsprosjektet ved UiOs ordbok på nettet, se:

<http://www.dokpro.uio.no/perl/ordboksoek/ordbok.cgi?OPP=semantikk&bokmaal=S%F8k+i+Bokm%E5lsordboka&ordbok=bokmaal&alfabet=n&renset=j> [30/4-2007]

⁵¹ HTML kan sees på som en konkret anvendelse av datidens generelle ”markup language”, nemlig Structurized Generalised Markup Language” (SGML). Men denne koblingen er – ihvertfall i praksis – blitt ”løs”, i den forstand at det som er gyldig HTML ikke nødvendigvis er gyldig SGML. Det kan også ha sammenheng med at HTML ble utviklet av noen som kjente til SGML men ikke var ekspert i det. Dette bekreftes mer eller mindre også i et diskusjonsarkiv om videreutviklingen av HTML, fra 1994: <https://listserv.heanet.ie/cgi-bin/wa?A2=ind9412&L=html-wg&P=582> [30/4-2007]

⁵² RDF står for ”Resource Description Framework”.

konvensjonelle teknologier som f.eks. relasjonsdatabaser. Som en avslutning på denne delen forsøker vi å se for oss hvilke muligheter det åpner for dersom personopplysninger om brukeren tilgjengeliggjøres som RDF-data.

I den siste delen (3.3) går vi igjennom hele stabelen av teknologier som til sammen er ment å utgjøre Semantic Web, og der RDF utgjør byggestenen. I denne delen samler vi de konkrete faktaene om hvorvidt, og i såfall hvor, teknologien er standardisert, samt beskriver kort hva de er ment å tilføre.

Til slutt (3.4) oppsummerer vi kort gjennomgangen og gir en foreløpig konklusjon på spørsmålet om forholdet mellom Semantic Web og modellene fra Schartums transformasjonsmetode.

3.1 Fra maskinlesbar HTML til maskinforståelige ”smarte data”

Da Tim Berners-Lee utviklet WWW på begynnelsen av 90-tallet,⁵³ definerte han det som senere skulle vise seg å bli nettets ”lingua franca”⁵⁴ for hvordan dokumenter skulle vises for brukeren, i form av ”Hypertext Markup Language”, HTML. I praksis innebærer HTML at det ble definert et sett med elementer, eller ”tagger”, som hver representerer deler som går igjen i dokumenter. Blant annet var det elementer som representerte forskjellige former for overskrifter, avsnitt, lister og tabeller. I tillegg var det en spesiell konstruksjon for referanser til andre dokumenter, såkalte *ankere* eller *hyperlinker*. Dette var et avgrenset sett med elementer, og kravet til nettleserne var at de forstod betydningen av hvert element, og presenterte det for brukeren deretter; teksten som var omsluttet av et <h1>-element, ble vist som en hovedoverskrift, og tekst omsluttet av <a>-element, ble en hyperlink som brukeren kunne klikke på, hvorpå nettleseren så viste dokumentet.

Siden den første versjonen er HTML blitt utvidet både i form av flere elementer⁵⁵, og i form av muligheter til å tilordne hvert element en kategori eller en identifikator. Dette brukes gjerne i kombinasjon med språket for å styre hvordan innholdet på siden skal presenteres, såkalte stilark (Cascading Style Sheets, CSS). I stedet for å bruke nettleserens standardfont for overskrifter, kan nettsideskapere angi at elementer skal vises med spesifikke skrifttyper, -størrelse og farve. Ved å tilordne et element en kategori (class) eller identifikator (id), kan det angis at formateringsinstruksjonene i stilarket bare skal gjelde for en enkelt, eller et utvalg av overskriftene på en side. For eksempel kan det angis at hyperlinker som er listet i venstremenyen skal se annerledes ut enn de som listes i den løpende teksten, ved at menylinkene tilordnes kategorien ”meny”:

```
<a class="meny" href="/om_oss.html">om oss</a>  vises forskjellig fra  
<a href="/om_oss.html">om oss </a>
```

I motsetning til elementnavnene, som det finnes en uttømmende liste over, kan forfatteren som lager nettsiden lage akkurat de kategoriene eller identifikatorene vedkommende selv

⁵³ Hannemyr, Gisle (2005) ”hva er INTERNETT”, Universitetsforlaget

⁵⁴ Lingua franca=et felles språk brukt av personer med forskjellige språk

⁵⁵ Et element som kan ha hatt stor betydning, var elementet for å knytte bilder til teksten, -elementet. Dette ble lagt til ”de facto”, ved at Marc Andreessen la til støtte for et sånt element i sin Mosaic-nettleser. Ettersom dette fort ble en av de mest populære nettleserene (kanskje nettopp på grunn av denne støtten) var det ikke noen vei utenom å legge til -elementet i HTML-standardene etterhvert. For annonseringen av -elementet, se: <http://1997.webhistory.org/www.lists/www-talk.1993q1/0262.html> [30/4-2007]

ønsker, ettersom det også er forfatteren selv som styrer formateringsinstruksjonene i stilarket. På denne måten har HTML en form for utvidbarhet.

3.1.1 HTML – maskinlesbart, men helt uforståelig

Dersom en bruker går på et nettsted som beskriver et arrangement, vil han mest sannsynlig finne data om *hvor* arrangementet finner sted, i form av en adresse, *når* det finner sted, *prisen* for å delta, og *betalingsinformasjon* som f.eks. kontonummer, adresse og referansenummer. Denne informasjonen er nødvendigvis maskinlesbar hvis den er kodet i HTML – men maskinen vil ikke forstå *betydningen* av dataene (semantikken), og kan følgelig ikke gjøre bruk av dem. Det er opp til brukeren å foreta koblingene mellom tidsangivelsen på siden og det samme tidspunktet i sin kalenderapplikasjon, og brukeren må selv fylle ut kontonummer og beløp i nettbanken, for å betale påmeldingen. Det er brukeren som ved hjelp av sin kunnskap får informasjon som han eller hun kan anvende, fra disse dataene.

Dersom de nevnte dataene på nettsiden hadde vært utstyrt med tilleggsinformasjon, populært kalt metadata, som sa eksplisitt hva slags betydning de har, ville det være en relativt enkel sak å automatisere den jobben som ellers må gjøres manuelt. Med et klikk ville brukeren kunne overføre informasjonen om begivenheten til kalenderen sin, og få opp en ferdig faktura i nettbanken for betaling. I vår fremstilling vil vi betegne en datamaskins evne til å identifisere data og kunne foreta operasjoner på dataene som beskrevet over, som at datamaskinen har *kunnskap*.⁵⁶ Dette innebærer ikke at datamaskinene nødvendigvis må gjøres smartere, for eksempel gjennom økt prosesseringskraft, enn de er i dag. Isteden kan dataene selv gjøres ”smartere”, og på den måten lette jobben med å instruere maskinene til å gjøre det vi ønsker.

Når vi sier at datamaskinen har kunnskap i avsnittet over, er det for å beskrive at den er i stand til å *få noe ut av* de dataene den presenteres for, i den forstand at datamaskinen kan gjøre operasjoner på eller som følge av dataene. Bing (1982) forklarer skillet mellom *data* og *informasjon* på en måte som er nyttig her. Data blir ikke til *informasjon* før de har kommet frem til noen som har midler til å forstå dem, og først da kan dataene transformeres til *kunnskap* hos mottakeren. I vår sammenheng vil vi utvide dette med å presisere at vi mener kunnskap inngår i de nevnte ”midler til å kunne forstå”⁵⁷ dataene. Med andre ord gir den kunnskap mottakeren allerede har, og som gir midler til å forstå data slik at de blir informasjon, *også* mulighet til å utvide mottakerens kunnskap (transformere fra informasjon til kunnskap). Dette igjen utvider mottakerens mulighet til å forstå flere data i neste omgang.

Som nevnt kan tilleggsinformasjon om dataene, metadata, gjøre dataene ”smartere”, og dermed øke muligheten for at datamaskinene kan transformere dem fra data til informasjon. Elementene i HTML er en form for metadata. For eksempel har HTML et element, <ADDRESS>, som virker nyttig i den forstand at det forteller mer om hva innholdet i elementet faktisk er, og ikke hvordan det skal presenteres på en nettside. Men i praksis er dette elementet altfor generelt til å kunne være til hjelp alene. Det er uklart om det er en postadresse eller en besøksadresse, en mottakeradresse eller avsenderadresse. Dessuten trenger vi å kjenne igjen de enkelte feltene i en adresse, å se forskjell på gateadresse, postnummer, poststed og land. Og til slutt må vi jo også vite *hva* som har denne adressen. Er det adressen til organisasjonen som eier nettstedet eller er det begivenheten nettstedet omtaler, og hva hvis nettstedet isåfall omtaler flere begivenheter? Om vi tross alt dette likevel skulle

⁵⁶ Hvorvidt og hvordan datamaskiner har kunnskap, er en stor diskusjon blant annet innenfor forskningsområdet kunstig intelligens (AI). Vi tar ikke stilling til denne diskusjonen her, men bruker ”kunnskap” i en praktisk betydning, at datamaskinen er i stand til å gjøre noe med data den mottar, slik det fremgår av neste avsnitt.

⁵⁷ Bing (1982:67)

lykkes i å gjøre praktisk bruk av <ADDRESS>-elementet, er det uansett bare et lite steg på veien. Hva med elementer for tidspunkt og betalingsinformasjon?

3.1.2 Nærmere om metadata og opplysningstyper

Fra avsnittet over kan vi konkludere med at HTML mangler en generell mekanisme for gjøre data ”smartere”. Men avsnittet inneholder også en rekke eksempler på opplysninger, eller rettere sagt *opplysningstyper*, som for eksempel ”adresse”, ”postnummer”, og ”betalingsinformasjon”. I Schartum (2005) gis det anvisning på hvordan regelverk som skal transformeres til beslutningssystemer bør analyseres for å identifisere *opplysningstypene*. Som et eksempel på opplysningstype nevnes ”kjønn”, og som et eksempel på en konkret opplysning av typen ”kjønn” nevnes ”kvinne”⁵⁸. Skillet mellom opplysningstype og opplysning er generelt nyttig i den videre gjennomgangen av Semantic Web, men spesielt i vårt tilfelle, ettersom vi skal knytte bruken av Semantic Web til rettslig informasjon, ved bruk av transformasjonsmetoden i Schartum (2005) der opplysningstyper er tema.

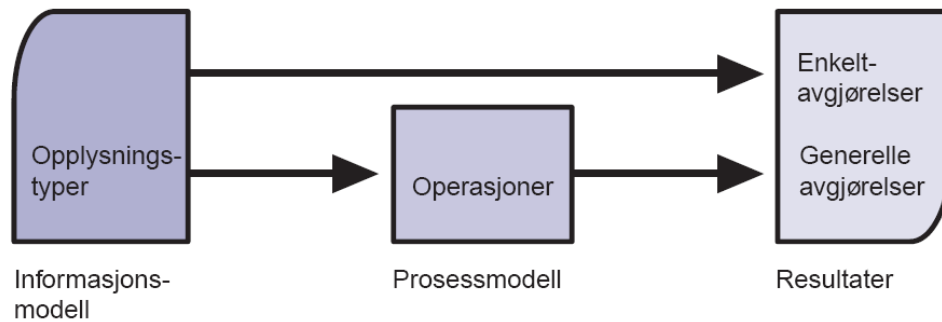
Som eksempelet med <ADDRESS> over viser, er det å oppgi hvilken *opplysningstype* data har som ofte er formålet med metadata. Informasjon om opplysningstypen gjør det enklere å instruere datamaskinen. Dersom en datamaskin mottar dataene ”FREDAG”, kan det både være kontaktinformasjon til vennen til Robinson Crusoe, og en ukedag. Dersom ”FREDAG” ledsages av metadata som sier at opplysningstypen er ”ukedag”, kan slik forveksling vanskelig skje.⁵⁹ ”FREDAG” er selvsagt et unødvendig sært eksempel. For tallverdier derimot, er det åpenbart behov for å vite opplysningstypen for å kunne ha nytte av verdiene.⁶⁰

Å angi opplysningstypen til data gir informasjon om dataens *betydning* og er derfor en viktig del av Semantic Web. Dersom en datamaskin er programmert med operasjoner som skal foretas på visse opplysningstyper, vil informasjon om opplysningstypen være det som er nødvendig for å koble den kunnskapen som ligger i datamaskinen, med de dataene den mottar, og som den dermed vil være i stand til å transformere til *informasjon*, i henhold til de definisjonene av data, informasjon og kunnskap vi bruker. Schartum (2005:58) har en modell som viser hvordan en rettslig informasjonsmodell med opplysningstyper, går (helt eller delvis) via en prosessmodell med operasjoner, og resulterer i avgjørelser.

⁵⁸ Schartum (2005:57)

⁵⁹ Verdien av å vite opplysningstypen forutsetter selvsagt at programmet har programmert inn støtte for dette, eller med våre ord, har ”kunnskap” til å utnytte informasjonen.

⁶⁰ Tallverdier formidles ofte i tabeller. Der fremgår gjerne opplysningstypen av rad- og/eller kolonneoverskriften, enten direkte eller indirekte. Data kodet som tabell i HTML vil vises som en tabell, og det er lett for leseren å se dataenes tilhørende opplysningstyper. Men hvis de samme data skal leses ved hjelp av skjermlesere som oversetter til ”lineære” format som f.eks. tale eller braille, vil det, dersom ikke forfatterne av tabellen har hatt denne typen brukere i tankene (jfr Web Accessibility Initiative, <http://www.w3.org/WAI> [30/4-2007]), være svært vanskelig å forstå innholdet i tabellen. En kalender for april 2007 som har ukedagene som kolonneoverskrifter, vil bli lest som følger: ”Mandag tirsdag onsdag torsdag fredag lørdag søndag 1 2 3 4 5 6 7 8 9 10 11 12 (osv)”.



Figur 1: Schartums modell som illustrerer rettslig informasjonsmodell, prosessmodell og resultater

Et spørsmål er om Semantic Web kan understøtte både informasjonsmodellen og prosessmodellen i Schartums modell. Men før vi kommer inn på det vil vi se om XML representerer en mulig løsning.

3.1.3 XML – for høye forventninger, men god på bunnen

Vi har sett at HTML ikke har en generell mekanisme for å formidle metadata i form av hvilke *opplysningstyper* dataene har. Hva så med XML? Det er en standard som nettopp har utvidbarhet (eXtensible Markup Language) som hovedformål. Her ville det være opp til hvem som helst å legge til hvilke som helst elementer, i sin egen anvendelse av XML. Men nettopp her ligger problemet. XML er det samme for *syntaks* som ASCII og Unicode er for tegnsett. De sistnevnte tegnsettstandardene gjør at det ikke trenger å være (så mye) tvil om hvilken bokstav eller hvilket tegn en gitt kombinasjon av null og ett-tall, er ment å symbolisere. XML på sin side gjør at partene som kommuniserer sparer mye arbeid ved at de ikke trenger å avtale syntaksen på det innholdet de skal utveksle. For eksempel er det utvetydig når et XML-dokument slutter, nemlig ved avslutningstagger som matcher åpningstagger. Siden syntaksen er gitt, kan generelle verktøy brukes til å kontrollere at dokumentet er komplett, i den forstand at det f.eks. ikke mangler avslutningstagger. Videre, ved å knytte et XML-dokument til et "skjema" (XML Schema) som sier hva som er lovlige elementer i et dokument og hva som er lovlige verdier for de enkelte elementene, kan generelle verktøy også brukes til å kontrollere "lovligheten" av innholdet. Dette gjør at utviklere av systemer som tar imot og behandler data på XML, kan spare mye av det utviklingsarbeidet som ellers må brukes til å sjekke om verdiene er på plass, og er innenfor lovlige verdier, før det prosesseres videre.

Selv om XMLs anvendelighet i form av forutbestemt syntaks for utvekslingsformat har gjort XML populært, er ikke XML alene løsningen på behovet for "smartere" data. Det er til liten nytte at man kan legge til de elementene man trenger, hvis det fører til at samme opplysningstyper får forskjellige navn i forskjellige dokumenter.

Fordelene og utfordringene med XML er tydelig i dagens Minside. I registertjenestene sender Minside en forespørsel til tjenesteleverandørene, i form av et XML-dokument som inneholder fødselsnummeret til brukeren, samt språkvalg og sikkerhetsnivå. Som svar kommer et XML-dokument som inneholder opplysningene knyttet til dette fødselsnummeret. Siden Minside vet at det er et XML-dokument som kommer i retur, har Minside nok informasjon til at det er mulig å anvende XML-transformasjon til HTML, ved hjelp av generelle XML-verktøy. Men fødselsnummeret navngis forskjellig i forespørselen fra Minside og i svaret fra registertjenestene, noe som understreker at XML alene ikke gir noen god løsning på å eksplisitt angi hva slags opplysningstype en opplysning har.⁶¹

⁶¹ I forespørselen fra Minside heter fødselsnummer "personid", og i svaret fra NAV heter fødselsnummer "fnr". Se eksempel på request- og response-XML for Minside i kapittel 5.2.2.2

De ovennevnte problemene med XML kan oppsummeres som følger:⁶² XML åpner for at

- 1) forskjellige data kan plasseres i samme element;
<Amount>NOK100 <Amount>
<Amount>HundreKroner<Amount>
- 2) samme data kan gis forskjellig struktur;
<Amount>NOK100 <Amount>
<Currency>NOK<Currency><Amount>100<Amount>
- 3) samme data kan plasseres i forskjellige element;
<Amount>NOK100 <Amount>
<Price>NOK100<Price>

3.1.4 Smarte data: Felles vokabular på felles format

”Smarte data” forutsetter en enighet om et ”standardvokabular”, som innebærer at dataenes betydning er avtalt mellom de som skal bruke dataene. Dessuten trengs det et felles format å pakke dataene inn i. En måte å gjøre dette på bruker mulighetene som ligger i class-attributtet til å lage såkalte ”microformats”. Et eksempel på et standardvokabular, som har eksistert lenge, er vokabularet for visittkort, vCard⁶³. Dette vokabularet definerer et sett med opplysningstyper som er vanlige for kontaktinformasjon (visittkort), og et format for å lagre visittkortet i et tekstformat i en standard tekstfil. De fleste e-postprogrammer kan lese innholdet i en slik tekstfil og automatisk legge til innholdet i brukerens adresseregister. Nedenfor er et eksempel på en slik tekstfil som i dette tilfellet kun inneholder en persons navn:⁶⁴

```
begin:vcard
fn:Ola Nordmann
version:3.0
end:vcard
```

Microformats.org har laget et Semantic Web-inspirert format for dette standardvokabularet. De har definert et format for vCard-vokabularet for bruk i HTML, kalt hCard. Gjennom hCard benyttes det samme vokabularet – dataene gis med andre ord den samme betydning – som i det opprinnelige vCard-formatet, men med den fordel at det kan skje direkte i nettsidens kildekode. Nedenfor er et eksempel på den samme informasjonen som i eksempelet over, generert med Microformats hCard-generator⁶⁵ slik at det kan limes direkte inn i HTML-koden til en nettside:

```
<div id="hcard-Ola-Nordmann" class="vcard">
<span class="fn">Ola Nordmann</span>
</div>
```

Microformats er et eksempel på (gjen)bruk av et standardvokabular for beskrivelse av informasjon som det allerede er stor støtte for – i form av at det er logisk i eksisterende applikasjoner for å gjøre bruk av opplysningene – som egner seg for bruk på Internett, fordi det baserer seg på de samme standardene vi bruker for å lage nettsider (HTML). Det finnes

⁶² Analysen med eksempler er hentet fra Øyvind Aasves introduksjon på Norstellas frokostseminar om semantisk interoperabilitet, 14. mars 2007, se: <http://www.norstella.no/semantiske-teknologier-ulike-teknologier-til-ulik-bruk-presentasjonene-er-lagt-ut.431794-1177.html> [16/3-2007]

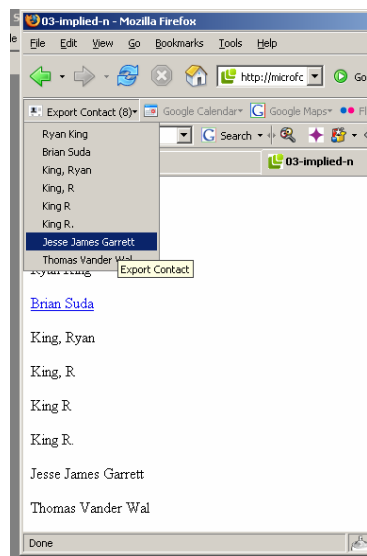
⁶³ Se mer om vCard på Wikipedia: <http://en.wikipedia.org/wiki/VCard> [16.3.2007]. For selve standarden (v.3.0), se: <http://tools.ietf.org/html/rfc2425> [16/3-2007]

⁶⁴ Det er selvsagt mulig å ha med andre opplysningstyper. For eksempel finnes *org* for organisasjon og *adr* for adresse.

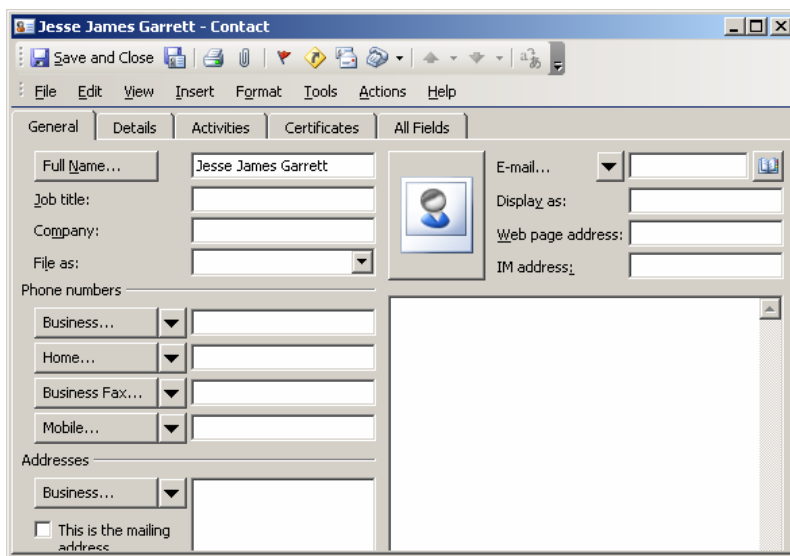
⁶⁵ Se: <http://microformats.org/code/hcard/creator> [16/3-2007]

flere standardvokabular som har fått sine microformat-varianter, bl.a. annet hCalendar for kalenderhendelser, og rel-license for angivelse av rettighetsinformasjon for innholdet på en nettside.⁶⁶

Med kontaktinformasjonen på en nettside kodet med en avtalt syntaks og vokabular, er det klart for applikasjonene vi bruker til å utnytte denne informasjonen direkte, uten at brukeren må klippe og lime. Nedenfor vises det hvordan et tillegg til nettleseren Firefox, kalt ”Operator”,⁶⁷ gjør det mulig å overføre kontaktinformasjon kodet som hCard på et nettsted, direkte til adresseboken på brukerens lokale maskin.



En nettside med kontaktinformasjon kodet i hCard-formatet vises i nettleserapplikasjonen Firefox med Operator-tillegget. Brukeren kan velge ”Export Contact Information”



Ved klikk på ”Export Contact Information” i nettleseren, åpner adressebokapplikasjonen et nytt kartotek kort for kontaktinformasjon. De feltene som var kodet i nettsiden er forhåndsutfyllt.

hCard er et eksempel på et såkalt *microformat*. Vi har brukt det for å illustrere ”smarte” data, og hvordan dette kan gjøre små gjøremål i det daglige enklere, ved at vi får utnyttet maskinene bedre.⁶⁸ Dette er et eksempel på Semantic Webs muligheter. Men Semantic Web

⁶⁶ Se Jon Bings artikkel om CopyMark for informasjon om Creative Commons og andre forslag til å angi rettighetsinformasjon på en nettside: <http://efn.no/copyright/copymarks.html> [16.3.2007]. Oversikt over eksisterende Microformat-spesifikasjoner finnes på http://microformats.org/wiki/Main_Page#Specifications [30/4-2007].

⁶⁷ Se: <https://addons.mozilla.org/firefox/4106/> [16/3-2007]. I tillegg til støtte for hCard-formatet, kan Operator også forstå hCalendar-formatet (og eksportere hendelser til brukerens Google-kalender), informasjon om lokalisering, for visning av sted i Google Maps, samt kobling til kategoriserings-”tag” på såkalte ”folksonomy-tjenester” som Del.icio.us, Flickr og Technorati. Dersom hCard-formatet blir en utbredt standard, vil vi anta at støtte for det etterhvert blir bygget inn i nettlesere og andre programmer uten at brukeren vil trenge å laste ned et eget tillegg.

⁶⁸ Vi har valgt å illustrere dette ved hjelp av microformats, selv om det ikke er en W3C-standard. W3C jobber med forslag til hvordan man integrerer RDF-data for å oppnå ”smarte” data, i vanlig HTML-sider. Et av forslagene er bl.a. en kombinasjon av RDFa og GRDDL. Forslagene forutsetter at det gjøres en liten endring i HTML-standarden, noe som ikke er nødvendig ved bruk av microformats. Fordelen er at løsningen gir full frihet til å uttrykke den grunnleggende datamodellen Semantic Web fra W3C baserer seg på, den såkalte RDF-

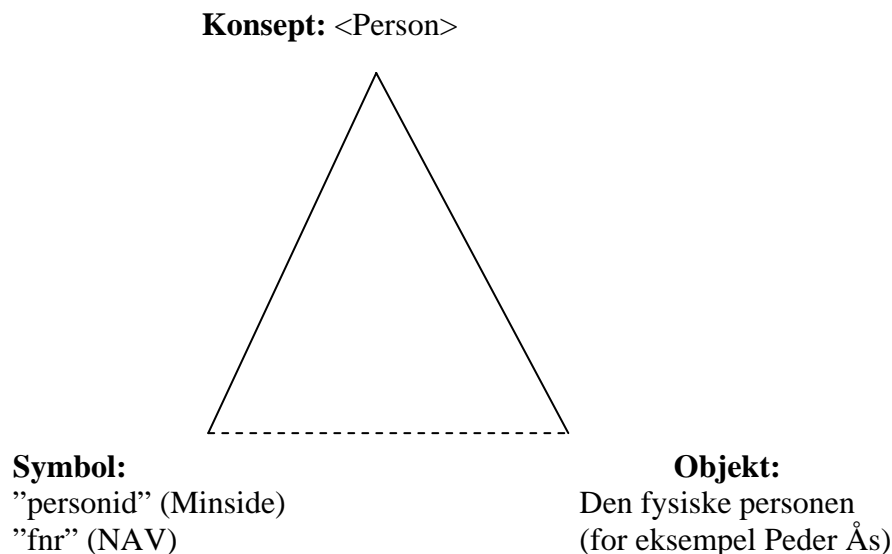
kan være mye mer enn et verktøy for bekvem surfing, som vi skal vise i denne oppgaven. Før vi gjør det vil vi gå nærmere inn på det som er byggestenen i Semantic Web, nemlig datamodellen RDF.

3.2 RDF: En datamodell for beskrivelse av data

Som vi så over er Microformats en måte å angi opplysningstyper for data på nettsted, som *nesten* er fullt ut kompatibel med RDF-datamodellen som er grunnlaget for Semantic Web. I dette kapitlet skal vi se nærmere på denne datamodellen. For å forklare hvorfor den er valgt, ser vi først på den generelle utfordringen med å få bokstaver på et papir (eller en fil) til å formidle mening.

3.2.1 Den semiotiske trekanten

Den semiotiske trekanten blir ofte brukt som illustrasjon på hva semantiske teknologier er.⁶⁹ Tanken er at semantiske teknologier gjør maskiner ”i stand til” – det vil si for praktiske formål – å forstå hvilket *konsept* (og eventuelt indirekte hvilket objekt) et symbol refererer til. I henhold til den semiotiske trekanten er et objekt en fysisk eller abstrakt ting, som et symbol, f.eks. en tekststreng, kan referere til. Forbindelseslinjen mellom objektet og symbolet er ikke direkte, men går gjennom *konseptet*, derav trekanten. Å koble mellom et symbol og et konsept er å forstå meningen i symbolet, med andre ord symbolets *semantikk*.



Når en datamaskin er utstyrt med programmeringsinstruksjoner som opererer på *symboler*, vil den ikke være i stand til å vite at "personid" og "fnr" er synonymer og betyr det samme. Det vil si at det semantiske innholdet ikke er likt, med mindre dette programmeres eksplisitt.⁷⁰ På

datamodellen, som vi skal se nærmere på i neste kapittel. For informasjon om RDFa og GRDDL se henholdsvis <http://www.w3.org/TR/xhtml-rdfa-primer/> og <http://www.w3.org/TR/grddl/> [30/4-2007]

⁶⁹ Se f.eks. Wikipedia for eksempel på beskrivelse av den semiotiske triangelen, http://en.wikipedia.org/wiki/Semiotic_triangle [30/4-2007]. For eksempel på bruk av den semiotiske triangel for å beskrive semantiske web-teknologier, se Øyvind Aasves presentasjon på Norstellas frokostseminar om Semantisk interoperabilitet 14. mars 2007, <http://www.norstella.no/getfile.php/416811.177.twfbfvwvde/SIMSemIntro.pdf> [16/3-2007]

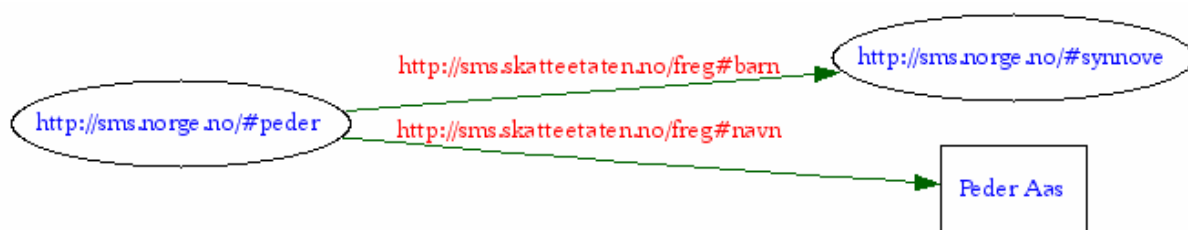
⁷⁰ I forbindelse med testingen av innloggingsløsningen til Minside høsten 2006, viste det seg at brukerne ikke fikk tilsendt midlertidig passord på e-post dersom de hadde valgt et annet språk enn engelsk når de ba om det,

samme måte vil datamaskinen ikke forstå at homonymer kan referere til forskjellige ting. I praksis innebærer det at et program bare kan operere på data innenfor et avgrenset domene, for eksempel i form av spesifikke format, hvor det er kontroll på hvilke symboler som brukes til hva.

Selv om Semantic Web på sin side ikke frigjør datamaskinene fra å operere på symbolene, så representerer den en løsning for standardisert håndtering av de utfordringene man har blant annet ved håndtering av synonymer og homonymer, og avgjørelse av det semantiske innholdet i symbolene. Semantic Web gjør dette ved å etablere en felles, generisk *datamodell*, som inkluderer bruk av URIer⁷¹ som en unik måte å identifisere objektene i datamodellen på.

3.2.2 RDF-tripler: Objekt, predikat, subjekt

Datamodellen til Semantic Web, RDF, er ment å være så enkel at alle andre datamodeller skal kunne overføres ("mappes") til RDF. Dette inkluderer modellen for relasjonsdatabaser – som er en av de mest vanlige datamodellene for data i dag.⁷² RDF-modellen består av "tripler". Hvert trippel består av et subjekt, et predikat og et objekt, og kan representeres som en rettet graf. Subjektet og predikatet *må* være en URI, mens objektet kan være en tekststreng.⁷³



Figur 2 -- Eksempel på RDF-data: To RDF-tripler om Peder Ås (navn og barn)

Eksempelet over er en grafisk representasjon av to tripler. Subjektet er det samme i begge triplene, og representerer S.M.S-brukeren 'Peder'. Oversatt til naturlig språk kan dette leses som "personen 'Peder' har barnet 'Synnove' og navnet 'Peder Aas'",⁷⁴ Navnet er representert som en tekststreng, mens 'Synnove' er representert som en URI.⁷⁵ Nedenfor bruker vi eksempelet til å illustrere 1) hvordan RDF bruker URIer for å unngå navnekollisjoner, 2)

fordi den programmerte testen sjekket om brukeren hadde valgt "mail", mens ved bruk av den norske versjonen så var valget "e-post".

⁷¹ URI står for "Uniform Resource Identifier", og er standardisert av IETF i RFC 3986, se:

<http://tools.ietf.org/html/rfc3986> [21.4.-2007]. Se mer om bruken av URI i punkt 1) i neste kapittel (3.2.2).

⁷² Tim Berners-Lee argumenterer for RDFs kompatibilitet med relasjonsdatabasemodellen i "Relational Databases and the Semantic Web", se: <http://www.w3.org/DesignIssues/RDB-RDF.html> [19/4-2007]. Et forum bestående av representanter fra W3C og industriaktører innen databaser fastslår i "The Cambridge Communiqué" fra 7. oktober 1999 at RDF er et medlem av ER-modelleringsfamilien, se:

<http://www.w3.org/TR/schema-arch> [21/4-2007]

⁷³ Det innebærer at objekter som er en tekststreng ikke kan være subjekt eller predikat i andre tripler, i motsetning til objekter som er en URI.

⁷⁴ Verktøyet vi har brukt for å lage grafisk representasjon av våre data håndterer dessverre ikke særnorske tegn særlig pent, så vi har derfor vært nødt til å døpe om Peder Ås og Synnøve. Verktøyet er tilgjengelig på <http://www.mindswap.org/2003/pellet/demo.shtml> [21/4-2007]

⁷⁵ En konvensjon i den grafiske representasjonen av RDF er å tegne URIer som sirkler og tekststrenger som firkanter. I beskrivelsen av dette eksempelet angir vi symbolene, som f.eks. navnet "Peder Ås" med doble hermetegn (""), mens når vi snakker om konseptet (eller den semantiske betydningen) angir vi det i enkle hermetegn ('). Dette er ment å tilsvare bruken av doble og enkle hermetegn i Jon Bings artikkel "Kunnskap" i Yulex 2005, se fotnote 1, side 7. Når de samme opplysningene er uttrykt som RDF vil en URI representere betydningen (sirkler), mens en tekststreng representere symbolene (firkanter).

hvordan URIer kan forkortes for bedre lesbarhet, og 3) hvordan de samme dataene kan gis en tekstlig representasjon. Deretter prøver vi å relatere skillet mellom *opplysningstyper* og *opplysninger* til vårt RDF-eksempel.

1) Bruk av URIer for å unngå navnekollisjoner

En hovedutfordring når data skal brukes på tvers av systemer, er å håndtere navnekollisjoner – samme symbol har forskjellig betydning i forskjellige systemer, og forskjellige symboler brukes i samme betydning. I RDF håndteres dette ved at navn (symbol) må ”kvalifiseres” ved å angi et navnerom (”namespace”⁷⁶). Dette gjøres ved å legge til en URI og en ”#” før navnet. URIen gjenspeiler et domene (og eventuelle undernivåer) og angir en kontekst for navnet. Det er de som eier domenet som kan bestemme hvilken betydning et navn som kvalifiseres ved hjelp av deres domene, skal ha. Et åpenbart eksempel er hvordan W3C definerer betydningen av nøkkelordene i RDF ved hjelp av sitt domene. Et nøkkelord i RDF er ”type” (for å angi datatype). For å vise at det er snakk om ”type” slik det er definert i RDF-standardene, kvalifiseres det ved navnerommet <http://www.w3.org/1999/02/22-rdf-syntax-ns>, slik at det fulle navnet RDF ’type’ er <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>.

I eksempelet over bruker vi to navnerom for å eksemplifisere hvordan det kan brukes, <http://sms.skatteetaten.no/freg> og <http://sms.norge.no/>⁷⁷. Det første er tenkt som navnerom for de opplysningene som finnes i folkeregisteret. Det vil si at når ”navn” er kvalifisert med dette navnerommet, så menes det navnet på en person slik det ligger i folkeregisterets felt for navn.⁷⁸ Det andre navnerommet er tiltenkt opplysninger fra S.M.S., som i dette tilfellet opplysningen om den konkrete brukeren av S.M.S., ’Peder’.

Nå som vi vet betydningen av URIene i eksempelet, kan vi presisere betydningen av dataene til å lyde som følger: ”S.M.S-brukeren ’Peder’ har i henhold til folkeregisteret barnet ’Synnove’ og navnet ”Peder Aas””

2) Hvordan forkorte URIene i RDF

Bruk av URIer gir en mekanisme for å kvalifisere navn, men det er samtidig ikke til å komme bort fra at URIer er lange og tungleste. I RDF⁷⁹ forkortes URIer ved at man innledningsvis angir et prefiks som kobles til en URI, og for resten av dokumentet er det tilstrekkelig å bruke prefikset (samt et kolon) for å kvalifisere navnet. En variant av dette er å angi en ”default”

⁷⁶ ”Namespace” er definert i som en mekanisme for å håndtere navnekollisjoner i XML-dokumenter generelt, som dermed også kan brukes i RDF. XML-namespaces er definert i ”Namespaces in XML 1.0 (Second Edition)”, se: <http://www.w3.org/TR/REC-xml-names/> [19/4-2007]. Dette innebærer også at det ikke er nødvendig å bruke RDF dersom målet kun er å håndtere navnekollisjoner ved utveksling av data. Kombinasjonen av XML, XML Schema og Namespaces er tilsammen tilstrekkelig for å sikre utveksling av data på en entydig måte mellom aktører. ”Core Components” fra UN/CEFACT, som bl.a. er en videreføring av arbeidet elektronisk fakturering i EDIFACT, nøyer seg med å bruke disse tre standardene for å oppnå tilstrekkelig grad av ”semantikk” ved utveksling av data mellom systemer. Brønnøysundsregistrenes ”Semantikkregisteret for elektronisk samhandling” (SERES) baserer seg også på denne måten å håndtere semantikk på. For mer informasjon om UN/CEFACT, se: <http://www.unece.org/cefact/> [19/4-2007]. For mer informasjon om SERES, se: <http://www.brreg.no/samordning/semantikk/> [29/4-2007]

⁷⁷ Navnerom som begynner med <http://sms...> er navnerom vi selv har definert for bruk kun i denne oppgaven. Vi kan ikke definere navnerom for domenet skatteetaten.no, ettersom vi ikke forvalter dette domenet. Vi bruker det likevel fordi det synliggjør hvordan opplysninger fra ulike etater kan brukes i S.M.S. Det er for øvrig god skikk med tanke på versjonshåndtering å gi navnerommet et tidsstempel jf. W3Cs bruk av år og måned i sine navnerom. Vi har latt det være for å øke lesbarheten.

⁷⁸ Riktignok har folkeregisteret tre felt for navn; fornavn, mellomnavn og etternavn. For en presentasjon folkeregisteret og feltene som inngår, se artikkelen ”Generelt om folkeregistrering” på Skatteetatens hjemmesider, <http://www.skatteetaten.no/Templates/Artikkel.aspx?id=6640&epslanguage=EN> [19/4-2007]

⁷⁹ Som i fotnoten om namespace over, dette er ikke en RDF-funksjonalitet, men del av XML-standardene.

URI som innebærer at alle navn som ikke inneholder en URI eller et prefiks, tilhører navnerommet som er angitt av denne ”default” URIen.

Navnerommet for RDF-standarden, som vi så på over, blir normalt angitt ved prefikset ”rdf:”. Dette gjør det mulig å skrive rdf:type istedenfor <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>. For de navnerom vi har angitt for våre eksempeldata, bruker vi ”skdf:” (som en forkortelse for ”Skattedirektoratet – folkeregisteret”) og ”sms:”.

3) Tekstlig representasjon av eksempeldataene

Eksempelet vårt viser RDF-data i en grafisk representasjon. RDF-data kan også representeres som tekst, f.eks. i et XML-format (RDF/XML)⁸⁰ eller i et betydelig mer lesevennlig men mindre kjent N3-format.⁸¹ Nedenfor er et eksempel på dataene i N3-formatet, inkludert tilordningen av prefiksene som beskrevet over:

```
@prefix skdf: <http://sms.skatteetaten.no/freg#> .
@prefix sms: <http://sms.norge.no/#> .

sms:peder
  skdf:navn "Peder Ås";
  skdf:barn sms:synnove .
```

Alle tripler består av de tre delene subjekt, predikat og objekt, og avsluttes med et punktum. URIs settes i vinkelparentes ('<' og '>'), tekststrenger (og tall) settes i doble hermetegn. Navn som er kvalifisert ved hjelp av prefiks skrives rett frem. Eksempelet viser også hvordan man kan ta en ”snarvei” når det er flere tripler om samme subjekt. Istedenfor å gjenta subjektet, skilles de kombinasjonene av predikat og objekt som gjelder for subjektet fortløpende med semikolon. Etter siste objekt, settes det punktum. Den samme mekanismen kan forøvrig brukes dersom både subjekt og predikat er likt.

3.2.3 RDF og opplysningstyper vs opplysninger

Schartums modell (se kapittel 3.1.2 over) forutsetter håndtering av *opplysningstyper* og *opplysninger*. Nedenfor ser vi nærmere på hvordan dette representeres i RDF.

Eksempelet over inneholder tre opplysninger og to *opplysningstyper*. De tre konkrete opplysningene var ’Peder’, ’Synnove’ og ’Peder Ås’. Opplysningstypene var ”barn” og ”navn”. Nedenfor har vi endret dette til en tabell som illustrerer en vanlig relasjonsdatabase:

(Id) ⁸²	skdf:navn	skdf:barn
sms:peder	”Peder Aas”	sms:synnove

Dette tydeliggjør at ”navn” og ”barn” er opplysningstyper. Modellert for eksempel i ER⁸³-ville det vært naturlig å omtale sms:peder som en person, som har et attributt (skdf:navn) og

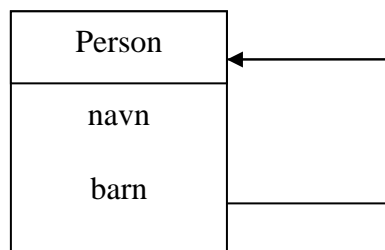
⁸⁰ Se ”RDF/XML Syntax Specification”, <http://www.w3.org/TR/rdf-syntax-grammar/> [30/4-2007]

⁸¹ Se Notation3 (N3) A readable RDF syntax, <http://www.w3.org/DesignIssues/Notation3> [30/4-2007]

⁸² Dersom disse dataene skulle vært representert i en tabell i en relasjonsdatabase, ville subjektet tilsvart primærnøkkel i tabellen. I dette eksempelet ser vi også at verdien i feltet ”barn” er en id som selv kan være primærnøkkel i en annen tabell (eller samme tabell), med andre ord en relasjon.

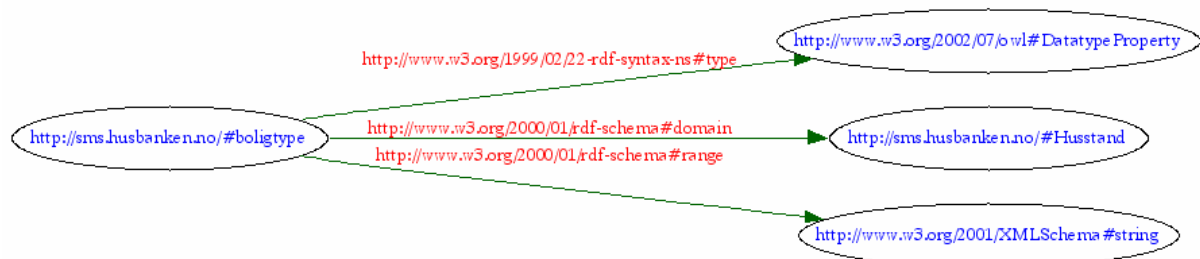
⁸³ ER står for Entity Relationship, for mer informasjon se: http://en.wikipedia.org/wiki/Entity-relationship_model [30/4-2007]

en relasjon (skdf:barn) som relaterer en Person til en annen Person. Dette er illustrert med et UML-skjema nedenfor:⁸⁴



Modellering av data, for eksempel ved bruk av UML-klassediagram, er en viktig del av arbeidet med å håndtere data i informasjonssystemer. I denne modelleringen vurderer man hva som er de sentrale opplysningstypene, og setter disse til å være entitene eller klassene. Andre opplysningstyper kan så være egenskaper eller attributter til disse igjen. Videre spesifiserer man eventuelle relasjoner mellom de sentrale opplysningene.⁸⁵

RDF gir også mulighet for å modellere dataene gjennom å spesifisere hvilke opplysninger vi skal håndtere, hvordan de henger sammen, og hvilke verdier de kan ha. Nedenfor er et eksempel på modellering av opplysningstypen ”boligtype” hentet fra regelverket for bostøtte fra Husbanken.



Figur 3 -- RDF-data som viser opplysningstyper

I eksempelet over har vi sagt at boligtype er en egenskap eller et attributt, og ikke en egen klasse, ved at vi har gitt den typen (rdf:type) ”DatatypeProperty”. Den er et attributt til klassen ”Husstand”,⁸⁶ og den kan ha datatypen ”string”, det vil si en vanlig tekststreng.

Vi har sett at det finnes et sett av nøkkelord til bruk for modellering av data i RDF, som bl.a. rdf:type, rdf:domain og rdf:range. Men RDF er en grunnsten i en stabel av standarder som tilsammen utgjør Semantic Web fra W3C, og over RDF finner vi blant annet RDF Schema og OWL. Begge deler utvider uttrykksmulighetene i datamodellen. I eksempelet over har vi allerede brukt et nøkkelord fra OWL, nemlig ”DatatypeProperty”, som selv igjen er en klasse.⁸⁷ Dessuten har vi sagt at ”boligtype” er et attributt til klassen ”Husstand”.⁸⁸ Det er

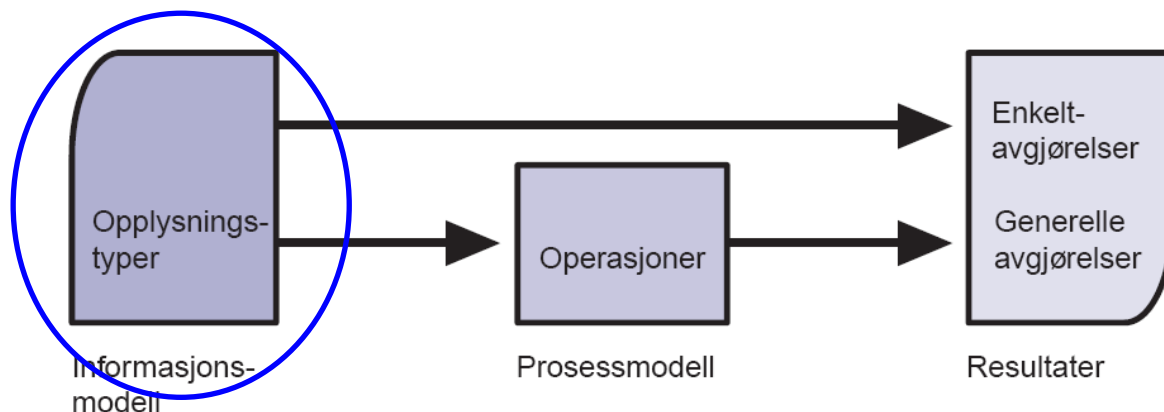
⁸⁴ UML står for Unified Modeling Language, se <http://www.uml.org/> [19/4-2007]

⁸⁵ Resultatet av modelleringen er en *datamodell*. Det er forvirrende at vi kan bruke RDF – som selv er en datamodell – til å uttrykke en annen datamodell. RDFs datamodell er en ”metadatamodell”, som kan brukes for å uttrykke andre datamodeller.

⁸⁶ Det er en konvensjon at klassenavn skrives med ”CamelCase”, dvs store forbokstaver.

⁸⁷ Det er også mulig å bruke nøkkelord fra den teknologien RDF bygger videre på, nemlig XML og XML Schema. Blant annet kan man gi nøyaktig spesifisering av datatypene (lovlige verdier) ved hjelp av XML Schema.

med andre ord sannsynlig at RDF-datamodellen gir oss muligheter til å etablere en rettslig informasjonsmodell i henhold til Schartums modell.



Figur 4 Sirkelen viser RDF-datamodellens anvendelsesområde i Schartums modell

3.2.4 En web av opplysningstyper – hva så?

Vi har sett at Semantic Web har en datamodell, RDF, som gir en svært fleksibel måte å håndtere data på, inkludert muligheten til å angi opplysningstyper. I motsetning til en relasjons- eller objektdatabase på en datamaskin gir den mulighet til å lage en samlet datamodell for data som finnes i forskjellige systemer. Nedenfor ser vi på hva dette betyr, og går videre til å se hvilke andre funksjoner som ligger i Semantic Web.

Når vi har dataene i en datamodell kan de trygt være input til alle Semantic Web-programmer, fordi datamodellen er avklart på forhånd. Dette betyr *ikke* at et Semantic Web-program vil ha kunnskap om – det vil si kunne operere på – alle data bare de er tilgjengelige som RDF-data. En kalenderapplikasjon vil for eksempel fortsatt bare gjøre fornuftig bruk av data som gjelder opplysningstyper som tidspunkter og steder og beskrivelser, men neppe data om opplysningstypen ”injeksjonsmengder” for medisiner. Men det viktigste her er at programmet vil ha nok informasjon til å kunne identifisere de data den kan gjøre noe med, og *la være* å foreta operasjoner på data den ikke har forutsetninger for å gjøre noe med.

Ved hjelp av RDF gir Semantic Web et verktøy for å håndtere dataenes betydning (semantikk) på tvers av systemer. Dermed kan data fra mange kilder danne et felles datagrunnlag, og gi resultater for eksempel ved spørringer, som ikke finnes i et enkeltsystem. Som et eksempel kan vi tenke oss at opplysningen om Peder og Martes nyfødte barn, og at de bor på samme adresse (fra Folkeregisteret) kombineres med opplysninger om deres inntekt (fra Skatteetaten) og til sammen gir informasjon som er nødvendig for å fastslå at de har rett til bostøtte og rentefritak. En slik spørring ville etter vår definisjon representere *kunnskap*. Semantic Web har også mekanismer for å gjøre denne typen kunnskap tilgjengelig for andre systemer.

I Semantic Web er RDF-datamodellen imidlertid bare en grunnsten for en ”stack” (stabel) av teknologier. Høyere opp i denne stabelen finner vi standarder som gjør det mulig å foreta og styre *operasjoner* på RDF-data. For eksempel finnes det standarder for å *spørre* etter RDF-data (SPARQL), for å utvide data i en RDF-datamodell slik at de kan inngå i en *ontologi*

⁸⁸ Konstruksjonen for klasse er `rdfs:Class`. Denne kan eventuelt tilordnes indirekte gjennom `rdfs:subClassOf` (ettersom ingenting kan være en subklasse av en klasse uten selv å være en klasse).

(OWL), og formulere *regler* som skal anvendes på dataene (RIF⁸⁹). I tillegg til dette er det også foreslått bruk av signaturer og kryptering for å kunne verifisere opphav og ”ubesudlehet” til data og dermed sikre at man kan ha tillit til dem. I kapittel 3.3 nedenfor går vi mer i detalj på disse teknologiene og ser om de tilbyr de funksjonene som er nevnt som aktuelle operasjoner i ”prosessmodellen” i Schartum (2005:74), aritmetiske operasjoner, logiske operasjoner og operasjoner for å hente ut data fra tilgjengelige kilder.

Dersom de enkelte systemenes kunnskap, i form av instruksjoner, ble formulert ved hjelp av teknologiene for spørringer, ontologier og regler i Semantic Web, ville det være mulig å *aggregere* denne kunnskapen. Det vil i så fall gi ytterligere muligheter for å få ut ”svar” fra Semantic Web som er resultat av kunnskap som ikke er bygget inn i noe enkeltssystem. Kan denne kunnskapen i så fall brukes til å veilede innbyggerne?

3.2.5 Semantic Web-basert ”kunnskap” for veiledning av innbyggerne?

Innbyggeren har forhåpentligvis dataene⁹⁰ om seg selv, og forvaltningen har mange av de samme dataene – men spredd i mange registre i mange organisatoriske enheter. Peder og Marte har likevel ikke informasjon om alle rettighetene sine. I vårt tilfelle fordi informasjonen om bostøtte og rentefritak ikke har nådd frem til dem. Selv om den nådde frem, er det imidlertid ikke sikkert at den ville blitt forstått, i den forstand at Peder og Marte hadde blitt klar over at nettopp *de* kunne få henholdsvis bostøtte og rentefritak. Det siste kan man kanskje se på som at de mangler *kunnskap* for å gjøre koblingene mellom opplysningene om seg selv og reglene for rentefritak.⁹¹ Vi kan anta at de dermed heller ikke har nødvendig kunnskap for å gjøre bruk av regelverket, i form av å gjøre krav på sine rettigheter.

En opplagt løsning på dette problemet er at forvaltningen bidrar med sin kunnskap. Når en nybakt mor mottar brev fire uker etter fødselen med varsel om vedtak om tildeling av barnetrygd, uten selv å ha løftet en finger,⁹² har forvaltningen selv brukt sin kunnskap om reglene på informasjon mottatt om fødselen. Men denne modellen forutsetter at alle forvaltningsorganene får informasjon om Peder og Marte. Husbanken har kunnskapene om reglene om bostøtte, men har ikke nødvendigvis kunnskap om Peder og Martes inntekt.⁹³ Det er også et ressurs spørsmål i hvor stor grad forvaltningen skal ”forfølge” innbyggere med informasjon.

Hva så med Semantic Web? Kan Semantic Web fôres med data om personer, og data om reglene, på en måte som gjør den i stand til anvende reglene på personopplysningene, og på den måten gi Peder og Marte veiledning? Og da veiledning i den forstand at innbyggeren blir gjort oppmerksom på regelverket (”kommer frem til”), og er i stand til å ”kunne forstå” og dermed handle ut fra det? Det er dette vi ønsker å teste gjennom løsningsforslaget til ”Semantisk Minside”. Som en del av arbeidet trenger vi å se nærmere på de enkelte delene av stabelen av Semantic Web-standarder.

⁸⁹ RIF står for ”Rules Interchange Format” og er ikke standardisert ennå. For informasjon om W3Cs arbeid med regler, se <http://www.w3.org/2005/rules/> [18/4.2007]

⁹⁰ Det har vist seg at brukere ved førstegangsinnlogging på Minside har blitt overrasket. Blant annet har brukere opplevd å få se eiendommer registrert i eiendomsregisteret som de hadde glemte at de eide.

⁹¹ Isåfall er det kanskje ikke i tråd med definisjonen av ”informasjon”, som innebærer at data skal ha blitt forstått. Men konklusjonen blir uansett den samme; de trenger hjelp fra andre – i form av andres kunnskap – for å bli klar over og gjøre koblingen mellom dataene om seg selv og dataene om rettighetene.

⁹² Dette er selvsagt ikke ment som en nedvurdering av innsatsen på fødestuen.

⁹³ Husbanken har rett til ligningsinformasjonen om inntekten til innbyggere, jfr [REF §§], men ikke slik at de kan gå ”aktivt inn” og søke etter mulige mottakere. Det brukes derimot til etterhåndskontroll av oppgitte opplysninger.

3.3 Nærmere om de forskjellige standardene for Semantic Web (Semantic Web-stacken)

I det forrige kapitlet har vi sett spesielt på RDF, men vi har også nevnt noen av de øvrige W3C-standardene som inngår i W3Cs Semantic Web, som for eksempel OWL og SPARQL. I dette kapitlet går vi igjennom disse og de øvrige standardene som er en del av Semantic Web.

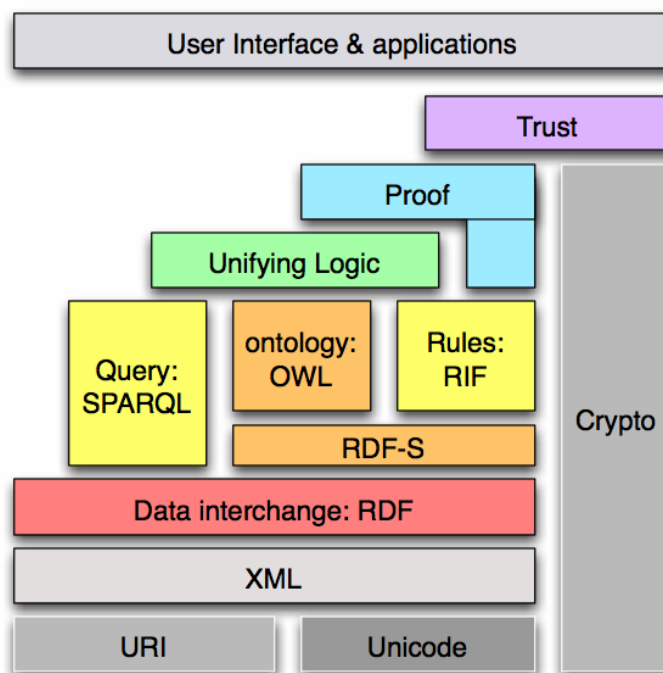
Semantic Web-arbeidet kom som et resultat av forsøk på å se for seg "the future web". Et tidlig forsøk på å beskrive "Semantic Web" er gitt i "Semantic Web Roadmap"⁹⁴ fra 1998, der Semantic Web oppsummeres som "web of data", og der følgende "byggesteiner" som må på plass for å realisere visjonen nevnes: 1) en datamodell (rdf), 2) et skjemalag (for opplysningstypene), 3) en mulighet for å konvertere data i henhold ett "skjema" til et annet, 4) et logikk-lag (slutningsmekanismer), 5) "proof-validation" for å kunne etterprøve påstander fra Semantic Web, 6) regelspråk, 7) spørrespråk og 8) elektroniske signaturer (for tillit).

Senere er ideen utdypet blant annet i en artikkel i Scientific American i 2001⁹⁵, og sist i "Semantic Web Revisited"⁹⁶ – som ser på fremveksten av Semantic Web. En gjenganger er presentasjonen av alle delene som en "stack" (stabel) av teknologi som til sammen utgjør Semantic Web. Kjennetegnet ved en slik stabel er at teknologier på et høyere nivå forutsetter teknologiene på nivåene under. Derimot er det ingen avhengigheter motsatt vei. Figuren som viser denne teknologistabelen har endret seg noe underveis, men hovedtrekkene er den samme som "Roadmap"-artikkelen fra 1998 som vi oppsummerte over.

⁹⁴ Dette dokumentet, som er skrevet av Tim Berners-Lee, er et utkast, men ligger tilgjengelig som en del av et sett notat om "DesignIssues". DesignIssues finnes her: <http://www.w3.org/DesignIssues/> Semantic Web Roadmap finnes her: <http://www.w3.org/DesignIssues/Semantic.html> [18/4-2007]

⁹⁵ Berners-Lee, Tim, James Hendler og Ora Lassila: "Semantic Web", Scientific American, mai 2001, se: <http://www.sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21> [30/4-2007]

⁹⁶ Shadbolt, Nigel, Tim Berners-Lee og Wendy Hall: The Semantic Web Revisited, IEEE Intelligent Systems 21(3) sidene. 96-101, Mai/Juni 2006, se: http://eprints.ecs.soton.ac.uk/12614/01/Semantic_Web_Revisted.pdf [20/4-2007]



Figur 5 -- Semantic Web Architecture -- Tim Berners-Lee 2006 -- [http://www.w3.org/2006/Talks/0718-aaai-tbl/Overview.html#\(14\)](http://www.w3.org/2006/Talks/0718-aaai-tbl/Overview.html#(14))

Nedenfor går vi igjennom den siste versjonen av figuren vi har funnet som illustrerer ”stacken”, og beskriver kort hva de forskjellige delene skal bidra med, og status for arbeidet.⁹⁷ Selv om vi allerede har gått igjennom RDF grundig i det forrige kapittelet, tar vi også det med her, for å komplettere fremstillingen og sette det i sammenheng.

3.3.1 URI, Unicode og XML

URI, Unicode og XML utgjør et fundament for alle W3C-teknologiene, både Semantic Web, videreutvikling av HTML (til XHTML) og Web Services. Disse tre standardene betegner henholdsvis standarden for å identifisere ”ting” på nettet (URI) som vi har sett er en viktig brikke for å håndtere navnekollisjoner, en tegnsattstandard som er omfattende nok til å håndtere de fleste tegn i de fleste språk i verden (Unicode), og en standard syntaks for å lage ”markup”-språk, eXtensible Markup Language (XML).⁹⁸

Alle tre standardene er velprøvde og stabile, kanskje med unntak av Unicode, der det fortsatt kan være utfordringer med at forskjellige applikasjoner støtter standarden på ulike måter. URI er standardisert av IETF som RFC 3986 ”Uniform Resource Identifier (URI): Generic Syntax”.⁹⁹ Unicode er standardisert av Unicode Consortium og siste versjon er ”Unicode

⁹⁷ Vi fant frem til denne ved å gå igjennom Tim Berners-Lees presentasjoner på oversikten over taler på W3Cs nettsted, <http://www.w3.org/Talks>. Figuren er hentet fra en presentasjon holdt 18. juli 2006 på ”The Twenty-First National Conference on Artificial Intelligence”, kalt ”AI and Semantic Web”, se: <http://www.w3.org/2006/Talks/0718-aaai-tbl/Overview.html> [30/4-2007]. Se Passin (2004:14) for en alternativ, og noe forenklet, presentasjon av den samme stabelen.

⁹⁸ De samme teknologiene kan sies å være fundamentet for dagens World Wide Web også, selv om HTML ikke er i overensstemmelse med XML, og det ikke er krav om bruk av Unicode-tegnsettet. Men det er antagelig to av de mest irriterende forholdene for de som lager nett-tjenester idag, nemlig å måtte forholde seg til de litt ”slappe” reglene i HTML samt alle de forskjellige tegnsettene som er i bruk (ASCII, ISO-8859-1, Windows-1252).

⁹⁹ Se: <http://tools.ietf.org/html/rfc3986> [30/4-2007]

5.0.0”.¹⁰⁰ XML er en W3C-standard. Siste versjon er ”Extensible Markup Language (XML) 1.0 (Fourth Edition)” fra august 2006.¹⁰¹

3.3.2 Data interchange: RDF

RDF står for Resource Description Framework, noe som indikerer at det opprinnelig var fokus på å utstyre nettsider med metadata, som for eksempel informasjon om hvem som var ansvarlig for innholdet, og hvorvidt det var egnet for mindreårige eller ei.¹⁰² I dag omtales standarden ofte bare som RDF, og fokuset er på at det er en generell datamodell som alle data skal kunne overføres til, og på den måten være et universelt format for utveksling av data mellom systemer. Navnekonflikter håndteres ved at navn uttrykkes som URIs som etablerer navnerom. I kapittel 3.2 gikk vi gjennom RDF-modellen og hvordan den baserer seg på tripler.

Et viktig designvalg bak RDF-datamodellen er at den er laget med tanke på at det høyere opp i teknologi-stabelen kommer standarder for bruk av formell logikk, som RDF dermed må understøtte. RDF er derfor laget med logisk konsistens for øyet. Dette er en ting som skiller Semantic Web-teknologiene fra W3C fra teknologien Topic Maps – som også blir ansett for å være en teknologi som kan realisere mye av visjonen for Semantic Web, og som har fått relativt stor utbredelse i Norge.¹⁰³

Det er identifisert noen svakheter i RDF-modellen, blant annet at den kanskje er *for* restriktiv. For eksempel vurderes det om tekststrenger kan være subjekter (idag må et subjekt være en URI). En viktig utvidelse vil også være en funksjon for å kunne referere til en RDF-triplet. Hvis man idag ønsker å si noe om en triplet, må man si noe om en *kopi* av den originale tripleten; en triplet kan ikke identifiseres ved hjelp av en URI. Dette kan være en svakhet med tanke på å kunne ha maskinlesbare uttalelser om hvorvidt en konkret triplet er til å stole på eller ei.¹⁰⁴

Den største svakheten ved RDF er antagelig formatet for å beskrive RDF-data i XML. Dette omtaler W3C selv som et unødvendig tungvint og uforståelig format, som dessuten ble spesifisert så tidlig at det *avviker* fra XML-standardene – hvilket gjør det ekstra vanskelig å arbeide med.¹⁰⁵ Muligens har vanskelighetene med dette formatet bidratt til å forsinke bruken

¹⁰⁰ Se: <http://www.unicode.org/versions/Unicode5.0.0/> [30/4-2007]

¹⁰¹ Se: <http://www.w3.org/TR/xml/> [30/4-2007]

¹⁰² Se ”Platform for Internet Content Selection”, som var et forslag til beskrivesspråk som RDF opprinnelig ble basert på, ifølge Mozillas informasjonsside om RDF, se: <http://www.mozilla.org/rdf/doc/> [30/4-2007]

¹⁰³ I forbindelse med Norstellas frokostseminar om semantisk interoperabilitet 14. mars 2007 uttrykte Steve Pepper, som regnes som en av opphavsmennene bak TopicMaps, at svakheten ved TopicMaps er at det ikke bygger på formell logikk. Men samtidig, påpekte han, er dette TopicMaps styrke. Det gjør TopicMaps egnet for andre formål og på andre områder enn semantiske teknologier som bygger på formell logikk. TopicMaps er standardisert av ISO (ISO13250). Se Kirill Miazine, ”Rettskilder og hyperstrukturer – om alternative grensesnitt til rettslige informasjonssystemer”, spesialoppgave ved NRCCL, Universitetet i Oslo, 2006, for TopicMaps som alternativ til fritekstsøk som navigasjonsmekanisme i tradisjonelle rettslige informasjonssystem. Oppgaven er tilgjengelig via Digitale Utgivelser ved Universitetet i Oslo, se:

<http://wo.uio.no/as/WebObjects/theses.woa/wa/these?WORKID=38662> [17/4-2007]. Prosjekter i Norge som bruker Topic Maps er blant annet arvtakeren til Odin, nettstedet Regjeringen.no.

¹⁰⁴ Ivan Hermans presentasjon i Beijing 16/10-2006 for oversikt over mulige endringer i RDF, se:

[http://www.w3.org/2006/Talks/1016-Beijing-IH/Slides.html#\(30\)](http://www.w3.org/2006/Talks/1016-Beijing-IH/Slides.html#(30)) [17/4-2007]

¹⁰⁵ Se faktaboks om ”Resource Description Framework” i Shadbolt (2006): ”The [XML-syntax] is actually quite clumsy syntactically, and its lack of transparency and readability might have been a factor inhibiting rapid adoption of RDF. However, there are alternative forms that are easier to interpret; for example, see the N3 notation (www.w3.org/DesignIssues/Notation3.html)”. Se også Ivan Hermans presentasjon 23/2-2007 om RDFs

av RDFs datamodell, fordi det opprinnelig ikke var tydelig nok at det var datamodellen som var det sentrale. På grunn av ”hype” rundt XML generelt, ble det viktigere å lage løsninger som brukte XML enn å se på fordelene ved å flytte data over på RDF-formatet.

Den opprinnelige RDF-standarden bestod av to forslag, hvorav ett beskrev ”model and syntax”, mens det andre beskrev ”schema”, begge forslagene 1999. I dag er disse erstattet av en samling av seks spesifikasjoner inkludert en ”primer”¹⁰⁶ som gir en oversikt over RDF som er mindre teknisk enn den formelle syntaktiske spesifikasjonen, og en spesifikasjon av ”test cases”¹⁰⁷ for å validere implementasjoner.

3.3.3 RDF-S

RDF-S er et ”skjemaspørsmål” for RDF. Forholdet mellom RDF og RDF-S er derfor sammenlignbart med forholdet mellom XML og XML Schema.¹⁰⁸ XML Schema er en erstatning for ”Document Type Definition” (DTD). DTDer ble brukt for SGML-dokumenter, og det er også mulig å bruke det for XML-dokumenter. En DTD uttrykker hvilken type elementer som var tillatt i en spesiell dokumenttype, som for eksempel HTML. I DTDen for HTML går det frem at <h1> og <h2> er lovlige elementer, mens f.eks. <etellerannet> ikke er det. DTDen spesifiserte også lovlig struktur, f.eks. at et <h1>-element ikke kan inngå i et hvilket som helst annet element.

Skjemaspørsmålet for XML gir utvidet funksjonalitet i forhold til DTDene. I XML Schema kan man i tillegg til å definere hvilke elementer som er tillatt, også definere hva som er et lovlig sett av verdier for det elementet. For eksempel kan man definere at et element <alder> kun skal kunne inneholde positive heltall, fra 0 til 150.

Ettersom RDF og RDF-S bygger på XML og XML Schema, er alle funksjonene i XML Schema tilgjengelige når man skal lage et RDF-skjema. Med andre ord bruker man konstruksjonene i XML Schema for å lage regler om datatypene, som for eksempel eksempelet med alder over. Det RDF-S tilfører, er noen enkle konstruksjoner for å definere *opplysningstyper* og relasjonene mellom disse. I RDF-datamodellen alene er det ikke noe skille mellom den konkrete opplysningen og opplysningstypen. Det er opp til den som lager RDF-dataene å selv håndtere det. Ved bruk av RDF-S, derimot, får man tilgang til noen enkle konstruksjoner for å beskrive ”vokabularet” dataene skal passe inn i.

Blant disse mekanismene finner vi konstruksjonen ”rdfs:Class”. Ved hjelp av denne og attributtet ”rdfs:subClassOf” kan man lage objektorienterte modeller der man definerer ”Person” som en opplysningstype av typen rdfs:Class, mens ”Husstandsmedlem” kan være en opplysningstype som er ”rdfs:subClassOf” ”Person”. Egenskaper som gjelder for Person vil dermed ”arves” til ”Husstandsmedlem”.

RDF-S gir større uttrykkskraft til RDF-data, men det er lagt vekt på å bevare enkelhet på bekostning av uttrykkskraft. Et nivå opp, derimot, finner vi OWL, som utvider mulighetene i RDF-S ytterligere.

XML-format: ”Good to know it was created when XML was not yet final ... ”,

[http://www.w3.org/2007/Talks/0223-Bangalore-IH/Slides.html#\(49\)](http://www.w3.org/2007/Talks/0223-Bangalore-IH/Slides.html#(49)) [30/4-2007]

¹⁰⁶ ”RDF Primer”, W3C recommendation 10/2-2004, se: <http://www.w3.org/TR/rdf-primer/> [17/4-2007]

¹⁰⁷ ”RDF Test Cases”, W3C Recommendation, 10/2-2004, se: <http://www.w3.org/TR/rdf-testcases/> [17/4-2007]

¹⁰⁸ For oversikt over XML Schema standarden fra W3C, se: <http://www.w3.org/XML/Schema#dev> [30/4-2007]

RDF-S er spesifisert i en av de seks spesifikasjonene som tilsammen definerer RDF (se kapittel 3.3.2 over). Det fulle navnet er "RDF Vocabulary Description Language 1.0: RDF Schema"¹⁰⁹

3.3.4 OWL

OWL står, interessant nok, for "Web Ontology Language".¹¹⁰ OWL er en ytterligere utvidelse av RDF-S, som gjør det mulig å presisere nærmere hvordan opplysningstypene i datamodellen forholder seg til hverandre. Ordet "ontologi" betyr "læren om tingenes eksistens, vesen og egenskaper".¹¹¹ Et ontologi-språk som OWL inneholder konstruksjoner som gjør det mulig å uttrykke mye om "tingenes egenskaper".

Hvis vi ser på eksempelet fra RDF-S, kan vi gå videre og si at en annen subklasse av "Person" er "Foreldre". Denne klassen kan igjen deles inn i to, "Mor" og "Far". Med OWL kan vi i tillegg si at ingen kan være medlem av begge disse to subclassene samtidig; de må være "owl:differentFrom" hverandre. En annen konstruksjon i OWL er "inverseOf". For eksempel kan vi slå fast at hvis en "far" "hasChild" "child", og "hasChild" er en "inverseOf" "hasParent", så vil vi automatisk kunne fastslå at "child" "hasParent".

Stikkordet i det siste eksemplet er "automatisk". Desto mer man har detaljert en datamodell med konstruksjonene i OWL, desto flere opplysninger er det mulig å fastslå automatisk gjennom logiske slutningsmekanismer, eller "inference". OWL finnes i tre varianter, "Lite", "DL" og "Full", som hver representerer en avveining i forhold til enkelhet og formell logisk uttrykkskraft. Som "DL" antyder, er formålet her å ha uttrykkskraft tilsvarende "Description Logic".¹¹² Behovet for avveining mellom uttrykkskraft og enkelhet, kan tydeliggjøres gjennom OWL Full; det har størst uttrykkskraft, men det er samtidig fare for at en slutningsmekanisme (inference engine) som jobber på data uttrykt med OWL Full vil kunne ende i et evig resonnement, som aldri gir resultat. OWL Lite og OWL DL gir garanti for at dette ikke skal skje (men ingen garanti mot at det tar *lang* tid før resultatet kommer).

Et spesielt forhold er at OWL også har en konstruksjon for klasser og subclasser. For OWL Lite og OWL DL er owl:Class definert som en subklasse av rdfs:Class, fordi ikke alt som kan være en rdfs:Class kan være en owl:Class, dersom OWL Lite og OWL DL skal bevare sikkerheten for at en slutning alltid vil gi et resultat. I OWL Full derimot, har man ikke denne restriksjonen, og derfor er owl:Class ekvivalent med rdfs:Class i OWL Full.¹¹³

OWL er definert i et sett av seks spesifikasjoner.¹¹⁴

¹⁰⁹ W3C Recommendation 10/2-2004, se: <http://www.w3.org/TR/rdf-schema/> [17/4-2007]

¹¹⁰ Vi har skrevet følgende forklaringen på dette i wikien til Semantic Web Special Interest Group (SWSIG): "Grunnen til at Web Ontology Language forkortes OWL har visstnok noe med den kloke uglen (Owl) i Ole Brumm, som staver navnet sitt "Wol". Forkortelsen er også ment å være det eneste som er inkonsistent i en standard som ellers kun har logisk konsistens for øye." Se: <http://swsig.computas.no/index.php/SWBegreper> [17/4-2007] Koblingen til Ole Brumm fremgikk av David Norheims presentasjon på Norstellas frokostseminar for Semantisk interoperabilitet 14. mars.

¹¹¹ Dokumentasjonsprosjektet ved UiOs bokmålsordbok, <http://www.dokpro.uio.no/perl/ordboksoek/ordbok.cgi?OPP=ontologi&begge=S%F8k+i+begge+ordb%F8kene&ordbok=bokmaal&alfabet=n&renset=j> [30/4-2007]

¹¹² Se Wikipedia-artikkel om Description Logic for mer detaljer, http://en.wikipedia.org/wiki/Description_logic [17/4-2007]

¹¹³ Se "OWL Web Ontology Language Reference", pkt 3.1, <http://www.w3.org/TR/owl-ref/#ClassDescription>

¹¹⁴ Se oversikt over W3C-spesifikasjonene på: <http://www.w3.org/2004/OWL/#specs>. [17/4-2007]

3.3.5 SPARQL

SPARQL står for "SPARQL Query Language for RDF", et navn som avslører informatikerens forkjærlighet for rekursive forkortelser. SPARQL er ikke standard fra W3C ennå, men et forslag til standard (Working Draft). Hovedfunksjonaliteten regnes likevel for å være så stabil at det allerede finnes relativt mange implementasjoner av SPARQL.¹¹⁵

SPARQL er tenkt å være for Semantic Web det SQL (Structured Query Language) er for relasjonsdatabaser. SQL har trolig vært mye av grunnen til at så mange har plassert så mye data i relasjonsdatabaser; SQL har sørget for at de har hatt en standardisert, relativt enkel og fleksibel måte å få de dataene de trenger ut av databasene igjen. Ved hjelp av SPARQL er det håp om at flere vil ønske å gjøre sine data tilgjengelige som RDF-data, ettersom det åpner for en enkel måte å integrere data fra flere kilder på.¹¹⁶

Ettersom dataene i RDF er på trippel-formatet, er syntaksen for spørringer basert på det. Nedenfor er et eksempel på en SPARQL-spørring som spør etter personer med navn "Peder Ås" og ber om å få se feltene navn og barn:

```
SELECT *
WHERE { ?Person skdf:navn "Peder Aas";
        skdf:barn ?Barn }
```

Innflytelsen fra SQL er tydelig, bruken av nøkkelordene "SELECT" og "WHERE" er hentet derfra. Ved angivelse av hva det spørres etter, kjenner vi igjen N3-formatet. Det nye her er bruken av parametere som innledes med "?". Det som kommer bak blir navnet på kolonnen i resultattabellen. Spørsmålet kan dermed leses omtrent som følger: "List opp alle personer som har navn (slik det er å forstå i navnerommet til Skatteetatens folkeregister) "Peder Aas" og deres barn".

Ved å laste eksempeldataene om Peder fra kapittel 3.2.2 inn i en demonstrasjonstjeneste på nettet kalt "Pellet", og stille spørsmålet som angitt over, får vi følgende resultat:

Query Results (1 answers):

Person	Barn
sms:peder	sms:synnove

Med spørrespråket SPARQL har Semantic Web en mekanisme vi mener vil understøtte etableringen av den rettslige prosessmodellen som inngår i Schartums modell. I SPARQL vil det blant annet være mulig med vilkårsprøving. Dessuten vil SPARQL også kunne brukes til "innhenting av saksopplysninger"¹¹⁷ i tilfeller som Schartum beskriver, der selve lovregelen angir at gitte opplysninger fra et gitt register på et gitt tidspunkt skal inngå i beslutningen.

¹¹⁵ For et "øyeblikksbilde", se siden "SparqlImplementations" på European Semantic Web Wiki, som pr 17/4- nevner et drøyt snes, inkludert løsninger fra bl.a. IBM, se: <http://esw.w3.org/topic/SparqlImplementations> [17/4-2007]

¹¹⁶ W3C arbeider med en spesifisering, GRDDL, for å gjøre det enklere konvertere data til RDF, se: <http://www.w3.org/TR/grddl/> [17/4-2007]. Merk at det ikke nødvendigvis betyr at disse dataene blir åpent tilgjengelig på Internett – mange vil trolig bruke det som et "lim" i sine interne systemer, som f.eks. skissert i Ivan Hermans presentasjon i Beijing 16/10 2006, se foilen kalt "SPARQL as a unifying point?", [http://www.w3.org/2006/Talks/1016-Beijing-IH/Slides.html#\(22\)](http://www.w3.org/2006/Talks/1016-Beijing-IH/Slides.html#(22)) [17/4-2007]

¹¹⁷ Schartum (2005:76)

3.3.6 "Rules: RIF"

RIF står for "Rule Interchange Format" og er foreløpig bare på det stadium i standardiseringsprosessen til W3C at det er utarbeidet et forslag (Working Draft) til en kravspesifikasjon for et slikt språk.¹¹⁸ Formålet er å utarbeide en standard som tillater at regler uttrykt med ett regelspråk kan transformeres til et annet regelspråk, og på den måten utveksles mellom regelsystemer.

Muligheten til å bruke automatisert slutning basert på ontologier (jfr OWL) og regler er en vesentlig del av potensialet i Semantic Web. For å praktisk kunne arbeide med regler har vi valgt å bruke det regelformatet som støttes i det verktøyet vi har jobbet med, CWM, som betegnes som "a general-purpose data processor for the semantic web".¹¹⁹ I dette avsnittet har vi derfor valgt å presentere og illustrere ved hjelp CWM, og på den måten få frem mulighetene i denne delen av Semantic Web-stabelen.

3.3.6.1 CWM

CWM er et lite, kommandolinjebasert verktøy programmert i Python, som i tillegg til å kunne lese inn og "spytt ut" RDF-data representert som tekst (både RDF/XML og N3), blant annet tilbyr en slutningsmekanisme. Denne slutningsmekanismen kan brukes både for å trekke ut "implisitte" data (ved hjelp av datamodellen/ontologien, jfr OWL), foreta spørringer (ved hjelp av spørrespråket SPARQL), og behandle *regler*.

Reglene i CWM baserer seg på N3-formatet, og består av tripler. Det settes opp en eller flere tripler i [Hvis]-delen av regelen. Disse triplene er koblet sammen med et logisk OG, slik at *alle* triplene må gi resultat, for at Hvis-delen skal være sann. Tilsvarende setter man en eller flere tripler i [Så]-delen. Mellom de to settene med tripler skriver man "=>" som er en innebygget kortform for "implies" hentet fra navnerommet `<http://www.w3.org/2000/10/swap/log#>`.

Nedenfor er et eksempel på en regel som sier at alle personer som har samme folkeregistrerte adresse, er husstandsmedlemmer:¹²⁰

```
{ ?x a skdf:Person. ?x skdf:adresse "Konkretveien 12" } =>
{ ?x a husbank:Husstandsmedlem }.121
```

Et forsøk på å oversette dette til et noenlunde naturlig språk ville gi noe sånt som "Alle x som er av typen Person (slik person er definert i navnerommet til Skatteetatens folkeregister), og har adresse "Konkretveien 12" er Husstandsmedlem (slik Husstandsmedlem er definert i navnerommet til Husbanken).

Dersom vi har et sett av data som inneholder en eller flere personer (x) med adresse "Konkretveien 12", vil konsekvensen av å be en slutningsmekanisme utføre regelen over, være at datasettet blir utvidet slik at de aktuelle personene får "hektet på" en ny opplysning,

¹¹⁸ "RIF Use Cases and Requirements", forslag, 10. juli 2006, se: <http://www.w3.org/TR/rif-ucr/> [20/4-2006]

¹¹⁹ Informasjon om CWM inkludert nedlastbar versjon, finnes på <http://www.w3.org/2000/10/swap/doc/cwm> [20/4-2007] CWM er først og fremst laget for å kunne eksperimentere med Semantic Web-teknologiene. Det er skrevet i Python og er blant annet ikke designet for å håndtere store datamengder eller levere svar med høy ytelse.

¹²⁰ Her har vi for enkelhets skyld sett bort fra leiligheter som kan ha samme adresse. For å få det nøyaktig ville det vært nødvendig å ta med et eventuelt leilighetsnummer fra folkeregisteret.

¹²¹ I praksis er dette en tredelt struktur, [noe] "impliserer" [noe], og det avsluttende punktumet avslører at dette behandles på samme måte som rdf-tripler av CWM.

nemlig at de er husstandsmedlem. Nedenfor har vi illustrert dette for Peder ved å vise dataene om ham *før* vi kjører dem gjennom CWM med regelen over, og resultatet etterpå.

Originale data

sms:peder

a skdf:Person;

skdf:adresse "Konkretveien 12" .

Resultatdata

sms:peder

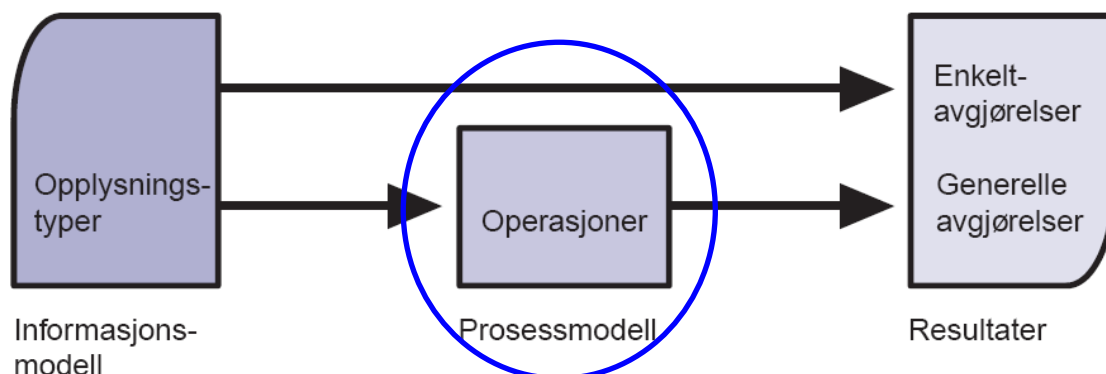
a husbank:Husstandsmedlem;

a skdf:Person;

skdf:adresse "Konkretveien 12".

CWMs regelimplementasjon tillater oss å bruke den logiske operatoren OG. Dette ser vi av regelen i form av de to triplene i første del, X må være en person OG ha en bestemt adresse. Realisering av den logiske operatoren ELLER løser man ved å ha et omkringliggende program med kontrollstruktur for ELLER, omtrent som angitt i følgende pseudokode: "Dersom kjøring av CWM for alternativ 1 gir negativt resultat, gå videre til alternativ 2" osv. CWM kan også teste på størrelser (større, mindre og er lik).

Basert på mulighetene ved bruk av regler og slutningsmekanismen i CWM, som gir et bilde på mulighetene til Semantic Web, mener vi at vi har et godt grunnlag for å konstatere at Semantic Web vil tilby regler og logikk som supplerer mulighetene i spørringen (jfr SPARQL over) og dermed gir oss det som er nødvendig for å etablere den rettslige prosessmodellen i henhold til Schartums modell, som vist nedenfor:



Figur 6 Sirkelen illustrer at SPARQL, regler og logikk understøtter prosessmodellen i Schartums modell

3.3.7 Unifying logic and proof

De forskjellige nivåene for å formulere ontologier og datamodeller (RDF, RDF-S og OWL) kan brukes på måter som gjør at en eller flere former for logikk ikke vil fungere med den måten man har beskrevet dataene på. Forskjellige slutningsmekanismer tilbyr forskjellige former for logikk, og for å være sikker på at den slutningsmekanismen man har valgt vil fungere, er man avhengig av å holde seg innenfor "evnene" til – det vil si beskrive sine data på en måte som ikke går utover det som kan forstås av – den slutningsmekanismen man har valgt.

Målet med "unifying logic" er å finne de subsettene av logikk som er tilstrekkelig kraftige og samtidige anvendelige i en web-sammenheng – det vil si en "åpen" verden der man for eksempel ikke kan garantere at det ikke finnes selvmotsigende regler, eller opplysninger som tilsier at X og Y er det samme, selv om en annen opplysning sier at X og Y må være forskjellige fra hverandre.

Vi bruker verktøyet CWM som har støtte for slutninger basert på ”første ordens predikatlogikk”¹²²

3.3.8 ”Crypto” og ”trust”

Kryptering og signering er mekanismer som kan bidra til å lage ”lukkete rom” på Internett. Innenfor disse kan vi med (stor grad av) sikkerhet si hvem som er opphavet til hvilke data og regler, og om de data en maskin mottar er de samme som en annen maskin sendte – eller om de er blitt tuklet med underveis. En reell tjeneste som gjør bruk av personopplysninger vil måtte ta stilling til hvordan disse skal sikres. Vi har valgt å avgrense oss fra detaljer rundt implementeringen av sikkerhet, og går derfor ikke nærmere inn på ”crypto” og ”trust”.

3.3.9 User Interface & applications

Hvis vi bruker bildet av Semantic Web som en stor, interoperatibel database, blir det tydelig at databasen i seg selv er til liten nytte dersom det ikke finnes programmer som gjør bruk av den. Data i en database er lite verdt dersom vi ikke har programmer som gjør det mulig for oss å lese dem eller endre dem. På samme måte må det etableres programmer og brukergrensesnitt som gjør bruk av de mulighetene som Semantic Web byr på.

CWM er et eksempel på en slik applikasjon. Vårt forslag til en ”Semantisk Minside” i kapittel 7 er et annet eksempel.

3.4 Oppsummering

I dette kapitlet har vi sett på Semantic Web fra W3C. Vi har lagt spesiell vekt på den viktigste byggestenen, datamodellen RDF. Underveis har vi sammenlignet Semantic Web-teknologiene med de elementene som fremgår av Schartums transformasjonsmetode, under henholdsvis den rettslige informasjonsmodellen og den rettslige prosessmodellen. Vi har konkludert med at Semantic Web-teknologiene både har støtte for å modellere opplysningstyper i tråd med informasjonsmodellen, samt formulere operasjonene i tråd med prosessmodellen. Dette gir oss grunn til å anta at det vil være mulig å håndtere et transformert regelverk i tråd med Schartums transformasjonsmetode, ved hjelp av Semantic Web-teknologi.

¹²² ”CWM is a popular Semantic Web program that can do the following tasks: [...] Perform inferences as a forward chaining FOPL [First Order Predicate Logic] inference engine”, Palmer, Sean B., se: <http://infomesh.net/2001/cwm/> [30/4-2007]

4 Behovsanalyse del 1 - offentlig informasjon

I innledningskapittelet så vi at Peder Ås og Marte Kirkerud overser en mulighet til å få flere tusen kroner i måneden i bostøtte, samt betalingsutsettelse med rentefritak fra Lånekassen, i forbindelse med at de er blitt foreldre. Årsaken er at informasjonen de finner om fødselspermisjon på NAVs nettsider, ikke sier noe om mulighetene hos henholdsvis Husbanken og Lånekassen. I dette kapittelet skal vi se nærmere på offentlig informasjon, nærmere bestemt informasjon om *rettigheter*. Aller først vil vi i kapittel 4.1 nedenfor se på bakgrunnen for etableringen av den statlige informasjonspolitikken. Deretter vil vi gi en overordnet presentasjon av innholdet i den statlige informasjonspolitikken, før vi analyserer innholdet i de fem prinsippene for forvaltningens informasjonsarbeid med tanke på å formulere *behov* til et informasjonssystem. Dette utgjør første del av vår *behovsanalyse*.

4.1 Tilgang til rettighetsinformasjon – et spørsmål om rettssikkerhet?

Et spørsmål vi kan stille oss innledningsvis, er hvorvidt det er et *rettslig* problem at Peder og Marte ikke er klar over sine rettigheter. Dette spørsmålet ville vært enklere dersom informasjonen om deres rettigheter *ikke fantes*, men lovhjemlene for rettighetene er kunngjort og tilgjengelige på Lovdata, og både Husbanken og Lånekassen informerer om ordningene på sine nettsider. Det kan med andre ord ikke være snakk om brudd på *publicatio legis*-prinsippet¹²³.

Krav til offentliggjøring (kunngjøring) av regler er et eksempel på et *rettssikkerhetskrav*. Nedenfor ser vi nærmere på rettssikkerhetsidealet, og hvilke krav det stiller, for å se om det finnes et eller flere rettssikkerhetskrav som kan være brutt i Peder og Martes tilfelle, ettersom de ikke er informert om viktige rettigheter.

Rettsstatsidealet tilsier blant annet at borgerne skal beskyttes mot overgrep og vilkårlighet fra myndighetenes side.¹²⁴ Dette idealet er utdypet gjennom formuleringen av rettssikkerhetskrav.¹²⁵ For å sikre oppfyllelsen av rettssikkerhetskravene, er det etablert en rekke virkemidler og tiltak. Disse betegnes som rettssikkerhetsgarantier. De generelle saksbehandlingsreglene for forvaltningen, gitt i forvaltningsloven, inneholder flere slike rettssikkerhetsgarantier. Noen av dem er knyttet til retten til *informasjon*. Dette gjelder blant annet partenes rett til *saksinnsyn* og forvaltningens *veiledningsplikt*.

Rettsstaten har utviklet seg til velferdsstat, og skal ikke lenger bare beskytte mot overgrep, men også sikre innbyggerne ytelser. Det er i dag aksept for at tildeling av velferdsstatens ytelser følgelig også er et område der rettssikkerhet må ivaretas. På samme måte som rettssikkerhetsidealet sørger for at vi etterstreber å forhindre overgrep, skal det sørge for at vi tilstreber at innbyggerne skal få de rettighetene de har krav på.

¹²³ Prinsippet om at lover skal kunngjøres, se bl.a. Grunnloven § 84, Lov 1969-06-19 nr 53 om Norsk Lovtidend m.v., Lov 1967-02-10 nr 00 om behandlingsmåten i forvaltningssaker (forvaltningsloven), § 37.

¹²⁴ En annen side ved rettssikkerheten er myndighetens plikt til å sørge for beskyttelse mot overgrep fra *andre* borgere.

¹²⁵ Torstein Eckhoff formulerte i artikkelen "Effektivitet og rettssikkerhet i den offentlige forvaltning" (Nordisk administrativt tidsskrift, 1958), fem rettssikkerhetskrav som ser ut til å være anerkjent som en relativt dekkende beskrivelse av kravene for å sikre innbyggerenes mot overgrep og vilkårlighet fra myndighetene. I tillegg til forutberegnlighet er det krav om "fair" og hensynsfull saksbehandling, upartisk- og uhildethet, likebehandling og til slutt demokratisk kontroll av forvaltningen. Vi har hentet oppstillingen av kravene fra Schartum (1993:30).

Fordi de ikke finner frem til den rette informasjonen er ikke Peder og Marte klar over to ordninger fra det offentlige som ville gjort det første halvåret som foreldre lettere økonomisk. Selv om dette ikke kan klassifiseres som et negativt *inngrep*, er konsekvensen ved å gå glipp av en ytelse man har krav på, langt på vei den samme som det å bli avtvunget f.eks. en uhjemlet avgift.

Et spesielt forhold er at myndighetens overgrep og vilkårlig maktutøvelse er noe som initieres av myndighetene selv, og de når dermed, per definisjon, *alle* innbyggerne som utsettes for overgrepet. Ved krav på ytelser, derimot, er det normale at innbyggeren selv må initiere en sak som velferdsstaten deretter reagerer på.¹²⁶

Et spørsmål blir dermed om krav til rettssikkerheten knyttet til tildeling av ytelser, er oppfylt så lenge en sak som er initiert blir behandlet i overensstemmelse med rettssikkerhetsidealet – eller om rettssikkerhetsidealet også inkluderer de som *ennå ikke er kjent med*, og dermed *ikke gjør krav på*, rettighetene sine? En rettssikkerhetsgaranti som er nærliggende å se på i den sammenheng, er den tidligere nevnte *veiledningsplikten* i forvaltningsloven.¹²⁷

4.1.1 Veiledningsplikten rekkevidde

Frihagen (1986) viser hvordan rekkevidden til veiledningsplikten i forvaltningslovens § 11 er forsøkt avgrenset i forarbeidene til loven.¹²⁸ For det første skal veiledningsplikten ikke dekke generell informasjon til allmennheten, men tar sikte på ”konkret veiledning til den enkelte”.¹²⁹ For det andre skulle veiledningsplikten gjelde overfor de som enten tok kontakt og ba om veiledning, eller som gjennom sin kontakt med forvaltningsorganet viste at det var påkrevd med veiledning.¹³⁰ Frihagen (1986) argumenterer med støtte i reelle hensyn, blant annet med henvisning til rettshjelpsundersøkelser og dokumentasjon på folks manglende kunnskap om sine rettigheter, for at veiledningsplikten ikke bør tolkes så snevert. Men eksemplene på hva han mener veiledningsplikten ”i hvert fall” også omfatter, sier ikke at den gjelder personer som *ikke* allerede har vært i kontakt med (et) forvaltningsorgan(et).¹³¹

Vi finner med andre ord ikke støtte i veiledningsplikten direkte for å si at Peder og Marte skulle ha blitt informert om rettighetene sine. Men Frihagen (1986) gjør en kobling mellom veiledningsplikten og begrepet ”aktiv informasjonsplikt” som muligens dekker deres

¹²⁶ Noen ytelser får man automatisk. Når et barn blir født, sender sykehuset melding om dette til Skatteetaten (folkeregisteret), som så gir melding til NAV. NAV fatter deretter et vedtak om tildeling av barnetrygd, og sender beskjed om dette til barnets mor, sammen med informasjon om klageadgang m.m.

¹²⁷ Forvaltningsloven, § 11.

¹²⁸ Frihagens kommentarutgave gjelder § 11 slik den lød i 1986, med et generelt prinsipp om forvaltningens veiledningsplikt (1. ledd) og en forskriftshjemmel (2. ledd). Frihagen ser også på forskriften som var gitt med hjemmel i § 11, 2. ledd, ”Forskrifter om veiledningsplikt av 16. desember 1977”, som inneholdt presiseringer av hvordan prinsippet i § 11, 1. ledd, skal praktiseres og tolkes. Siden er § 11 utvidet med hovedinnholdet fra forskriften, jfr Eckhoff (1997:392): ”Ved endringslov av 12. januar 1995 nr 4 er de viktigste av bestemmelsene i forskriften tatt inn i lovens § 11 (jfr. Ot.prp. nr. 75, 1993-94)”. Av den grunn er Frihagens argumentasjon med støtte i forskriften fortsatt relevant, jfr Eckhoff (1999:395): ”Fordi fvl. § 11 [...] bygger på de eldre forskriftene om veiledningsplikt, er også de utførlige kommentarene til disse forskriftene i *Frihagen* [...] fortsatt av interesse.”

¹²⁹ Frihagen (1986:202)

¹³⁰ Frihagen (1986:203)

¹³¹ ”Reelle hensyn taler i hvert fall for at veiledningsplikten også omfatter de tilfelle hvor vedkommende ikke selv makter å ta opp spørsmålet om hjelp eller åpenbart har misforstått systemet hvor dette f.eks. går fram av hans søknad.” Frihagen (1986:203)

situasjon. Frihagen trekker frem hvordan forarbeidene behandler spørsmålet om hvorvidt statsforvaltningen har en ”aktiv informasjonsplikt”. I forarbeidene fastslås det at en slik plikt ikke er nødvendig, og at den derfor ikke hører hjemme under forvaltningsloven, men isteden vil komme innunder Statens informasjonstjeneste, som den gang nylig var opprettet, etter anbefaling fra *Informasjonsutvalget*.¹³²

Hvis vi forfølger dette arbeidet videre, ser vi at det er en sammenheng mellom det nevnte Informasjonsutvalget, som leverte sin anbefaling i 1962, for siden å bli etterfulgt av et utvalg som la frem sin rapport i 1978, som igjen ble etterfulgt av Viksveen-utvalget i 1992. Det er det sistnevnte arbeidet som er utgangspunkt for dagens informasjonspolitikk, som inkluderer ”aktiv informasjon” som et prinsipp. Viksveen-utvalget anbefaler opprinnelig, i NOU 1992:21 ”Ikke bare ord... Statlig informasjon mot år 2000”, at det gis en generell *lovbestemmelse* som fastslår at staten *har* en ”aktiv informasjonsplikt”.¹³³ Forslaget til bestemmelse inneholder bl.a. krav til at informasjon skal være korrekt og ”gjøres lett tilgjengelig for aktuelle mottakere”. Hensikten med forslaget er blant annet å ”åpne for et kommunikasjonsperspektiv”. Denne lovbestemmelsen ble ikke innført, men forslag fra utvalget gikk inn i den statlige informasjonspolitikken, der det ble formulert fem prinsipper, blant annet prinsippet om aktiv informasjon og kommunikasjonsprinsippet.

Selv om det på denne måten kan etableres en kobling mellom veiledningsplikten og informasjonspolitikkenes prinsipper om ”aktiv informasjon” (og ”kommunikasjon”), gjenstår det både et spørsmål om informasjonspolitikkenes rettslige status, og det konkrete innholdet i prinsippene ”aktiv informasjon” og ”kommunikasjon”.

Schartum konkluderer med at ”[d]et eksisterer ingen lovbestemt plikt for statlig forvaltning til å drive aktiv informasjon overfor borgerne.”¹³⁴ Det er etter vår oppfatning lite sannsynlig at rettssikkerhetsidealet i dag går så langt at vi kan si at det er et brudd på rettssikkerheten at Marte og Peder ikke finner frem til bostøtte- og rentefritaksordningene. Slik vi tolker det er rettssikkerhetsgarantiene myntet på to ”typer” innbyggere. Den ene er den generelle innbyggeren. Dette er en innbygger som forvaltningen ikke vet hvem er. Her tilsier rettssikkerhetsgarantiene at det skal være mulig for innbyggeren å gjøre seg kjent med sine rettigheter. Dette kan antagelig kvalifiseres; det er ikke nødvendigvis tilstrekkelig at de *autentiske tekstene* (lov- og forskrifter) er tilgjengelige for innbyggeren. Forvaltningen har trolig et ansvar for å forsøke å formidle innholdet i regelverket på en mer lettfattelig måte. Dette kan til og med innebære at forvaltningen skal forsøke å inndele de generelle innbyggerne i kategorier, og forsøke å tilpasse informasjonen til forskjellige kategorier dersom det er nødvendig for å gjøre informasjonen lettfattelig.

Den andre typen er den konkrete innbyggeren. Dette er en innbygger som forvaltningen har en dialog med, gjennom at en sak er initiert. I dette tilfellet følger det av veiledningsplikten at det er utvilsomt at innbyggerne har krav på hjelp – selv om omfanget av hjelpen kan diskuteres – til å bli kjent med sine rettigheter.

¹³² Frihagen (1986:199), som siterer Ot.prp. 38 (1964-65) side 50.

¹³³ Forslaget til lovbestemmelse består av to ledd, hvor det andre gir Kongen myndighet til å fastsette nærmere regler. Første ledd lyder: ”Statlige forvaltningsorganer har innenfor sitt område aktiv informasjonsplikt. Informasjonen skal være korrekt, fullstendig, aktuell og gjøres lett tilgjengelig for aktuelle mottakere.”

¹³⁴ Udatert artikkel på AFINs nettsider, se:

http://www.afin.uio.no/forskning/andre_publicasjoner/sem260598/dws.html#ref3 [30/4-2007]

Etter vår mening kan det være fruktbart å snakke om en tredje type innbygger. Det er innbyggeren som ikke har henvendt seg til forvaltningen i en konkret sak, og dermed en innbygger som forvaltningen ikke vet hvem er. Men med dagens teknologi er det likevel mulig å *vite noe om* en person uten at vi vet hvem vedkommende er.¹³⁵ Dette er tydelig i Minside. Minside har tilgang til opplysninger om innbyggeren. Selv om innbyggeren ikke har henvendt seg til forvaltningen om en konkret sak bør det være mulig å gi innbyggeren veiledning på bakgrunn av hva f.eks. Minside vet om vedkommende. På den måten vil man kunne utvide omfanget av veiledningsplikten til også å innebære et krav om konkret veiledning til innbyggere som forvaltningen *vet noe om*.

Forvaltningslovens § 11, første ledd, andre punktum, legger opp til at veiledningen må ”tilpasses det enkelte forvaltningsorgans situasjon og kapasitet til å påta seg slik virksomhet.” Dette underbygger etter vår mening et argument om at veiledningsplikten kan utvides til å gjelde den tredje typen innbyggere, dersom det kan gjøres på en tilstrekkelig ressurseffektiv måte.

Historisk sett kunne en eventuell slik utvidelse av rettssikkerhetsbegrepet sammenlignes med utvidelsen som følge av overgangen fra rettsstat til velferdsstat. På vår vei over i *informasjonsamfunnet* kan man argumentere for at rettssikkerhetsbegrepet kanskje bør utvides til å omfatte mer ”aktiv” veiledning.

Selv om vi ikke kan fastslå at det er et rettssikkerhetsbrudd at Peder og Marte ikke blir gjort kjent med sine rettigheter til bostøtte og rentefritak, kan vi uansett fastslå at det innebærer et reelt problem, som rammer både innbyggerne som ikke får ytelse de er tiltenkt, og forvaltningen, som har etablert ordninger som ikke er kjent for målgruppen. Ut fra koblingen mellom rettssikkerhetsgarantien ”veiledningsplikt” i forvaltningsloven, og lovgivers kobling av dette til informasjonsutvalget og deres etterfølgere, mener vi at *i den grad det finnes* noe som kan kalles rettssikkerhetsgarantier med anvendelse i Peder og Martes tilfelle, er det sannsynlig at disse er formulert i den statlige informasjonspolitikken. Der fremstår de riktignok bare i form av *prinsipper*, og det er vanskelig å tenke seg at et organ eller en tjenestemann vil kunne holdes ansvarlig for brudd på disse prinsippene. Men vi velger å se på prinsippene i den statlige informasjonspolitikken som det nærmeste vi kommer formuleringen av rettssikkerhetsgarantier *forut* for en saksbehandling - for å sikre at innbyggerne er kjent med sine rettigheter. Dette gjør den statlige informasjonspolitikken velegnet til å undersøke problemstillingen vår; dersom Semantic Web-teknologi kan realisere funksjonalitet som understøtter den statlige informasjonspolitikken på området ”informasjon om innbyggerens rettigheter”, er det stor sannsynlighet for at løsningen vi skisserer gir innbyggerne bedre tilgang til rettighetsinformasjon.

¹³⁵ Liberty Alliance, som er et konsortium som standardiserer måter å bruke internett-teknologier for å håndtere *identitet* på Internett (Identity Web Services Framework – ID-WSF). Gjennom løsningen til Liberty Alliance er det mulig for en tjenesteleverandør på Internett og få en mulighet, via tiltrodde tredjeparter til å *garantere* at en person f.eks. er over 18 år – m.a.o. få vite noe om brukeren – uten at *identiteten* til vedkommende blir avslørt for tjenesteleverandøren – m.a.o. uten å vite *hvem* vedkommende er. Selvsagt er grensen mellom ”å vite noe om” en person og ”å vite hvem” personen er, flytende. I siste instans er det jo fordi vi vet noe om en person at vi kan si hvem det er. Men i praksis kan det være nyttig å endre fokus på ”personaliserte tjenester” fra å fokusere på identiteten, og bygge opp informasjon rundt identiteten til en bruker, til isteden å fokusere på de opplysningene man trenger om vedkommende for å kunne gjennomføre en tjeneste, som f.eks. veiledning. Mer om arbeidet til Liberty Alliance finnes på <http://www.projectliberty.org>. Se også Mahler, T. og Thomas Olsen ”Identity Management and Privacy Protection Law” for en vurdering av Libertys rammeverk i forhold til europeisk personvernlovgivning.

4.2 Informasjonspolitik for statsforvaltningen

Vi tar utgangspunkt i dokumentet som beskriver hvordan det offentlige informerer, den statlige informasjonspolitikken; ”Informasjonspolitik for statsforvaltningen”.¹³⁶

I tillegg til å redegjøre for innholdet i informasjonspolitikken, vil vi forsøke å fortolke de delene av den som angår rettighetsinformasjon. På den måten kommer vi frem til en konkretisering av hvilke *behov* det offentlige har til å informere i henhold til den statlige informasjonspolitikken. Når det er gjort vil vi i kapittel 5 gå igjennom de to løsningene vi har valgt å se nærmere på – LivsIT og Minside. For hver av disse løsningene vil vi se hvordan de svarer på behovene vi har utledet fra den statlige informasjonspolitikken.

Formålet med sammenligningen mellom nett-tjenestene og informasjonspolitikken er å etablere et tilstrekkelig detaljert bilde av nå-situasjonen for innbyggere på jakt etter rettighetsinformasjon på nettet, med tanke på *videreutvikling* av tilbudet. I den forstand tilsvarende gjennomgangen en typisk *forstudie* i en systemutviklingsprosess; før vi kan gå videre i utviklingsarbeidet trenger vi en idé om hva som allerede finnes.

Resultatet av vurderingen av LivsIT og Minside opp mot de behovene vi har utledet av informasjonspolitikken, blir et gap som beskriver behov som *ikke* er dekket. Disse gjenstående behovene vil i neste omgang brukes til å utlede konkrete *krav* som kan stilles til videreutvikling av dagens Minside til ”Semantisk Minside” (S.M.S.) i kapittel 6, Funksjonell kravspesifikasjon.

4.3 Den statlige informasjonspolitikken (SIP)¹³⁷

Den statlige informasjonspolitikken¹³⁸ av 1993 ble revidert i 2001, og utgjør en viktig ramme for utformingen av døgnåpen offentlig forvaltning og utvikling av informasjonsstrategier i offentlig sektor.

Informasjonsarbeid er en viktig del av oppgavene til offentlig sektor. Informasjonspolitikken omfatter både intern og ekstern informasjon så vel som informasjonsressursforvaltning (IRF). For brukeren av offentlig informasjon og tjenester vil det ikke være viktigst hvilke deler av offentlig sektor som er avsender, men om informasjonen man får er korrekt eller ikke.

Den statlige informasjonspolitikken består av *tre mål og fem prinsipper* for hvordan statlige virksomheter skal informere og kommunisere. Til sammen legger målene og prinsippene føringer for hvordan staten, på generelt grunnlag, bør løse sine informasjons- og kommunikasjonsvirksomheter.

Den beste begrunnelsen for at den statlige informasjonspolitikken er relevant i vår sammenheng gis i forordet til informasjonspolitikken. Der fremgår det at den statlige informasjonspolitikken ble revidert i 2001 nettopp fordi den skulle tilpasses en situasjon der ny teknologi har skapt nye informasjonskanaler:

¹³⁶ Selv om den statlige informasjonspolitikken ikke uten videre er gjeldende for kommunal sektor, velger vi å se på den statlige informasjonspolitikken som felles for statlig og kommunal sektor.

¹³⁷ I tillegg til primærkilden har vi også brukt artikkelen ”Offentlig informasjonspolitik” på Kommunikasjonsforeningens nettsted ”Syskrinet”, se: http://www.syskrinet.no/view_article.asp?id=140 [28/4-2007]

¹³⁸ Informasjonspolitik for statsforvaltningen, Arbeids- og administrasjonsdepartementet, 2001, se: http://www.regjeringen.no/en/dep/fad/Documents/rapporter_planer/Rapporter/2001/Informasjonspolitik-for-statsforvaltningen.html?id=438513 [28/4-2007]

”Målene og prinsippene informasjonspolitikken bygger på, er av allmenngyldig og tverrpolitisk karakter. Vi har derfor funnet liten grunn til å forandre dem. Men rammene rundt informasjons- og kommunikasjonsvirksomheten i staten har endret seg kraftig [...] *Ny teknologi*, nye medier og store endringer i sammensetningen av kommunikasjonskanaler stiller nye krav til hvordan arbeidet legges opp.” (Informasjonspolitikk for statsforvaltningen, forordet, vår utheving)

I avsnittet ”Informasjonstjenester på nettet” sies det i klartekst at SIP stiller krav til utformingen av offentlige nett-tjenester:

”På lik linje med annen informasjon må informasjonstjenestene på nettet være utledet av etatens mål og oppgaver og *utformet i henhold til målene og prinsippene i informasjonspolitikken*. De tekniske valgene som gjøres, må derfor fremme kommunikasjon med brukerne.” (side 11, vår utheving).

Selv om SIP i dette sitatet snakker om *informasjonstjenestene* på nettet, velger vi å tolke dette bredt, slik at det innbefatter *tjenester* på nettet også. I de påfølgende avsnittene til sitatet over gjøres koblingen mellom informasjonstjenester og andre tjenester. Det påpekes at bruken av Internett endres fra hovedsaklig informasjonsformidling til tilbud av tjenester, og avsnittet avsluttes som følger:

”Internett gir mulighet for en tettere sammenkobling mellom informasjon og tjenester enn de tradisjonelle kanalene.” (side 11)

SIP stiller krav til utviklingsprosjekter for offentlige nettsteder. Men det gjøres i form av relativt runde formuleringer og det finnes så vidt vi vet ingen ”sanksjonsmuligheter” dersom utviklingsarbeid foregår uten å ta hensyn til informasjonspolitikken. Dermed er det grunn til å anta at informasjonspolitikken blir vektlagt mindre i forbindelse med utviklingsarbeid enn andre, mer rettslig bindende, krav – som f.eks. lov om offentlige anskaffelser, arkivloven, forvaltningsloven og personopplysningsloven.

I tillegg til målene og prinsippene dekker SIP også en gjennomgang av en del konkrete problemstillinger, som spørsmålet om krisekommunikasjon, om økonomi og betaling for informasjon, og om forholdet mellom direktorat og departement i forhold til informasjonsarbeidet. Dette er problemstillinger som etter vår mening dekker konkrete forhold knyttet til det offentliges organisering av arbeidet med informasjon. Vi ønsker å se på hvilke krav som stilles til *innholdet* i informasjonen – og i den forbindelse er det målene og prinsippene som er relevante.

4.3.1 De tre målene

Befolkning, næringsliv og organisasjoner skal:

- ha lik og alminnelig tilgang til å delta aktivt i den demokratiske prosessen
- få informasjon om *sine* rettigheter, plikter og muligheter (vår utheving)
- ha reell tilgang til informasjon om det offentliges aktiviteter

Vår problemstilling er knyttet til det andre punktet. Selv om informasjon om *befolkningens* rettigheter, plikter og muligheter er tilgjengelig, er problemet at den ikke er enkelt nok tilgjengelig. Vi vil konsentrere oss om målsetningen om å gjøre informasjon om rettigheter tilgjengelig. I tillegg vil vi konsentrere oss om *befolkningen* som målgruppe, og da i egenskap av enkeltindivider som er på jakt etter informasjon om hva som gjelder for dem selv. Med andre ord har vi perspektivet til en person som er på jakt etter informasjon om *sine* rettigheter. Som begrunnelse for dette perspektivet låner vi en formulering fra SIP:

”Grunnleggende rettigheter som folk ikke kjenner til, kan de seg heller ikke benytte seg av.”¹³⁹

4.3.2 De fem kommunikasjonsprinsippene

I de fleste tilfeller er informasjon og toveis kommunikasjon en forutsetning for at offentlige oppgaver lar seg gjennomføre. De fem prinsippene for offentlig informasjonsvirksomhet er:

- 1) Kommunikasjonsprinsippet
- 2) Prinsippet om aktiv kommunikasjon
- 3) Helhetsprinsippet
- 4) Linjeprinsippet og
- 5) Prinsippet om informasjon om lederansvar.

Nedenfor tar vi en gjennomgang av hvert prinsipp. For hvert prinsipp gir vi først en oppsummering av innholdet i prinsippet, og deretter en vurdering av relevansen av prinsippet, gitt perspektivet beskrevet over.

For hvert prinsipp har vi til slutt utledet hvilke behov vi mener SIP har i forhold til en offentlig nett-tjeneste dersom den skal kunne oppfylle informasjonspolitikken. Disse behovene er utledet gjennom fortolkning av teksten som beskriver prinsippene i ”Informasjonspolitik for statsforvaltningen”. I noen tilfeller finner vi de samme behovene uttrykt under beskrivelsen av flere av prinsippene. Vi har valgt å bare ta med behovet første gang det opptrer. Et eksempel på dette er behovet for tilpasning av informasjon til brukerens konkrete situasjon, som vi finner under kommunikasjonsprinsippet, og behovet for målretting som vi finner under helhetsprinsippet. Vi mener dette i praksis innebærer det samme, og har derfor kun tatt det med som et behov under kommunikasjonsprinsippet (se B-2).¹⁴⁰

4.3.2.1 Kommunikasjonsprinsippet¹⁴¹

Forvaltning og brukere skal være likeverdige parter i en toveis kommunikasjonsprosess, og de skal veksle i rollene som avsendere og mottakere. Begge parter skal kunne ta initiativ og legge premisser for samhandlingen. Dette forutsetter kunnskap om mål- og mottakergruppers synspunkter, og kartlegging av disse gjennom bl.a. brukerundersøkelser. ”En forvaltning som kommuniserer godt, kjenner sine ulike brukergrupper og tar rede på deres ulike behov, ønsker og forutsetninger.”¹⁴² Kunnskapen om brukernes behov må ligge til grunn for både den langsiktige utformingen av tjenester og informasjonsvirksomhet og for det daglige arbeidet.

I vår sammenheng er kommunikasjonsprinsippet viktig ettersom det understreker at målgruppen ikke er én, homogen gruppe, men (grupper av) individer med forskjellige behov. Gjennom kommunikasjonsprinsippet påtar det offentlige seg en forpliktelse til å ta hensyn til disse individuelle forskjellene i sitt informasjonsarbeid.

Ut fra kommunikasjonsprinsippet, utleder vi følgende behov fra SIP:

B-1. Det bør være mulig å bruke tjenesten til *dialog* og ikke bare enveiskommunikasjon

¹³⁹ SIP, side 6 - skrivefeilen er ikke vår. Vi antar at det korrekte skulle vært: ”Grunnleggende rettigheter som folk ikke kjenner til, kan de heller ikke benytte seg av.”

¹⁴⁰ Behovene er nummerert fortløpende, og markert med ”B” for behov, for å kunne skille dem fra kravene vi skal utlede i kapittel 6 Funksjonell kravspesifikasjon.

¹⁴¹ SIP, side 7

¹⁴² SIP, side 7

- B-2. Det bør være mulig å tilpasse tjenesten til brukerens konkrete situasjon
- B-3. Det bør være mulig å la innbyggeren ta initiativet til kommunikasjon i tjenesten
- B-4. Det bør være mulig å la det offentlige ta initiativet til kommunikasjon i tjenesten

4.3.2.2 Prinsippet om aktiv informasjon¹⁴³

Innsyn i forvaltningen er ikke nok. Alle offentlige virksomheter har et ansvar for å gi befolkningen, næringslivet og de frivillige organisasjonene nødvendig informasjon om plikter, rettigheter og muligheter, noe som forutsetter en systematisk planlegging på grunnlag av kunnskap om ulike målgruppers situasjon og behov. Informasjon og tjenester skal være tilgjengelige når noen trenger det, og utformes for å tilfredsstillere disse behovene. Spesielt gjelder det de gruppene av befolkningen som har vanskeligst for å nå frem til det offentlige. Aktiv informasjon kan også bety at statlige virksomheter samarbeider med private organisasjoner og interessegrupper for å nå befolkningen indirekte gjennom dem. Aktiv informasjon innebærer ikke nødvendigvis mer informasjon, men det kreves at informasjonen er målrettet og dekker det faktiske informasjonsbehovet og er forståelig for brukerne.

Prinsippet om aktiv informasjon er helt sentralt for å oppfylle målsetning nummer to. Her går det tydeligere frem at det ikke er tilstrekkelig at informasjon om rettigheter og plikter er tilgjengelig, og kan fås f.eks. på forespørsel. Isteden har det offentlige et ansvar for å gi slik informasjon, og – i tråd med kommunikasjonsprinsippet – ta hensyn til ulike målgruppers situasjon og behov. Her trekkes det også frem at ulike grupper har ulike behov til ulik tid – informasjonen skal være tilgjengelig *når noen trenger den*. Avgrensningene er også viktige; mer informasjon er ikke nødvendigvis bedre – informasjonen må være dekkende og forståelig, og *målrettet*.

Vi utleder følgende behov fra prinsippet om aktiv informasjon:

- B-5. Det bør være mulig å senke kravet til egeninnsats for prioriterte grupper
- B-6. Det bør være mulig å få tilgang til ulike målgruppers situasjon og behov gjennom tjenesten
- B-7. Det bør være mulig å bruke tjenesten i samarbeide med private organisasjoner og interessegrupper
- B-8. Det bør være mulig å nå tjenester fra flere virksomheter gjennom én inngang (offentlig servicekontor)
- B-9. Det bør være mulig å bruke tjenesten hele døgnet (døgnåpen forvaltning)
- B-10. Det bør være mulig å dekke det faktiske informasjonsbehovet på en måte som er forståelig for målgruppen
- B-11. Det bør være mulig å se om informasjonen har blitt sett av mottakeren gjennom tjenesten

4.3.2.3 Helhetsprinsippet¹⁴⁴

Informasjon som gis fra en statlig virksomhet skal i størst mulig grad samordnes med informasjon og tjenester fra andre offentlige virksomheter, og overfor brukerne bør det legges opp til samspill og koordinering på tvers av forvaltningsnivåer og sektorer. Hvis ikke brukerne ser helheten i informasjonen som gis, vil det være vanskelig for dem å ivareta egne interesser på en god måte. Offentlig sektor bør derfor betraktes som en helhet, der man legger opp til samspill på tvers av forvaltningsnivåer og -sektorer. Helhetsprinsippet skal også sikres

¹⁴³ SIP, side 7 og 8

¹⁴⁴ SIP, side 8 og 9

gjennom internetportaler og felles grensesnitt som gjør det mulig å søke informasjon med utgangspunkt i den enkeltes situasjon og behov.

Helhetsprinsippet må sees i sammenheng med det neste prinsippet – linjeprinsippet, se nedenfor.

Fra helhetsprinsippet utleder vi følgende behov:

B-12. Det bør være mulig å bruke tjenesten uten å vite på forhånd hvilken virksomhet som står bak

B-13. Det bør være mulig å sette sammen tjenester fra flere virksomheter til en helhet

4.3.2.4 Linjeprinsippet¹⁴⁵

Informasjonsansvaret skal være knyttet til det organisasjonsleddet som har fag- og saksansvaret, og virksomhetene må sette seg inn i de informasjonsbehov og -utfordringer som gjelder for de enkelte saksområdene.

Som nevnt over må linjeprinsippet sees i sammenheng med helhetsprinsippet. Linjeprinsippet understreker at ansvaret for informasjonsarbeidet ligger hos den ”nærmeste” organisatoriske enheten. Dette gir en fare for at informasjon fra det offentlige fremstår som inkonsistent og usammenhengende. Dermed kommer helhetsprinsippet inn som en motvekt, for å sikre at det offentlige ikke ”taler med to tunger”. Helhetsprinsippet nevner spesielt *internetportaler* og *felles grensesnitt*, og legger dermed vekt på hvordan informasjonen fremstår utad.

Helhetsprinsippet og linjeprinsippet innebærer en vanskelig balansegang. Det organisasjonsleddet som har fag- og saksansvaret på et område vil antagelig også være den beste enheten til å gi inngående informasjon om området. Men på samme måte som organisasjonsleddets handlinger må koordineres med andre organisasjonsledd – slik at vi ikke får meningsløse handlinger der det offentlige ”tar med den ene hånda og gir med den andre” – må også informasjonen fra organisasjonsleddet passe inn i en kontekst – en kontekst som består av informasjon fra tilgrensende, over-, under og sideordnede områder.

Linjeprinsippet er i likhet med prinsippet om informasjon som lederansvar, et ”innoverrettet” prinsipp, og det er vanskelig å utlede behov som skal tilfredsstilles, men vi utleder dette som et minstemål:

B-14. Det bør være mulig å få vite hvem som er ansvarlig for tjenesten

4.3.2.5 Prinsippet om informasjon som lederansvar

Uansett hvilken formidlingskanal som benyttes, har ledelsen ansvaret for virksomhetens informasjon og at den er forankret i virksomhetens strategier og mål.

Prinsippet om informasjon som lederansvar er også et ”innoverrettet” prinsipp som stiller krav til den interne organiseringen når offentlige virksomheter jobber med informasjon. Prinsippet om informasjon som lederansvar skal sikre at informasjonsarbeidet får høy prioritet, på samme måte som i det foregående prinsippet – linjeprinsippet – ved at det skal være tydelig hvem som har ansvaret. Men det er vanskelig å utlede konkrete behov til hvordan informasjon

¹⁴⁵ SIP, side 9

til befolkningen skal utformes, basert på dette prinsippet, og etter vår mening er det bare ett behov som kan utledes, og det er det samme behovet som vi utledet for linjeprinsippet:

B-14. Det bør være mulig å få vite hvem som er ansvarlig for tjenesten

4.3.3 Oppsummering av behovene

Vi har dermed kommet frem til en rekke behov som den statlige informasjonspolitikken inneholder på området rettighetsinformasjon. Disse behovene oppsummerer første del av fasen ”behovsanalyse” i systemutviklingen vår. Oppsummert er dette behovene vi har kommet frem til:

Kommunikasjonsprinsippet:

B-1. Det bør være mulig å bruke tjenesten til *dialog* og ikke bare enveiskommunikasjon

B-2. Det bør være mulig å tilpasse tjenesten til brukerens konkrete situasjon

B-3. Det bør være mulig å la innbyggeren ta initiativet til kommunikasjon i tjenesten

B-4. Det bør være mulig å la det offentlige ta initiativet til kommunikasjon i tjenesten

Prinsippet om aktiv informasjon:

B-5. Det bør være mulig å senke kravet til egeninnsats for prioriterte grupper

B-6. Det bør være mulig å få tilgang til ulike målgruppers situasjon og behov gjennom tjenesten

B-7. Det bør være mulig å bruke tjenesten i samarbeide med private organisasjoner og interessegrupper

B-8. Det bør være mulig å nå tjenester fra flere virksomheter gjennom én inngang (offentlig servicekontor)

B-9. Det bør være mulig å bruke tjenesten hele døgnet (døgnåpen forvaltning)

B-10. Det bør være mulig å dekke det faktiske informasjonsbehovet på en måte som er forståelig for målgruppen

B-11. Det bør være mulig å se om informasjonen har blitt sett av mottakeren gjennom tjenesten

Helhetsprinsippet:

B-12. Det bør være mulig å bruke tjenesten uten å vite på forhånd hvilken virksomhet som står bak

B-13. Det bør være mulig å sette sammen tjenester fra flere virksomheter til en helhet

Linjeprinsippet og prinsippet om informasjon som et lederansvar:

B-14. Det bør være mulig å få vite hvem som er ansvarlig for tjenesten

Vi vil videre se hvilke av disse behovene som er dekket av LivsIT og Minside og på den måten komme frem til hvilke behov som stilles i SIP, men som ikke er dekket av LivsIT og Minside. Dette ”gapet” mellom hvilke behov som er dekket og de som er udekket vil vi senere omforme til krav (i den funksjonelle kravspesifikasjonen) som stilles til vår videreutvikling av Minside – den ”Semantiske Minside” (S.M.S.).

5 Behovsanalyse del 2 – LivsIT og Minside – to eksisterende offentlige informasjonstjenester

I dette kapittelet gir vi en presentasjon av to eksisterende offentlige informasjonstjenester, LivsIT og Minside. For hver av dem vurderer vi hvordan disse informasjonstjenestene tilfredsstillere behovene vi har utledet fra den statlige informasjonspolitikken i forrige kapittel. Resultatet blir en liste over behov som *ikke* er dekket. Dette gapet er resultatet av denne andre delen av vår behovsanalyse, og utgjør det som vi går videre med til kapittel 6, hvor vi lager en funksjonell kravspesifikasjon for S.M.S.

Ettersom vi skal ta utgangspunkt i dagens Minside når vi designer vårt forslag til forbedret løsning for rettighetsinformasjon, trenger vi mer inngående kjennskap til Minside enn LivsIT. Derfor vil Minside bli mer utførlig behandlet enn LivsIT i dette kapittelet. Men siden organiseringen av informasjon i Minside bygger på samme tankegang som LivsIT, er begge løsningene nyttige å ha med.

5.1 LivsIT

5.1.1 Idé og formål

LivsIT startet som et prosjekt i 1996,¹⁴⁶ men ideen bak kan spores tilbake til Viksveenutvalget (1992:121). Formålet med tjenesten var å samle offentlig informasjon på en mer hensiktsmessig og brukervennlig måte. Målsettingen for tjenesten var ambisiøs:

"Ville det ikke vært nyttig med en tjeneste som samlet all relevant informasjon om et emne på *ett* sted, uavhengig av hvilke kilder den kommer fra? Det er det LivsIT tar mål av seg til å tilby når det gjelder offentlig informasjon."¹⁴⁷

LivsIT er en forkortelse for livssituasjonsbasert IT-system.¹⁴⁸ Grunntanken bak er at man skal kunne finne all offentlig informasjon om et emne uten å på forhånd vite hvem som er den ansvarlige offentlige instansen. Navigasjonen er basert på at du skal navigere deg frem på bakgrunn av hvilken livssituasjon du befinner deg i. Skal du for eksempel bygge hus skal du via den kommunale hjemmesiden få opp den offentlige informasjonen som er relevant for deg som boligbygger, uavhengig om informasjonen er kommunal, fylkeskommunal eller statlig.

Årsaken til at de kommunale nettstedene ble valgt som "portal" for LivsIT var de kommunale servicekontorenes erfaring med at brukerne/innbyggerne ikke har kunnskap nok om offentlig sektor til å kontakte den korrekte offentlige instansen direkte. Innbyggerne vet ikke nok om, eller bryr seg ikke om, skillet mellom hva som er statlig, fylkeskommunalt eller kommunalt arbeidsområde, men kontakter det offentlig kontoret som er nærmest og mest kjent - det kommunale.¹⁴⁹

5.1.2 Funksjonalitet

Strukturen til LivsIT er basert på metadata. Metadata betegnes ofte som "data om data".

¹⁴⁶ LivsIT - elektronisk informasjon i livssituasjoner, prosjektrapport 1998 - Statens informasjonstjeneste

¹⁴⁷ Ølnes, Svein og Terje Aaberge: Håndbok for innføring av LivsIT" Vestlandsforskning VF-rapport nr. 4/2004

¹⁴⁸ Ølnes, Svein og Terje Aaberge: Enklare tilgang på offentlig informasjon - Evaluering av LivsIT"

Vestlandsforskning VF-rapport nr. 7/2003

¹⁴⁹Ølnes (2004)

Metadataene blir ikke satt inn i de ulike dokumentene som inneholder offentlig informasjon, men derimot plassert i en frittstående database. En lenke til originaldokumentet binder databasen og dokumentet sammen. Tanken bak LivsIT var at metadataene skulle legges inn av forfatteren av dokumentet uavhengig av om denne forfatteren var lokalisert på sentralt eller lokalt nivå innen offentlig sektor. Hvilke metadata forfatteren kan velge fra er allerede definert i LivsIT-taksonomien.¹⁵⁰

En annen hovedfunksjonalitet med LivsIT er at det har en teknisk løsning for å distribuere temastrukturen og de tilknyttede informasjonsressursene. På den måten kan kommunene på en enkel måte utvide sin informasjonstjeneste slik at offentlig sektor fremstår som enhetlig utad. Det at strukturen klargjøres sentralt bør også sees i et informasjonsressurs-forvaltningsperspektiv (IRF). Hver enkelt kommune slipper arbeidet med å samle ressursene hver for seg og kan heller bruke ressursene på den lokale informasjonen.

I praksis er det i dag Norge.no som kategoriserer den statlige og fylkeskommunale informasjonen, mens de ulike saksbehandlerne i kommunen kategoriserer den lokale informasjonen. Kategoriseringen foregår etter den fastlagte (LivsIT-) taksonomien. Resultatet av klassifiseringen er et sett metadata i en database sammen med en referanse (URI) til originaldokumentet. Dette lastes ned til kommunen, og kommunen presenterer dette sammen med sin egen informasjon på sitt lokale nettsted.

Resultatet for brukeren av den kommunale hjemmesiden er at han eller hun blir presentert for informasjonen uten å måtte vite om dette er statlig, fylkeskommunal eller kommunal informasjon – alt er offentlig informasjon.

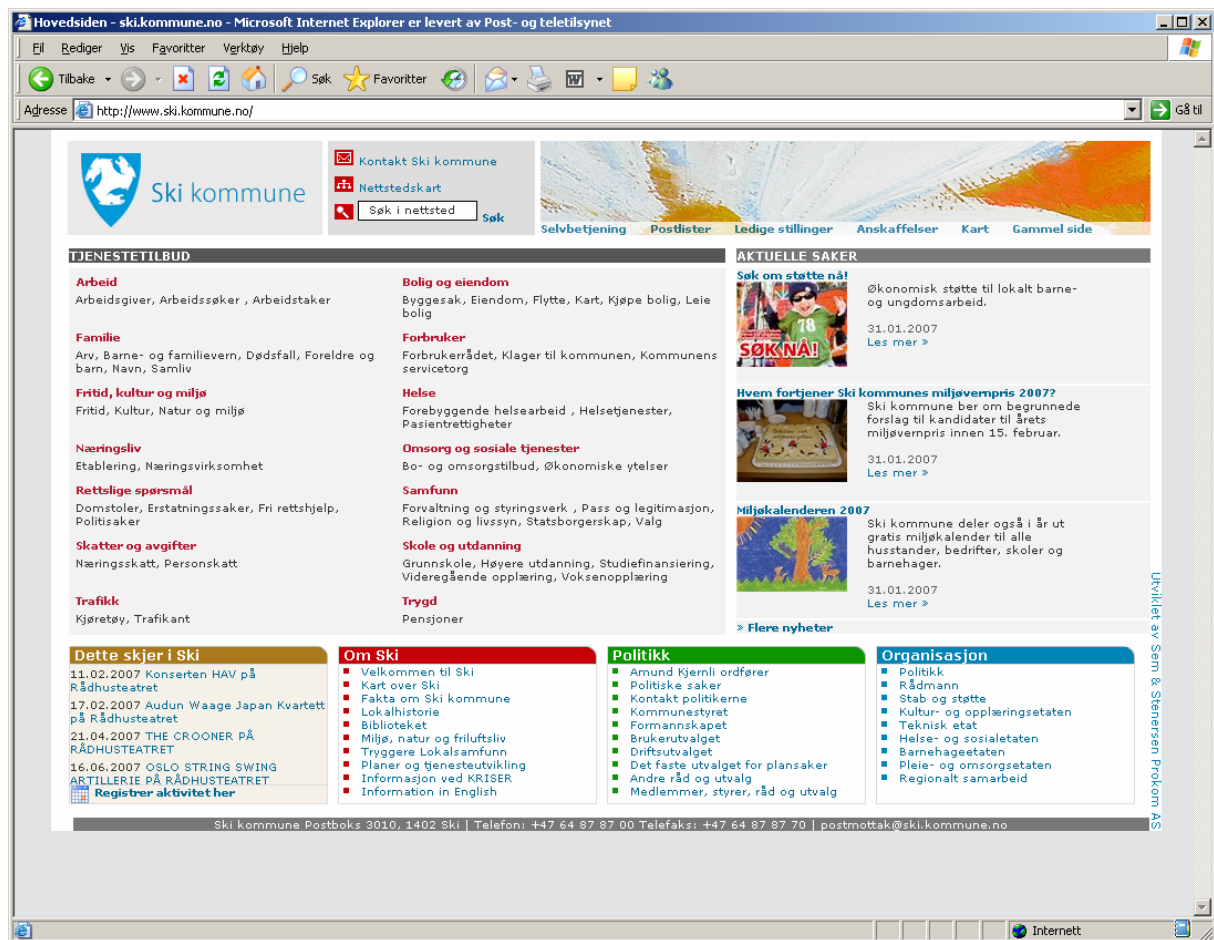
5.1.3 LivsIT og scenarioet

For Peder og Marte vil informasjonen fremstå som oversiktlig på kommunens nettsted.¹⁵¹ LivsIT-taksonomien vil fremstå som en rekke meny punkter og under disse meny punktene vil kommunen enten dele inn informasjonen som kommunal informasjon og annen offentlig informasjon eller de vil slå dette sammen som en samlet tjeneste.

¹⁵⁰ taksonomi=emne hierarki

¹⁵¹ Peder og Marte bor i Oslo kommune, og Oslo kommune er ikke en del av LivsIT, så vi bruker Ski kommune og Asker kommune som eksempler for å illustrere.

Et eksempel fra Ski kommunes hjemmeside¹⁵² viser at menypunktene¹⁵³ tilsvarer emnene i LivsIT-taksonomien:

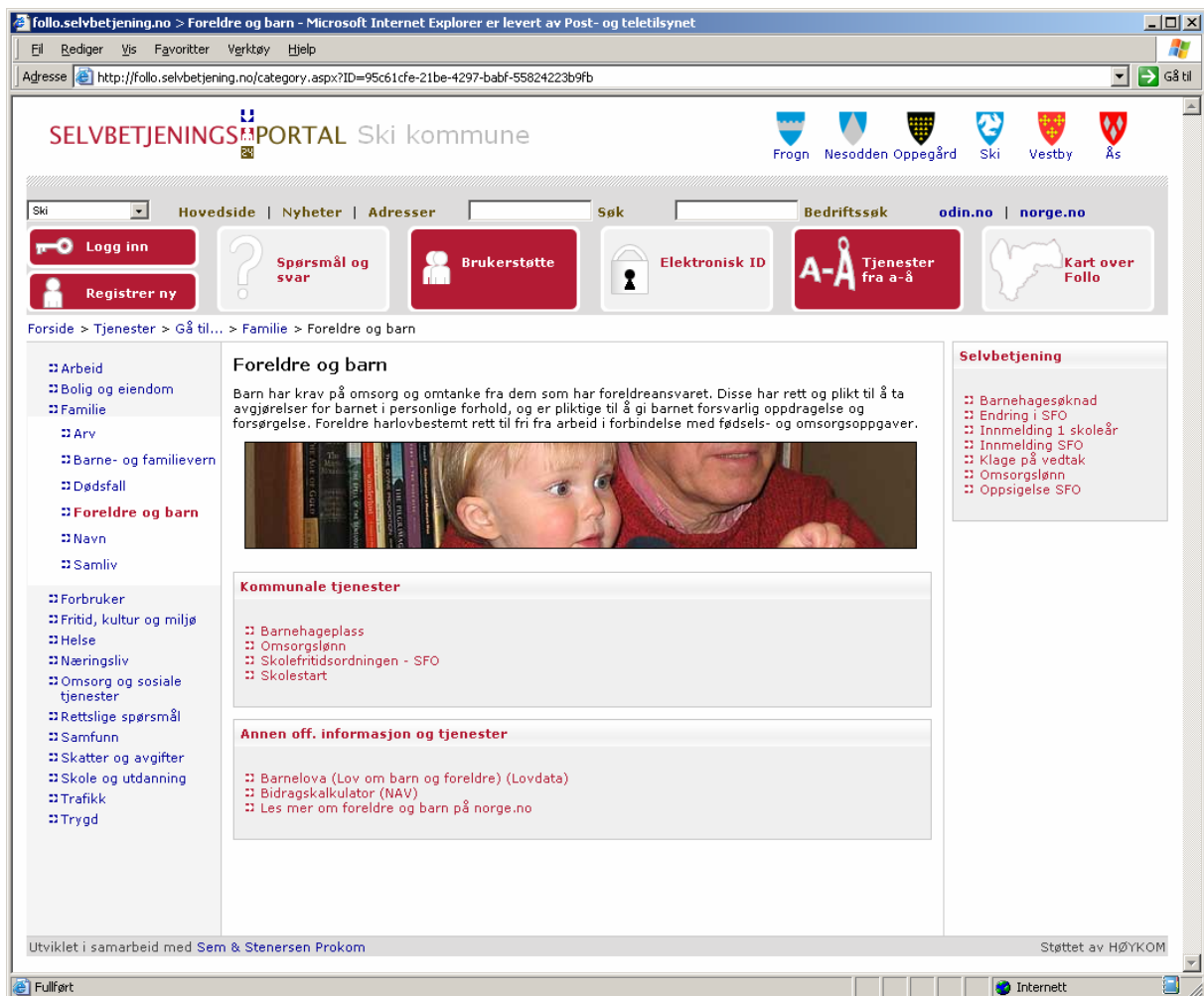


Figur 7 - fra Ski kommunes hjemmeside [6/2-2007]

¹⁵² <http://www.ski.kommune.no> [6/2-2007]

¹⁵³ "Arbeid", "Bolig og eiendom", "Familie", "Forbruker", "Fritid, kultur og miljø", "Helse", "Næringsliv", "Omsorg og sosiale tjenester", "Rettslige spørsmål", "Samfunn", "Skatter og avgifter", "Skole og utdanning", "Trafikk", "Trygd" er alle emner i LivsIT-taksonomien.

For Peder og Marte ville det naturlige valget i deres nye situasjon være å klikke på "foreldre og barn" under overskriften Familie. De vil da få opp dette skjermbildet:



Figur 8 – Skjermbilde etter å ha klikket på "Foreldre og barn" under overskriften "Familie" [6/2-2007]

Det kommer da opp fire valg under kommunale tjenester:

- Barnehageplass
- Omsorgslønn
- Skolefritidsordning - SFO
- Skolestart

Under "Annen off. informasjon og tjenester" får de opp lenker til disse dokumentene:

- Barnelova (Lov om barn og foreldre) (Lovdata)
- Bidragskalkulator (NAV)
- Les mer om foreldre og barn på norge.no

Verken ved å følge lenkene under "Kommunale tjenester" eller "Annen off. informasjon og tjenester", finner Peder og Marte noe om Husbankens støtteordninger. Disse ordningene ligger under en annen kategori i LivsIT-strukturen. De skulle ha gått inn under overskriften "Bolig og eiendom" og valgt "Leie bolig". Dette illustrerer et av problemene med LivsIT – ikke alle vet hvilken livssituasjon de er i! Og den livssituasjon man er i samsvarer ikke nødvendigvis med de livssituasjoner LivsIT baserer seg på. En annen svakhet med LivsIT er at man ofte er i flere livssituasjoner samtidig – for eksempel barnefamilie OG leietakere.

Et annet interessant poeng med dette eksempelet er hvor forskjellig ulike kommuner annonserer det samme tilbudet. Ser man på informasjonen om "Bostøtte" i LivsIT for kommunene Asker og Ski viser det seg at selv om innholdet i bostøtte-tjenesten fra Husbanken er likt i kommunene informeres det om tilbudet ulikt.

Asker¹⁵⁴:

Bostøtte

Bostøtte er en statlig, økonomisk støtteordning som administreres av Husbanken og kommunene.

Målgruppe

- Barnefamilier
- Eldre over 65 år
- Alderspensionister, uføre- og etterlattepensionister med svak økonomi.
- Personer med attføring, rehabiliteringspenger og grunn-/hjelpetønad

Målet med ordningen er å bidra til at eldre, uføre og barnefamilier med svak økonomi skal kunne anskaffe - eller bli boende i - en god og hensiktsmessig bolig. samt å utjevne inntektsforskjeller mellom pensjonistgrupper som følge av forskjeller i boutgiftene.

Hvordan gå frem?

Søknad sendes til Sosialtjenesten, sammen med:

- Bekreftelse fra banken på boliglån (hvis eid bolig)
- Kopi av leiekontrakt (hvis leid bolig)
- Kopi av lønsslipp for gjeldende termin, barnebidrag, studielån, stipend, sosial stønad og annet dersom man søker på dagens inntekt.

Leveringsfrister:

Leveres innen fristene (10. februar, 10. juni og 10. oktober) til sosialkontoret eller Servicetorget. Kommunen overfører dataene i søknaden til en sentral database. Der blir opplysningene bearbeidet frem til endelig vedtak. Utbetalingene er månedlig

Søknadsskjema og hjelp til utfylling fås på Servicetorget og sosialkontoret. Søknadsskjema fra Husbanken kan også lastes ned her. Mer informasjon om bostøtte finner du hos [Husbanken](#).

Saksgang

Bostøtten blir behovsprøvd mellom boutgiftene og husstandens samlede inntekt

Størrelsen av bostøtten beregnes utfra husstandens inntekt og godkjente boutgifter. Boligen skal være helårsbolig og ha eget kjøkken, bad, oppholdsrom, soverom og egen inngang, samt være over 40 kvm utenom fellesareal. Les om unntak på Husbankens side. En husstand er den eller de personer som faktisk bor i og samtidig er registrert bosatt i boligen, i henhold til Folkeregistret. Det er et krav at dette er oppfylt per situasjonsdato for hver termin:

- 1. januar for 1. termin
- 1. mai for 2. termin
- 1. september for 3. termin.

¹⁵⁴ <http://www.aker.kommune.no/Tjenester-til-deg/Bolig-og-eiendom/Ulike-tilskudd/Bostotte/> pr. 06.02.07

Utbetalingene er månedlig. Kommunen overfører dataene i søknaden til en sentral database. Der blir opplysningene bearbeidet frem til endelig vedtak. Deretter blir søker informert.

Oppdatert: 06.10.2006 13:17:13

Mens Ski kommune har en noe kortere forklaring på Bostøtten:

Ski kommune¹⁵⁵:

Bostøtte

Bostøtte er en statlig økonomisk støtteordning som administreres av Husbanken og kommunene. Bostøtte kan du søke om dersom husstanden din har lave inntekter og høye boutgifter.

Støtteberettigelse

For å få bostøtte må én i husstanden din være støtteberettiget. Støtteberettiget person er:

- barn under 18 år eller person over 65 år
- person med visse typer trygder/pensjoner/stønader fra det offentlige (alders- uføre- og etterlattepensjon, atføerings- og rehabiliteringspenger, grunn- og hjelpestønad etc.
- person med sosial stønad som eneste inntekt, med en varighet på minst ett år

Krav til boligen

For å få bostøtte må boligen din som regel være av en viss størrelse og funksjon. For noen typer boliger stilles det også krav til hvordan boligen er finansiert. For mer informasjon om dette, se kontaktinformasjon nedenfor.

[Mer informasjon om bostøtte finner du på Husbankens nettsider](#)

LivsIT har foretatt en evaluering og kommet frem til flere mulige endringer av tjenesten, men siden det per dags dato (mai 2007) ikke er implementert noen endringer har vi ikke fokusert på disse mulige endringene i oppgaven. Hovedpoengene i den nye strukturen er ikke av en vesentlig art i forhold til dagens teknologiske valg, men mer enn videreføring og forbedring av bruken.

5.1.4 Gjennomgang av LivsIT i lys av behovene fra SIP

Ser vi behovene vi fant i SIP og sammenholder den med LivsIT får vi denne oversikten:

5.1.4.1 Kommunikasjonsprinsippet

B-1. Det bør være mulig å bruke tjenesten til dialog og ikke bare enveiskommunikasjon

Nei. I utgangspunktet er tjenesten lagt opp som enveiskommunikasjon. I visse tilfeller er det lenker eller e-postadresser til den korrekte offentlige virksomheten, men ikke som en hovedregel. LivsIT "skjuler" til en viss grad hvem som er ansvarlig for informasjonen eller hvem som er den opprinnelige forfatteren.

B-2. Det bør være mulig å tilpasse tjenesten til brukerens konkrete situasjon

Nei. Navigasjonen krever at du selv kjenner til hvilken situasjon du befinner deg i. Informasjonen du finner hvis du kjenner til din situasjon (i henhold til LivsIT-taksonomien) er generell informasjon om situasjonen og ikke konkret informasjon med utgangspunkt i dine spesifikke behov.

¹⁵⁵ <http://follo.selvbetjening.no/servicedescription.aspx?id=bf4888d5-bd81-4db5-a365-9f67012786dd&categoryid=78fcaaa-daf1-4cdd-a3b1-2f95157540aa> pr. 06.02.07

- B-3. **Det bør være mulig å la innbyggeren ta initiativet til kommunikasjon i tjenesten**
Nei. Brukeren er prisgitt informasjonen som ligger i tjenesten og har ingen muligheter for å endre på denne.
- B-4. **Det bør være mulig å la det offentlige ta initiativet til kommunikasjon i tjenesten**
Delvis. Kommunen kan legge inn mer informasjon, men det er begrenset til de allerede eksisterende livssituasjonene som er kategorisert. Andre offentlige instanser er prisgitt hva Norge.no velger å legge inn og da også innenfor rammene av de allerede definerte taksonomiene.

5.1.4.2 Prinsippet om aktiv informasjon

- B-5. **Det bør være mulig å senke kravet til egeninnsats for prioriterte grupper**
Nei. All informasjon er lik, selv om man kan prioritere informasjon til de svakere stilte kan man ikke senke egeninnsatsen gjennom tjenesten.
- B-6. **Det bør være mulig å få tilgang til ulike målgruppers situasjon og behov gjennom tjenesten**
Nei. Selv om statistikk og brukerundersøkelser kan hjelpe det offentlige til å bedre tjenesten, finnes det ikke noen løsning for å gjøre noe med dette gjennom LivsIT direkte.
- B-7. **Det bør være mulig å bruke tjenesten i samarbeide med private organisasjoner og interessegrupper**
Nei. Informasjon fra andre enn offentlig sektor finnes ikke.
- B-8. **Det bør være mulig å nå tjenester fra flere virksomheter gjennom én inngang (offentlig servicekontor)**
Ja.
- B-9. **Det bør være mulig å bruke tjenesten hele døgnet (døgnåpen forvaltning)**
Ja.
- B-10. **Det bør være mulig å dekke det faktiske informasjonsbehovet på en måte som er forståelig for målgruppen**
Delvis. Beskrivelsen for hver livssituasjon kan gjøres konkret og derigjennom øke sannsynligheten for at det er forståelig, men det vil samtidig forutsette mange forskjellige livssituasjoner, som igjen skaper problemer med å finne den som er relevant for brukeren. Slik vi opplever LivsIT-informasjonen vi har funnet, er den preget av at informasjonen er for generell.
- B-11. **Det bør være mulig å se om informasjonen har blitt sett av mottakeren gjennom tjenesten**
Nei. Siden det ikke er noen form for pålogging for å bruke tjenesten kan man ikke utlede noe om brukerne av tjenesten og dermed ikke finne ut om de man ønsker å nå med tjenesten faktisk er de samme som leser den.

5.1.4.3 Helhetsprinsippet

B-12. Det bør være mulig å bruke tjenesten uten å vite på forhånd hvilken virksomhet som står bak

Ja. Tanken bak tjenesten er nettopp at du ikke trenger å ha kunnskap om offentlig sektor for å kunne bruke tjenesten.

B-13. Det bør være mulig å sette sammen tjenester fra flere virksomheter til en helhet

Delvis. Det er mulig å se flere tjenester samlet, men det er ingen interaksjon mellom dem

5.1.4.4 Linjeprinsippet, og 5. Lederansvar

B-14. Det bør være mulig å se hvem som er ansvarlig for tjenesten

Delvis. Det er mulig å se at kommunen er "ansvarlig" for tjenesten LivsIT, men for de ulike tjenestene i strukturen kan ansvaret være skjult for brukeren.

Disse funnene vil, i slutten av dette kapitlet, sammenholdes med de funnene vi finner i vår gjennomgang av Minside.

5.2 Minside

5.2.1 Idé/formål

"Hver enkelt innbygger får nå sin egen skreddersydde side med offentlige elektroniske tjenester og dialog med offentlig forvaltning."¹⁵⁶

Etter tre utsettelse ble Minside lansert mandag 18. desember 2006 kl 11:00. Norge.no annonserte lanseringen med følgende beskrivelse på www.norge.no:

"Internettportalen Minside er nå opna. Minside gir deg ein felles inngang til elektroniske tenester frå offentlege etatar. Du kan og få oversikt over informasjon som er registrert om deg i ulike offentlege register."¹⁵⁷

To og et halvt år tidligere hadde daværende moderniseringsminister lansert ideen om Minside. En beskrivelse av hva Minside skulle bli var blant annet at det skulle være *en inngang* til offentlige tjenester – det skulle være mulig å finne frem til offentlige tjenester uten å vite hvordan det offentlige var organisert. Eksempler på tjenester som ble nevnt tidlig var flyttemelding, endring av skattekortet og påminnelse om EU-kontroll for bilen.

Tanken bak Minside var basert på bankenes suksess med å overlate arbeidet til kunden, rasjonalisere og samtidig få mer fornøyde kunder. Som illustrasjon nevnte Meyer "søknad om ledsagerbevis" som en tjeneste han hadde brukt i hjemkommunen Hamar. Når han søkte om dette via nettet, måtte han fylle ut et skjema, som han så måtte skrive ut, signere, sende i vanlig brev, for at kommunen deretter måtte skrive inn de samme opplysningene i sine datasystemer før Meyer kunne få tilsendt sitt ledsagerbevis. Minside skulle derfor være et viktig skritt på veien til selvbetjeningsløsninger der brukeren gjorde mer av jobben selv, men hvor brukeren samtidig ikke ble spurt om opplysninger som det offentlige allerede hadde.

¹⁵⁶ Inngressen fra FADs forside i forbindelse med lanseringen av Minside 18.12.2006, hentet fra <http://odin.dep.no/fad> [27.12.2006]

¹⁵⁷ Hentet fra "Aktuelt no" på Norge.no, <http://www.norge.no/aktueltno/#nyheter> [26.12.2006]

Minside ble tidlig definert til å være en *portal* – et nettsted som skulle være et utgangspunkt for forskjellige funksjoner og tjenester. Den skulle videre være en ”tynn” portal, i den forstand at den skulle være så generell som mulig, og ha så lite ”forretningslogikk” som mulig. Det var tjenestene som ble tilgjengeliggjort via Minside som skulle inneholde den nødvendige forretningslogikken.

5.2.2 Funksjonalitet

Den versjonen av Minside som ble lansert i desember 2006 inneholdt følgende hovedfunksjonalitet, som vil bli gjennomgått i dette kapitlet:

- 1) Registertjenester
- 2) Transaksjonstjenester
- 3) Kommunale tjenester
- 4) Meldingsboks
- 5) Kalender

For hver av disse tjenestene finnes det implementasjonsguider som viser hvordan etatene kan ta disse tjenestene i bruk. Disse implementasjonsguidene finnes på www.norge.no¹⁵⁸.

Navigasjonen mellom tjenestene på Minside er basert på LivsIT – eller nærmere bestemt førsteutkast til temastruktur i ”offentlig tjenestekatalog”, slik den forelå på nettstedet www.norge.no etter omleggingen til nye nettsider sommeren 2005.

I tillegg inneholdt Minside generell funksjonalitet som hjelpetekster, søk, funksjonalitet for å velge kommuner brukeren er interessert i, og funksjonalitet for tilpasning av hvilke temaer som skal vises i venstremenyen.

5.2.2.1 Registertjenester

Registertjenestenes primære funksjon er å gi innsyn i offentlige registre. Eksempler på registertjenester som var tilgjengelige ved lanseringen var

- 1) ”Mine kjøretøy” som gir innsyn i hvilke kjøretøy som er registrert på en person i Statens vegvesens kjøretøyregister,
- 2) ”Mine eiendommer” som gir innsyn i hvilke eiendommer som er registrert på en person i Norsk eiendomsinformasjons grunnregister,
- 3) ”Min adresse”, som viser et utvalg av opplysningene om en person i folkeregisteret.

Når brukeren er innlogget i Minside og velger en av disse registertjenestene, gjør Minside-applikasjonen en forespørsel til registreieren, og oppgir fødselsnummeret til brukeren. Tilbake får Minside et svar som enten inneholder de opplysningene som finnes om brukeren, eller et svar som sier at det ikke finnes opplysninger om denne brukeren i registeret.

Det er ikke mulig å endre opplysninger gjennom registertjenestene. Dersom brukeren finner feil i opplysningene, må vedkommende henvende seg til de som er ansvarlige for opplysningene, f.eks. Statens vegvesen dersom det er registrert feil opplysninger om bilen brukeren eier. Dersom den ansvarlige etaten tilbyr det, kan denne henvendelsen skje gjennom meldingstjenesten til Minside, men ved lanseringen var det ingen som tilbød dette.

¹⁵⁸ Se <http://www.norge.no/minside/teknisk.asp>. Det er ikke en egen implementasjonsguide for kommunale tjenester, ettersom dette enten er registertjenester eller transaksjonstjenester, med den forskjellen at de bare vises for innbyggerne i den aktuelle kommunen.

Forespørselen og svaret er i form av XML-meldinger. Innholdet i svaret blir ”transformert”, ved hjelp av et ”stilark” (XSL, XML Style Language), til HTML, som så leveres fra Minside-serveren og vises i nettleseren til brukeren. Minside lagrer ikke disse opplysningene utover at de ”caches” i fem minutter, slik at en ny forespørsel etter samme data innen fem minutter, kun vil resultere i at Minside sender den samme nettsiden tilbake til nettleseren. Men etter denne tiden, eller så fort brukeren logger ut av Minside, forsvinner dataene fra Minside servere.

Generelt gir registertjenestene kun *innsyn* i opplysninger; opplysningene blir ikke brukt av Minside på noen måte. Unntaket er registertjenesten ”Min adresse”. Denne henter opplysninger fra folkeregisteret, og denne tjenesten blir automatisk kalt når en bruker logger seg på for første gang.

Oppslag i folkeregisteret ved førstegangs innlogging

Når en innbygger tar i bruk Minside for første gang, må vedkommende først gi samtykke til at Minside skal videreformidle fødselsnummeret for å kunne gjøre oppslag via registertjenestene. Videre samtykker brukeren i at Minside lagrer følgende opplysninger om brukeren: Fødselsnummer, fornavn, etternavn, språkvalg og postnummer. Fødselsnummer og språkvalg overføres til Minside som del av innloggingsprosessen. Men opplysningene fornavn, etternavn og postnummer henter Minside fra folkeregisteret ved å bruke registertjenesten ”Min adresse”.

Dersom Minside av en eller annen grunn ikke får kontakt med registertjenesten ved første gangs innlogging, vil den forsøke igjen ved senere innlogginger. Men så fort disse opplysningene er hentet og lagret, gjøres det ikke sjekker for å se om det er noen av opplysningene som er endret i folkeregisteret. Dette må brukeren selv sørge for ved å velge ”oppdater personlige opplysninger” under ”Endre Minside”, dersom vedkommende f.eks. har skiftet navn eller meldt flytting.

Fornavn og etternavn brukes kun for å synliggjøre at Minside er en personalisert tjeneste, ved at navnet til den innloggete brukeren vises på Minside. Språkvalget brukes for at Minside skal huske fra gang til gang hvilket språk brukeren har valgt. Dette kan brukeren endre ved hjelp av en lenke for språkvalg øverst til høyre på nettsiden.

En interessant opplysning, for vår del, er bruken av postnummeret. Postnummeret brukes for å knytte brukeren til en kommune, som får status som hjemstedskommune. Nederst til høyre presenterer så Minside kontaktinformasjon for hjemstedskommunen. I tillegg vil tjenester fra hjemstedskommunen automatisk bli vist i venstremenyen, mens tjenester fra andre kommuner ikke blir vist med mindre brukeren eksplisitt ber om det.

Bruken av postnummer i Minside er foreløpig det eneste eksempelet på at Minside bruker en opplysning fra et offentlig register til å tilpasse informasjonen som presenteres brukeren. Det er med andre ord det første forsøket på å *målrette* informasjonen basert på hva det offentlige vet om vedkommende. Dersom Minside skal videreutvikles slik at den i større grad skal bruke informasjon fra offentlige registre til å ”hjelp” brukeren videre, vil registertjenestene, som henter disse opplysningene, følgelig være svært viktige. Vi har derfor valgt å gi en mer detaljert beskrivelse av denne tjenestetypen enn de øvrige tjenestetypene. I det følgende kapitlet ser vi konkret på hva som utveksles mellom Minside og etatene.

Nærmere om registertjenestene

Når en bruker velger en registertjeneste, foretar Minside en "web service request" mot tjenesteleverandørens "web service". I retur får Minside en "web service response". Både forespørsel og respons er pakket inn i en konvolutt, "envelope", i henhold til standarden for web services, SOAP.¹⁵⁹

Request – fra Minside

Kodeeksempelet¹⁶⁰ nedenfor viser en "request" fra Minside til en tjenesteleverandør. Denne er felles for alle registertjenestene. Foruten brukerens fødselsnummer inneholder forespørselen foretrukket språk (sprak) og sikkerhetsnivå (autentiseringsstyrke):

```
<?xml version="1.0" encoding="UTF-8"?>
<env:Envelope
  xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:enc="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:ns0="http://www.norge.no/minside/types/request_1_2">
  <env:Body>
    <ns0:MinSideRequest>
      <personid>10101099999</personid>
      <sprak>nb</sprak>
      <autentiseringsstyrke>S</autentiseringsstyrke>
    </ns0:MinSideRequest>
  </env:Body>
</env:Envelope>
```

Figur 9 -- Eksempel på request XML fra Minside til tjenesteleverandør

Response - fra NAV

I kodeeksempelet nedenfor vises en XML-melding med et svar fra tjenesten "Min fastlege", som er en tjeneste fra NAV og som returnerer informasjon fra fastlegeregisteret:

¹⁵⁹ SOAP står for Simple Object Access Protocol, se: <http://www.w3.org/TR/soap/>

¹⁶⁰ Dette kodeeksempelet er hentet fra "Implementasjonsguide for registerinformasjon v 0506". Denne, samt annen teknisk dokumentasjon er tilgjengelig fra Norge.no, se: <http://www.norge.no/minside/teknisk.asp> [30/4-2007]. De øvrige kodeeksemplene i dette kapitlet er hentet fra eksempelkode Minside-prosjektet brukte for å feilsøke en XSL-feil i en registertjeneste 22/12-2006.

```

<?xml version="1.0" encoding="UTF-8" ?>
<soapenv:Envelope xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://tjenester-ws.nav.no/minfastlege-services/wsdl">
  <soapenv:Header />
  <soapenv:Body>
    <MinFastlegeResponseRTV>
      <parameters>
        <responsestatus>
          <returstatus>OK</returstatus>
          <feilkode xsi:nil="true" />
          <feilbeskrivelse xsi:nil="true" />
        </responsestatus>
        <innbygger>
          <fnr>10101099999</fnr>
          <navn>Per Pettersen</navn>
          <status>HARFASTLEGE</status>
        </innbygger>
        <fastlege>
          <navn>Hans Hansen</navn>
          <praksis>
            <navn>Hansen Legesenter</navn>
            <postnr>1234</postnr>
            <gateadresse>Gågata</gateadresse>
            <poststed>Oslo</poststed>
            <telefonnummer>12345678</telefonnummer>
          </praksis>
        </fastlege>
      </parameters>
    </MinFastlegeResponseRTV>
  </soapenv:Body>
</soapenv:Envelope>

```

Figur 10 Eksempel på XML response fra tjenesteleverandør (NAV)

Et interessant funn her er hvordan Minside og NAV ikke bruker samme navn om de samme dataene – fødselsnummer. Minside bruker <personid> og NAV bruker <fnr>. Jf. kap. 3.2.1 den semiotiske trekanten.

Transformasjon med XSL

Nedenfor er et utdrag fra XSL-stilarket for denne registertjenesten. Den første delen av utdraget viser hvordan alle feltnavnene har forskjellige navn på forskjellige språk. I tillegg til de som vises inneholder stilarket også feltnavnene på engelsk (language="en") og samisk (language="se"). Det neste utdraget viser hvordan stilarket tester for innhold i svaret fra fastlegeregisteret, og presenterer det for brukeren.

```

[...]
```

```

<xsl:variable name="labels">
  <label language="nb" id="overskrift">Følgende informasjon er registrert
om min fastlege</label>
  <label language="nb" id="lege">Lege</label>
  <label language="nb" id="praksis">Praksis</label>
  <label language="nb" id="besoksadresse">Besøksadresse</label>
  <label language="nb" id="telefon">Telefon</label>
  <label language="nb" id="HARFASTLEGE">Har registrert fastlege</label>
  <label language="nb" id="HARIKKE FASTLEGE">Har ikke registrert
fastlege</label>
  <label language="nb" id="IKKETILGANG">Fastlegeinformasjon kan ikke vises
på internett</label>
  <label language="nn" id="overskrift">Følgjande informasjon er registrert
om fastlegen min</label>
  <label language="nn" id="lege">Lege</label>
  <label language="nn" id="praksis">Praksis</label>
  <label language="nn" id="besoksadresse">Besøksadresse</label>
  <label language="nn" id="telefon">Telefon</label>
  <label language="nn" id="HARFASTLEGE">Har registrert fastlege</label>
  <label language="nn" id="HARIKKE FASTLEGE">Har ikkje registrert
fastlege</label>
  <label language="nn" id="IKKETILGANG">Fastlegeinformasjon kan ikkje
visast på Internett</label>

```

```

[...]
```

```

<xsl:template name="main">
  <xsl:choose>
    <!-- If the message received is not OK, show error message -->
    <xsl:when
test="//dfn:parameters/dfn:responsestatus/dfn:returstatus!='OK'">
      <div class="registererror"><xsl:value-of
select="//dfn:parameters/dfn:responsestatus/dfn:feilbeskrivelse"/></div>
    </xsl:when>
    <xsl:when test="//dfn:parameters/dfn:innbygger/dfn:status='HARFASTLEGE'">
      <div class="registertitle"><xsl:call-template name="label"><xsl:with-
param name="id" select="'overskrift'"/></xsl:call-template></div>

```

```

[...]
```

Figur 11 Eksempel på utdrag fra XSL-stilark

Den første testlinjen bruker et Xpath-uttrykk, som henter ut innholdet i <dfn:returstatus>-elementet og tester om det er ulikt "OK". Isåfall skrives det ut en feilmelding med teksten hentet fra elementet <dfn:feilbeskrivelse>, som i vårt eksempel er tomt ("xsi:nil="true").

Den andre testlinjen bruker et Xpath-uttrykk, som henter ut innholdet i elementet <dfn:status>, og tester om det er likt "HARFASTLEGE". Det er tilfelle i vårt eksempel, og resten av XSL-stilarket brukes for å kombinere informasjonen om fastlegen med de feltnavnene som er definert tidligere i stilarket.

Resultatet: HTML

Resultatet, etter at XML-svaret fra fastlegeregisteret er "transformert" i henhold til instruksjonene, er følgende HTML-kode:

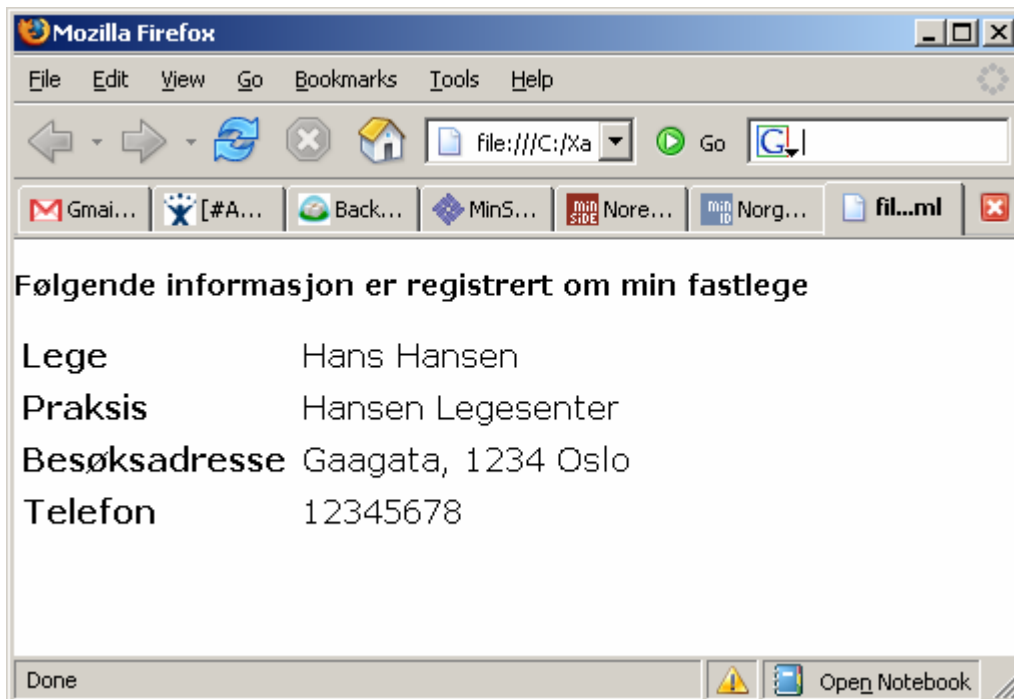

```

<table xmlns:dfn="http://tjenester-ws.nav.no/minfastlege-services/wsd1"
class="registertable" cellspacing="0" cellpadding="4" border="0">
<tr><td class="registerlabel">Lege</td><td>Hans Hansen</td></tr>
<tr><td class="registerlabel">Praksis</td><td>Hansen Legesenter</td></tr>
<tr><td
class="registerlabel">Besøksadresse</td><td>Gaagata, &nbsp;1234&nbsp;0
slo</td></tr>
<tr><td class="registerlabel">Telefon</td><td>12345678</td></tr>
</table>

```

Figur 12 Eksempel på kode transformert fra XML til HTML

I nettleseren, blir dette presentert som følger:



Figur 13 Eksempel på visning av registerdata i nettleser

Som vi så innledningsvis vil denne informasjonen, når den presenteres i selve Minside, bli plassert i ”hjertet” av siden, og med formatering i henhold til stilarket for Minside.



Figur 14 Eksempel på visning av registertjeneste i Minside

Minside bruker en XSL-transformasjonsmotor, Xalan, som er tilgjengelig som åpen kildekode. Implementasjonsguiden for registertjenester viser hvordan de som er interessert kan sette dette opp for å teste dette lokalt.

5.2.2.2 Transaksjonstjenester

Mens registertjenester vises som egne sider på selve Minside, er transaksjonstjenester tjenester der det åpnes nye nettleservinduer, og selve tjenestene ligger på nettstedene til de ansvarlige etatene. Ved lanseringen av Minside var bl.a. følgende transaksjonstjenester tilgjengelige:

- 1) "Melde flytting", som er Skatteetatens flyttemeldingsskjema i Altinn
- 2) "Endre skattekort", som er Skatteetatens tjeneste for å endre skattekort på www.skatteetaten.no

- 3) ”Nedbetalingsplan studielån”, som er Lånekassens side med opplysninger om nedbetalingsplan for studielånet for påloggede kunder.

Idet brukeren klikker på en link til en transaksjonstjeneste, åpnes som nevnt et nytt nettleservindu, med URLen til den valgte tjenesten i adressefeltet. For Skatteetatens skattekorttjeneste er denne URLen følgende:

<https://minsideskort.skatteetaten.no/skd/skattekort/skattekort>

Når nettleseren ber om siden /skd/skattekort/skattekort fra serveren minsideskort.skatteetaten.no vil serveren kontrollere om denne nettleseren representerer en bruker som har identifisert seg for den felles innloggingsløsningen som brukes av Minside og transaksjonstjenestene der. Etersom brukeren allerede er innlogget på Minside, vil svaret være bekreftende, og skattekort-tjenesten vil få overført fødselsnummer, sikkerhetsnivå og språkvalg fra innloggingsløsningen. Dermed kan skattekort-skjemaet fylles ut med de opplysningene Skatteetaten har om brukeren.

Transaksjonstjenester ble i kravspesifikasjonen også omtalt som ”sikre URL-hopp”, og betegner at brukeren skal kunne gå fra et nettsted til et annet, og komme ferdig innlogget frem – såkalt ”Single Sign On”, uten at uvedkommende får innsyn i informasjonen som utveksles. Løsningen forutsetter en ”Identitetsleverandør” som står i mellom, og som garanterer for at en bruker er den han utgir seg for å være. Løsningen som brukes for å oppnå dette for Minside og transaksjonstjenestene, er basert på ”Security Assertion Markup Language” (SAML) versjon 2.0, fra OASIS, og nærmere spesifisert av ”Liberty Alliance”[REF Hekt på referanser til disse mer informasjon].

Sikre URL-hopp er en funksjonalitet som er uavhengig av Minside (men avhengig av identitetsleverandøren). Transaksjonstjenestene i Minside er følgelig kun lenker til andre tjenester som er beskyttet av den felles innloggingsløsningen. Hvem som helst kan legge lenker til de samme tjenestene – og Minside er en tjeneste på lik linje med disse (med adressen <https://minside.norge.no/>)

Transaksjonstjenester har den fordelen at de er svært enkle å legge til i Minside. Man kan ta utgangspunkt i eksisterende tjenester, som f.eks. flyttemeldingen eller tjenesten for å endre skattekortet som begge deler fantes som elektroniske tjenester før Minside ble lansert, og endre disse slik at de kan brukes med den felles innloggingsløsningen for offentlige tjenester. Så fort det er gjort, er det eneste som gjenstår for å legge tjenesten inn på Minside å legge til URLen til tjenesten, gi tjenesten et navn og velge hvor i temastrukturen tjenesten hører hjemme.

Det er etaten som er ansvarlig for tjenestene som styrer hva slags funksjonalitet en transaksjonstjeneste skal tilby. Ideen var at siden registertjenestene er begrenset til å gi innsyn, skulle transaksjonstjenester brukes i tilfeller der det skulle sendes informasjon fra brukeren til etaten, som f.eks. ved en søknadsprosess i form av innsending av et skjema. Men transaksjonstjenester ble fra lansering også brukt til rene innsynstjenester, som f.eks. tjenesten ”Mitt studielån”, som sender brukeren til en side hos Lånekassen som viser opplysninger om eventuelt studielån brukeren har.

Transaksjonstjenester og forhåndsutfylling

Som nevnt over var en av ideene at Minside skulle gjøre det unødvendig for innbyggerne å oppgi opplysninger som det offentlige allerede har om dem, gjennom forhåndsutfylling. Flere av transaksjonstjenestene som er nevnt består av skjema som er forhåndsutfylte eller delvis utfylte. Dette gjelder f.eks. flyttemeldingen og tjenesten for å bytte fastlege. Men denne forhåndsutfyllingen er ikke et resultat av opplysninger som er overført fra Minside til disse tjenestene. Minside overfører *ingen* opplysninger til noen av transaksjonstjenestene. Den felles innloggingsløsningen derimot, kan – på forespørsel fra tjenestene – gi opplysninger om brukeren. Men kun opplysninger som innloggingsløsningen selv besitter. Og per i dag er det kun tre opplysninger; fødselsnummer, språkvalg og sikkerhetsnivå.

Dette betyr at den forhåndsutfyllingen som gjøres i transaksjonstjenestene er avhengig av at etaten selv sitter med de data som skal forhåndsutfylles, og at disse er knyttet til et fødselsnummer, slik at de vet hvilke data som skal brukes. Tjenestene fra Skatteetaten, NAV og Lånekassen fungerer på denne måten, fordi disse etatene bruker fødselsnummer som identifikator i sine systemer, og har tilgang til folkeregisteret for bl.a. informasjon om folkeregistrert adresse.

Andre tjenester derimot, som f.eks. de kommunale tjenestene, kan både mangle knytning til folkeregisteret og deres interne systemer kan bruke andre identifikatorer enn fødselsnummer. Dermed kan brukeren oppleve å velge en tjeneste på Minside, og i neste omgang bli bedt om å oppgi navn og adresse – informasjon som han eller hun nettopp har konstatert er tilgjengelig på Minside under tjenesten ”Min adresse”. Se skjermdump fra [REF ledsagerbevis fra Frogn].

5.2.2.3 Kommunale tjenester

Kommunale tjenester er ingen egen type tjenester, men det er isteden en egen ”merkelapp” på tjenestene for å kunne styre presentasjonen av tjenestene i venstremenyen i Minside. Utgangspunktet er at tjenester fra f.eks. Asker kommune kun er relevante for personer som bor i Asker. Derfor bør ikke tjenestene fra Asker kommune vises for andre enn innbyggerne i Asker. Minside lagrer fem opplysninger om brukerne; fødselsnummer, språkvalg, fornavn, etternavn og postnummer.¹⁶¹ Basert på sistnevnte presenteres alle brukerne av Minside for de tjenestene som finnes i deres hjemkommune. I tillegg kan brukeren velge – gjennom tjenesten ”Mine kommuner” – å få se tjenester fra andre kommuner. Dette er aktuelt for personer som forholder seg til tjenester fra flere kommuner, f.eks. pendlere, personer som har barn i barnehage i nabokommunen, personer med hytte i en annen kommune, etc.

Måten kommunale tjenester i Minside ”filtreres” – ved at det i utgangspunktet kun vises tjenester for den kommunen brukeren er bosatt i – er interessant for vår problemstilling. Dette er det eneste området der Minside bruker en opplysning som den får tilgang til om brukeren, via oppslaget mot folkeregisteret, til å tilpasse innholdet på Minside, slik at det bare er *relevante* tjenester som vises. Denne funksjonaliteten mener vi er i tråd med SIP og målsetningen om at informasjonen skal målrettes.

¹⁶¹ Dette er forøvrig en feil i Minside versjon 1.2. Det er nemlig ikke slik at det postnummeret som er registrert i folkeregisteret på en person ligger i den kommunen vedkommende er bosatt i. Dette må derfor endres slik at Minside istedenfor å lagre postnummer, må lagre kommunenummer.

Funksjonelt sett tilsvarer denne funksjonaliteten et søk. Søkeparameteret som sendes til databasen over tilgjengelige tjenester, er kommunenummeret. I retur får brukeren kun de kommunale tjenestene som tilhører denne kommunen. Det er også mulig for brukeren å legge til kommuner. Dette vil føre til at søkeparameteret blir utvidet med de valgte kommunenes kommunenummer – som et boolsk ”ELLER”, slik at alle tjenester fra alle valgte kommuner returneres.

Dette er en tilsvarende funksjonalitet som vi ser i Regelhjelp.no¹⁶². Her oppgir brukeren organisasjonsnummeret til bedriften vedkommende er interessert i å finne relevante regler for. Regelhjelp.no bruker dette som søkeparameter i et oppslag i Enhetsregisteret¹⁶³, og får tilbake en liste over bransjekoder. Disse kodene blir så gjenbrukt i et boolsk ”ELLER”-søk i Regelhjelp.nos regeldatabase, og bare regler som er koblet til en eller flere av disse bransjekodene, vises.

5.2.2.4 Meldingsboks

Meldingsboksen i Minside gir en enkel funksjonalitet for sikker e-postkommunikasjon mellom det offentlige og innbyggerne. Tradisjonell, åpen e-post har svakheter som gjør den uegnet som kommunikasjonsform til erstatning for f.eks. brev. Dette er svakheter som både er av sikkerhetsmessige og organisatorisk art.

Tradisjonell e-post sendes ukryptert over Internett fra avsender til mottakerens postkasse. I noen tilfeller skjer også henting av e-post i e-postkassen ukryptert, og til og med brukernavn og passord sendes ukryptert når det som (forhåpentligvis) er eieren identifiserer seg for å få tilgang til e-posten sin. Avsendernavn og avsenders e-postadresse er redigerbare felt i en e-postmelding, noe som gjør det lett å utgi seg for å være andre enn man er. Det finnes ingen mekanismer som fungerer i praksis som kan gi avsenderen en bekreftelse på at e-posten er mottatt og eventuelt også lest (dvs. åpnet av mottakeren). Når en etat ønsker å sende en e-post til en innbygger er det dessuten ingen sikker måte å vite hvilken e-postadresse innbyggeren har, ettersom folkeregisteret ikke inneholder e-postadressen til innbyggerne.

Meldingsfunksjonaliteten i Minside er basert på samme teknologi som tradisjonell e-post (SMTP) men med følgende tiltak for å bøte på de ovennevnte svakhetene:

- 1) e-post sendes ikke åpent over Internett mellom Minside og etatene, men transporten krypteres
- 2) innbyggeren kan kun lese e-posten gjennom Minside, og innloggingen og sikkerheten på Minside gir ett forutsigbart sikkerhetsnivå for alle mottakerne¹⁶⁴
- 3) e-postsystemet er lukket, med kontroll over hver nye mottaker og avsender i systemet, slik at det er en tredjepart, Norge.no, som garanterer for at avsenderen er den han eller hun utgir seg for å være
- 4) dersom avsenderen ber om det kan vedkommende få bekreftelse på når meldingen er mottatt og lest
- 5) for de brukerne som har tatt i bruk Minside er e-postadressen basert på fødselsnummeret. Dette er kombinert med en oppslagstjeneste som gjør det mulig å sjekke om en person har tatt Minside i bruk
- 6) en etat kan åpne for å ta i mot spesielle typer henvendelser. Dermed slipper de å ha et apparat på plass som kan håndtere alle mulige henvendelser pr e-post. Et eksempel på

¹⁶² Mer om regelhjelp.no, se: <http://www.regelhjelp.no/> [30/4-2007]

¹⁶³ Enhetsregisteret, Brønnøysundsregistrene, se: <http://w2.brreg.no/enhet/sok/> [30/4-2007]

¹⁶⁴ Autentiseringen skjer gjennom den felles innloggingsløsningen, nettsidene utveksles kryptert (SSL), og de kan ikke mellomlagres (no-cache og no-store direktiv)

en spesiell type henvendelse kan være meldinger om feil i opplysningene som innbyggerne har fått tilgang til i registertjenestene.

Meldingsfunksjonaliteten i Minside har ikke signering.

I den versjonen av Minside som ble lansert 18.12. var det ingen meldingstjenester på plass. Den første meldingstjenesten forventes å være varsel fra Statens vegvesen om periodisk kjøretøykontroll. Denne tjenesten vil i første omgang kun være en-veis, det vil si kun meldinger fra vegvesenet til innbyggeren.

5.2.2.5 Kalender

Minside tilbyr en kalenderfunksjonalitet. Denne kan utstyres med hendelser av Norge.no sentralt – for datoer og frister som er aktuelle for hele eller store deler av befolkningen (f.eks. innleveringsfrist for selvangivelsen). I tillegg kan det, ved bruk av meldingstjenesten, plasseres inn hendelser i enkeltpersoners kalender. En mulig bruk av denne tjenestetypen kan være ved enkeltvedtak. Når etaten oversender enkeltvedtaket til en innbygger, kan de samtidig legge inn relevante frister, som f.eks. klagefrist, i innbyggerens kalender.

For å legge til en hendelse i en innbyggers kalender, sendes en melding til kalendermottaket, med en melding i et gitt XML-format, som bl.a. må inneholde 1) fødselsnummeret til vedkommende som hendelsen gjelder for, 2) tidspunktet for hendelsen og 3) beskrivelse av hendelsen.

Ved lanseringen av Minside var det ikke etablert noen kalendertjenester. Den første forventes å komme fra Statens vegvesen, i tilknytning til meldingstjenesten deres for varsel om periodisk kjøretøykontroll, i form av en kalenderhendelse for siste frist for periodisk kjøretøykontroll.

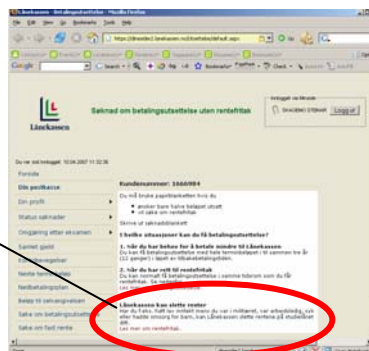
5.2.3 Minside og scenarioet

Dersom Peder og Marte hadde besøkt Minside i dag, ville de ikke funnet bostøtte-ordningen, rett og slett fordi søknad om bostøtte ikke er en blant tjenestene på Minside. Heller ikke søknad om rentefritak er en tjeneste i Minside. Det er riktignok mulig å finne den indirekte, ved å velge tjensten ”Søknad om betalingsutsettelse studielån” under menyvalget ”Skole og utdanning”. Det valget fører brukeren til et søknadsskjema for betalingsutsettelse på Lånkassens nettsider. På den første siden får brukeren informasjon om at skjemaet *ikke* kan brukes dersom vedkommende vil søke om rentefritak. I den forbindelse får vedkommende en kort informasjon om rentefritaks-ordningen:

Lånkassen kan slette renter

Har du f.eks. hatt lav inntekt mens du var i militæret, var arbeidsledig, syk eller hadde omsorg for barn, kan Lånkassen slette rentene på studielånet ditt.

[Les mer om rentefritak.](#)¹⁶⁵



Vi anser det uansett som svært usannsynlig at Peder og Marte, på jakt etter informasjon om rettigheter i forbindelse med å ha blitt foreldre, ville velge å følge menyvalgene ”Skole og utdanning” videre til ”Søknad om betalingsutsettelse” før de valgte ”Les mer om rentefritak”.

Dersom Minside hadde inneholdt tjenesten ”Søknad om bostøtte”, kunne vi risikert samme problem. I LivsIT-kapittelet så vi at informasjon om bostøtte-ordningen stod under menyvalget ”Leie bolig”, som var et undervalg på menyvalget ”Bolig og eiendom”. Dersom ”Søknad om bostøtte” ble lagt til på Minside ville det sannsynligvis havnet under samme menyvalg, eller eventuelt ett nivå opp, direkte under ”Bolig og eiendom”, dersom det ikke var for mange tjenester der allerede. Isåfall ville tjenesten ”Søknad om bostøtte” på Minside ikke vært til noe mer hjelp for Peder og Marte enn bostøtte-informasjonen i LivsIT.

Det er samtidig ingenting i veien for at tjenesten ”Søknad om bostøtte” knyttes til andre menyvalg i Minside. Hvis vi antar at det ble plassert under ”Foreldre og barn”, kan vi anta at Peder og Marte hadde funnet frem til tjenesten. Men isåfall ville det vært en tjeneste som var synlig for alle.¹⁶⁶ Dermed gjenstår det et spørsmål om Peder og Marte hadde forstått at dette var et valg til en tjeneste som var relevant for dem.

Et mulig svar på det spørsmålet er at siden Minside generelt inneholder tjenester som er generelle, så vil en bruker antagelig raskt venne seg til det faktum at det er ”overflødige” tjenester på Minside – som for eksempel alle tjenestene knyttet til studielån vises også for alle som ikke er kunder i Lånekassen. Det er derfor sannsynlig at Peder og Marte ikke vil tolke menyvalget ”Søknad om bostøtte” dithen at det representerer noe de har krav på. Dette avdekker et annet problem. En tjeneste i Minside er kun representert ved navnet på tjenesten. Det er ingen ekstra informasjon som beskriver den, og dermed er det lite grunnlag for brukerne å vurdere om tjenesten gjelder for dem eller ei.¹⁶⁷

Hvis Peder og Marte til slutt hadde valgt tjenesten ”Søknad om bostøtte”, ville de kommet til Husbankens søknadsskjema i Altinn. Forhåpentligvis ville de fått den informasjonen de trenger for å fylle ut skjemaet korrekt, og fått innvilget bostøtte senere. Men søknadsskjemaet hos Altinn gir ingen veiledning i den forstand at innbyggere som ikke har krav på bostøtte får beskjed om det før skjemaet er ferdig utfylt og klar til å sendes – i motsetning til Husbankens bostøtte-kalkulator som gir beskjed så fort det mangler grunnlag for å gå videre.

Mangelen på denne formen for veiledning i bostøtte-skjemaet er selvsagt ikke noe problem for Peder og Marte i vår konkrete situasjon, men på sikt vil det kunne bli det. Ettersom alle brukerne på Minside får tilgang til alle søknadsskjema, er det en fare for at brukerne bruker tid på å fylle ut søknadsskjema for ordninger de ikke har rett på, uten at de får tilbakemelding før skjemaet er ferdig utfylt. Dette kan gi en skjema-tretthet som gjør at man ikke finner det bryet verdt å prøve tjenester som er tilgjengelig i Minside, fordi man erfarer at skjemaene ikke gjelder en selv.

Vi konkluderer med at Minside i liten grad er til hjelp for Peder og Marte slik den er idag, og at det skyldes at Minside ikke har noen sikker mekanisme for å gjøre brukerne oppmerksomme på hvilke tjenester som er relevante for dem i en gitt situasjon. Dette skyldes både at alle tjenester vises for alle, samt at informasjonen knyttet til hver tjeneste er så begrenset at det gir et dårlig grunnlag for å vurdere relevansen av tjenesten.

5.2.4 Minside og SIP

Nedenfor er en gjennomgang av kravene vi har utledet fra SIP, med funksjonaliteten i Minside. Målsetningen er å avgjøre om Minside oppfyller kravet eller ei.

5.2.4.1 Kommunikasjonsprinsippet

B-1. Det bør være mulig å bruke tjenesten til *dialog* og ikke bare enveiskommunikasjon

Ja. Minside er en personalisert tjeneste som kan brukes som kanal for å motta individuell informasjon via meldingsboksen og kalenderen, og innbyggeren har returkanaler i form av meldingstjenesten og eventuelt skjemaer.

B-2. Det bør være mulig å tilpasse tjenesten til brukerens konkrete situasjon

Delvis. Den versjonen som tilbys i dag bruker opplysningen om innbyggerens hjemstedskommune (postnummer fra folkeregisteret) til å tilpasse menyen slik at bare tjenester fra hjemstedskommunen presenteres. Men det er begrensninger både i form av hvilke opplysninger Minside har om innbyggeren, og mekanismer for å utnytte dette til tilpasning av tjenesten. Et eksempel på mangelfull tilpasning er at innholdet i menyen ikke tilpasses i forhold til alder eller kjønn, selv om dette er informasjon som fremgår av fødselsnummeret.

B-3. Det bør være mulig å la innbyggeren ta initiativet til kommunikasjon i tjenesten

Ja. Meldingstjenesten kan gi innbyggeren mulighet til å sende melding til en etat. Transaksjonstjenestene gir også innbyggerne mulighet til å ta initiativ overfor en etat i form av å f.eks. sende en søknad.

B-4. Det bør være mulig å la det offentlige ta initiativet til kommunikasjon i tjenesten

Ja. Meldingstjenesten gjør det mulig for det offentlige å sende en melding til innbyggeren, på etatens initiativ.

5.2.4.2 Prinsippet aktiv informasjon

B-5. Det bør være mulig å senke kravet til egeninnsats for prioriterte grupper

Delvis. Vi kommer riktignok ikke utenom kravet om at brukeren på en eller annen måte må logge seg inn på Minside, men dersom vi forutsetter at brukeren er der, kan prioriterte grupper få informasjon *tilsendt* via meldingsboksen eller kalenderhendelser, fremfor å måtte finne frem til informasjonen selv, forutsatt at den ansvarlige etaten har oversikt over hvilke konkrete personer som er innenfor den prioriterte målgruppen. Derimot er det bare i begrenset grad mulig å automatisk fremheve den informasjonen som er ekstra viktig for prioriterte grupper, å la mekanismen for visning av tjenester fra hjemstedskommunen i menyen. Til det vet Minside for lite om innbyggeren.

B-6. Det bør være mulig å få tilgang til ulike målgruppers situasjon og behov gjennom tjenesten

Delvis. I motsetning til en åpen netjtjeneste, kan en personalisert tjeneste som krever innlogging, slik Minside gjør, knytte bruksmønsterstatistikk til individene. Dessuten kan opplysninger om innbyggeren som alder og bosted (fra fødselsnummer og poststed) brukes til å identifisere grupper. Ved å kombinere dette kan bruksmønsterstatistikk gi et bilde av hva forskjellige brukergrupper er opptatt av. Meldingstjenesten gir også en mulighet for å få direkte tilbakemeldinger fra innbyggerne om hvilke behov de har.

Men etter vår mening er ingen av disse mekanismene gode nok til å være en god løsning på behovet for å få tilgang til ulike målgruppers situasjon og behov.

B-7. Det bør være mulig å bruke tjenesten i samarbeide med private organisasjoner og interessegrupper

Nei. Minside er en personlig tjeneste. Det er riktignok mulig for en person å gi andre tilgang på sine vegne ved å dele autentiseringsinformasjonen (fødselsnummer samt PIN-koder og passord), men etter vår mening er ikke det det samme som at tjenesten er laget for slik bruk.

B-8. Det bør være mulig å nå tjenester fra flere virksomheter gjennom én inngang (offentlig servicekontor)

Ja. Minside er nettopp en inngang til flere virksomheters tjenester.

B-9. Der er mulig å bruke tjenesten hele døgnet (døgnåpen forvaltning)

Ja.

B-10. Det bør være mulig å dekke det faktiske informasjonsbehovet på en måte som er forståelig for målgruppen

Nei. Det er vanskelig å se for seg at Minside generelt sørger for at det *faktiske* informasjonsbehovet dekkes på en *forståelig* måte for målgruppen. Etter vår mening er det kun en sum av tjenester som også inkluderer individuell veiledning, uavhengig av teknologi, som kan komme i nærheten av å tilfredsstillende et så vidt favnende krav.

B-11. Det bør være mulig å se om informasjonen har blitt sett av mottakeren gjennom tjenesten

Delvis. Loggnivået kan settes slik at det er mulig å se hva hver enkelt bruker har gjort, og dermed er det teoretisk mulig å se om en bruker som man vet tilhører en målgruppe, har sett informasjonen/tjenestene som er tiltenkt denne målgruppen. I praksis er dette derimot vanskelig gjennomførbart, både på grunn av omfanget av loggingen, og fordi det ofte vil være vanskelig å vite hvem informasjonen konkret er myntet på.

Dersom det på forhånd er kjent hvem som skal ha informasjonen, vil det isteden være bedre å sende en personlig melding gjennom meldings- eller kalendertjenesten. I så fall vil avsenderen få en bekreftelse når meldingen er *åpnet*. Det er selvsagt ikke det samme som at informasjonen er lest eller forstått. Eksempel på slik bruk av tjenesten er varsling av frist for periodisk kjøretøykontroll. Her sendes varselet til kjøretøyeieren, og Statens vegvesen kan se om vedkommende har åpnet meldingen.

Det kan også tenkes at informasjonen gis via et registeroppslag. En tenkt tjeneste kan være "har du rett til stønad x"? Registertjenesten sender brukerens fødselsnummer til ansvarlig etat, som så opplysninger knyttet til fødselsnummeret, og returnerer svaret, f.eks. i form av en bekreftelse og informasjon om hvor/hvordan det er mulig å søke om stønaden. Her vil det i så fall være den ansvarlige etaten som eventuelt logger at brukeren har brukt tjenesten.

5.2.4.3 Helhetsprinsippet

B-12. Det bør være mulig å bruke tjenesten uten å vite på forhånd hvilken virksomhet som står bak

Ja. Dette sikres gjennom bruk av LivsIT-temaord for navigasjon mellom tjenestene.

B-13. Det bør være mulig å sette sammen tjenester fra flere virksomheter til en helhet

Delvis. Minside er i seg selv en løsning for å sette sammen tjenester fra flere virksomheter. Men samspillet mellom tjenestene er avhengig av at brukeren selv interagerer med dem. Med det menes det at en tjeneste fra en virksomhet ikke utveksler data med en annen tjeneste hos en annen virksomhet via Minside. Et eksempel som illustrerer dette kan være flyttemelding og tjenesten for å bytte fastlege. Gjennom flyttemeldingen, som leveres av Skatteetaten (via Altinn), kan en innbygger melde flytting til en annen del av landet. Dersom flyttemeldingen og tjenesten for å bytte fastlege utgjorde en sammensatt tjeneste, kunne man se for seg at Minside sørget for at brukeren som nettopp hadde levert flyttemelding, fikk tilbud om å gå til tjenesten for å bytte fastlege, med liste over ledige fastleger i området vedkommende flyttet til.¹⁶⁸

5.2.4.4 Linjeprinsippet, og Informasjon som lederansvar

B-14. Det bør være mulig å se hvem som er ansvarlig for tjenesten

Ja, alle tjenestetypene gir den ansvarlige etaten mulighet til å ustyre informasjonen med tydelig avsender, enten det er registerinformasjon, meldingstjenesten, kalenderhendelser eller transaksjonstjeneste. Sistnevnte tjenestetype er spesiell, ettersom brukeren her vil befinne seg på den ansvarlige etatens nettsted.

5.3 Oppsummering av behovene for en Semantisk Minside

Som en oppsummering av behovsanalysen sammenfatter vi, i tabellen nedenfor, de behovene vi kom frem til at den statlige informasjonspolitikken har til offentlig informasjon, og hvilke av disse behovene som er tilfredsstilt gjennom prosjektene LivsIT og Minside:

	LivsIT	Minside
Kommunikasjonsprinsippet		
B-1. Det bør være mulig å bruke tjenesten til dialog og ikke bare enveiskommunikasjon	NEI	JA
B-2. Det bør være mulig å tilpasse tjenesten til brukerens konkrete situasjon	NEI	DEL VIS
B-3. Det bør være mulig å la innbyggeren ta initiativet til kommunikasjon i tjenesten	NEI	JA
B-4. Det bør være mulig å la det offentlige ta initiativet til kommunikasjon i tjenesten	DEL VIS	JA
Prinsippet om aktiv informasjon		
B-5. Det bør være mulig å senke kravet til egeninnsats for prioriterte grupper	NEI	DEL VIS
B-6. Det bør være mulig å få tilgang til ulike målgruppers situasjon og behov gjennom tjenesten	NEI	DEL VIS
B-7. Det bør være mulig å bruke tjenesten i samarbeide med private organisasjoner og interessegrupper	NEI	NEI
B-8. Det bør være mulig å nå tjenester fra flere virksomheter gjennom én inngang (offentlig servicekontor)	JA	JA
B-9. Det bør være mulig å bruke tjenesten hele døgnet (døgnåpen forvaltning)	JA	JA
B-10. Det bør være mulig å dekke det faktiske informasjonsbehovet på en måte som er forståelig for målgruppen	DEL VIS	DEL VIS
B-11. Det bør være mulig å se om informasjonen har blitt sett av mottakeren gjennom tjenesten	NEI	DEL VIS
Helhetsprinsippet		
B-12. Det bør være mulig å bruke tjenesten uten å vite på forhånd hvilken virksomhet som står bak	JA	JA
B-13. Det bør være mulig å sette sammen tjenester fra flere virksomheter til en helhet	DEL VIS	DEL VIS
Linjeprinsippet og Lederansvar		
B-14. Det bør være mulig å se hvem som er ansvarlig for tjenesten	DEL VIS	JA

Som man ser av tabellen svarer Minside på alle de samme behovene som LivsIT. I tillegg besvarer den flere som LivsIT har fått "NEI" eller "DEL VIS" på. Dette passer fint inn i vårt undersøkelsesopplegg i det vi går videre med utgangspunkt i Minside. I neste kapittel vil vi ta

for oss de behovene som ikke blir tilfredsstilt i Minside, og bruke disse for å utlede krav i en funksjonell kravspesifikasjon til vår ”Semantiske Minside”.

6 Funksjonell kravspesifikasjon

I forrige kapittel avdekket vi et gap mellom dagens Minside og LivsIT sammenlignet med de behov vi utledet fra den statlige informasjonspolitikken i kapittel 4. I dette kapittelet skal vi gå videre med det gapet, som utgjør de antatte behovene for videreutvikling av dagens Minside, og lage en funksjonell kravspesifikasjon for en videreutviklet løsning av Minside.

I tabellen nedenfor har vi listet opp resultatet av gapanalysen i forrige kapittel.

	LivsIT	Minside
Kommunikasjonsprinsippet		
2. Det bør være mulig å tilpasse tjenesten til brukerens konkrete situasjon	NEI	DELVIS
Prinsippet om aktiv informasjon		
5. Det bør være mulig å senke kravet til egeninnsats for prioriterte grupper	NEI	DELVIS
6. Det bør være mulig å få tilgang til ulike målgruppers situasjon og behov gjennom tjenesten	NEI	DELVIS
7. Det bør være mulig å bruke tjenesten i samarbeide med private organisasjoner og interessegrupper	NEI	NEI
11. Det bør være mulig å dekke det faktiske informasjonsbehovet på en måte som er forståelig for målgruppen	DELVIS	DELVIS
Helhetsprinsippet		
14. Det bør være mulig å sette sammen tjenester fra flere virksomheter til en helhet	DELVIS	DELVIS

6.1 Forholdet til de øvrige behovene i SIP og andre krav

Tabellen inneholder behov som ikke er dekket. Vi bygger videre på dagens Minside - og dermed dekker vi de samme behovene som den gjør. Men for å sikre at den nye funksjonaliteten vi utvikler ikke går på tvers av de behovene som allerede er dekket, må vi tillegg ta hensyn til den samlede listen med utledete behov fra SIP. For eksempel er behovet for å kunne knytte informasjon til den ansvarlige etaten (fra linjeprinsippet og prinsippet om informasjon som lederansvar) et behov som vi må sørge for at blir understøttet i den nye funksjonaliteten.

På samme måte som de øvrige behovene fra SIP er relevante ved videreutvikling av løsningen, finnes det også et sett av andre krav som er styrende for systemutvikling i offentlig sektor¹⁶⁹. Vi ser bort fra slike krav i denne fremstillingen, med et lite unntak for personvern. Vi gjør riktignok ikke en fullstendig gjennomgang av alle personvernrelaterte problemstillinger, men forsøker å følge prinsippet om å unngå å lagre personopplysninger så langt det lar seg gjøre.

6.2 Utledning av krav for hvert behov

Her vil vi gå gjennom hvert av de gjenstående behovene (B) fra tabellen over og komme frem til ulike krav (K) vi mener må stilles til en ny tjeneste for at behovene skal tilfredsstilles.

6.2.1 B-2. Det bør være mulig å tilpasse tjenesten til brukerens konkrete situasjon

Det offentlige sitter i dag på en mengde opplysninger om hver enkelt innbygger. Dersom disse opplysningene ble sett under ett, ville det gi et bilde av en konkret situasjon.¹⁷⁰

Opplysningene vil ikke kunne fange alle fasetter av en persons situasjon, og detaljgraden ville være varierende. Det er også et spørsmål om større eller mindre deler av det samlede bildet ville være utdatert. Men det er sannsynlig at det i mange tilfeller ville være tilgjengelig tilstrekkelig informasjon til å kunne tilpasse en aktiv veiledningstjeneste slik at den ble spisset inn mot en helt konkret situasjon. I scenarioet vårt ville ikke Husbanken trenge å spørre Peder og Marte om hvorvidt de har barn, eller om noen av dem mottar stønader, dersom Husbanken hadde hatt tilgang til alle opplysningene hos NAV og Folkeregisteret. Det offentlige sitter allerede med den nødvendige informasjonen.

Selv om en slik løsning kan virke innbyggervennlig ettersom innbyggeren slipper å svare på spørsmål om seg selv, er det neppe ønskelig ut ifra et personvernenssyn at *alle* etater skal ha tilgang til *alle* opplysninger om *alle* innbyggere.¹⁷¹

En annen side av det å få konkret informasjon er fravær av irrelevant informasjon. Informasjon som brukeren opplever som irrelevant bør "skjules". Opplysningstypen alder vil for eksempel enkelt kunne brukes til å konstatere at informasjon om eldre personer er lite relevant for dagens 17-åringer.

I Minside er tanken at det er innbyggeren selv som har tilgang til opplysningene om seg, og kan se opplysninger fra flere etater.¹⁷² Dette er opplysninger som ikke er tilgjengelige for de enkelte etatene. For eksempel har ikke Folkeregisteret opplysning om hvilken fastlege innbyggeren har, selv om det er opplysninger både fra folkeregisteret og fastlegeregisteret tilgjengelig i Minside. Ved å bruke opplysningene som er tilgjengelige i Minside som grunnlag for tilpasning av tjenester, uten at etatene får tilgang til dem, kan tjenesten tilpasses brukerens konkrete situasjon uten at alle etater må dele all informasjon om innbyggeren.

Dersom vi tar utgangspunkt i at brukeren går til Minside, mener vi vedkommende bør oppleve følgende:

- 1) Lenker til tjenester som er relevante for brukeren synliggjøres eller markeres. Med relevant mener vi her at tjenesten det lenkes til representerer en rettighet vedkommende har, eller kan ha, krav på.
- 2) Det er mulig å få se informasjon om *hvorfor* tjenesten er synliggjort/markert i henhold til pkt 1 over.
- 3) Tjenester som ikke er relevante i henhold til definisjonen i punkt 1, skjules eller markeres på andre måter.
- 4) Det er mulig å få velge å få se oversikt over tjenestene som er skjult i henhold til pkt 3, inkludert informasjon om *hvorfor* de er skjult.
- 5) Ved valg av tjeneste, bør skjema være forhåndsutfylt/ikke spørre etter opplysninger som allerede er tilgjengelig for brukeren.

Use case 1: Marte og Peder *før* de får barn:

Logger inn på Minside

Tjenesten "Rentefritak" og "Bostøtte" vises ikke

Velger oversikt over "skjulte" tjenester.

Finner Bostøtte og Rentefritak

Finner informasjon om opplysningstyper som enten mangler eller har verdi som gjør at de ikke har krav på rettigheten

Use case 2: Marte og Peder har fått barn

Logger inn på Minside

Tjenestene "Rentefritak" og "Bostøtte" vises/markeres som ytelser brukeren kan ha krav på.

Går til "Rentefritak"

Alle opplysninger det offentlige har er forhåndsutfylt i søknadsskjemaet

Fyller ut eventuelt andre opplysninger. Fortløpende oppdatering av eventuelt simulert resultat.

Avslutter (trykker "Send" eller "Lagre")

Går til "Bostøtte"

Alle opplysninger det offentlige har er forhåndsutfylt i søknadsskjemaet

Fyller ut eventuelt andre opplysninger. Fortløpende oppdatering av eventuelt simulert resultat.

For at S.M.S. og tjenestene som er tilknyttet S.M.S. skal kunne tilpasses brukerens konkrete situasjon, må informasjon om brukerens situasjon være tilgjengelig. Opplysninger om brukeren er tilgjengelig for Minside i dag via registertjenestene. Men dagens Minside er kun en løsning for å *presentere* registeropplysningene for brukeren, og det er først når brukeren aktiverer en registertjeneste at opplysningene hentes. For at opplysningene fra registertjenestene skal kunne brukes for å tilpasse tjenesten og forhåndsutfylle skjema, må derfor opplysningene om brukeren være tilgjengelig på S.M.S.'s initiativ, og slik at de kan "behandle" utover bare det å vise dem frem til brukeren. Basert på dette kommer vi frem til følgende krav (K).

- K-1. S.M.S. må i innloggingsperioden kunne hente opplysninger om den innloggede brukeren
- K-2. S.M.S. må kunne bruke de innhentede opplysningene om brukeren til å tilpasse tjenesten basert på disse opplysningene.

Som vi har sett tidligere er Minsides bruk av registertjenester fokusert på *presentasjonen*. Minside vet ikke hva en opplysning betyr, kun hvordan den skal vises for brukeren. S.M.S. skal gjøre automatisert veiledning på grunnlag av de samme opplysningene, og også kunne gjenbruke dem til forhåndsutfylling av eventuelle skjema. Derfor trengs det en måte å identifisere opplysninger på tvers av systemene.

- K-3. Det må finnes en måte å identifisere opplysningstyper på tvers av S.M.S. og tjenestene som er tilknyttet S.M.S.

Når S.M.S. har tilgang til opplysningene om brukeren må det foretas en automatisert rettslig resonnering for å avgjøre om en konkret tjeneste er relevant for brukeren eller ei, for at S.M.S. skal kunne tilpasse informasjonen som gis til brukeren, og gi automatisert rettslig veiledning.

- K-4. Basert på vurderingen av de innhentede opplysningene om brukeren skal S.M.S. tilpasse informasjonen slik at bare tjenester som er relevante for brukeren vises.

Vurderingen av opplysningene i henhold til krav K-4 over innebærer transformasjon av rettsregler, og S.M.S. må derfor understøtte elementene i Schartums transformasjonsmetode. Kravene nedenfor er relatert til den rettslige informasjonsmodell og den rettslige prosessmodellen slik i henhold til Schartums metode:

- K-5. Opplysningene i S.M.S. må kunne knyttes til opplysningstyper i informasjonsmodellen i det transformerte regelverket
- K-6. S.M.S. må ha støtte for de operasjonene som er angitt som del av prosessmodellen

Når en tjeneste tilpasses opplysninger om brukerens konkrete situasjon, vil det i praksis innebære forhåndsutfylling av skjema. Avhengig av utformingen av tjenesten kan dette ha form av at menyer i tjenesten tilpasses slik at kun relevante valg vises, tilsvarende som for menyen i Minside, eller det kan være datafelt som inneholder ferdig utfylte data. Men det er viktig at det går klart frem for brukeren hvor data er hentet fra og hvordan skjemaet er automatisk utfylt, og det bør også være mulig å overskrive opplysninger enten fordi de er foreldet, gale eller fordi brukeren ønsker å se hva resultatet kan bli med andre verdier.¹⁷³

- K-7. Når brukeren velger en tjeneste skal skjema være forhåndsutfylt med data som allerede er kjent for Minside
- K-8. Brukeren må ha mulighet til å overskrive de forhåndsutfylte dataene
- K-9. Brukeren må få se hva som er kilden til de forhåndsutfylte dataene

Vårt fokus er at brukeren så enkelt som mulig skal finne informasjon om *sine* rettigheter. Derfor ønsker vi at teknologien skal ha mulighet for å håndtere tilpasning av navigasjonsmenyen slik at bare tjenester som er relevante vises. Men det er viktig at brukeren enkelt kan få se hva som er filtrert bort, og også hva som er grunnlaget for denne filtreringen. Umiddelbart ser vi flere gode grunner for dette. For det første kan det tenkes at det er feil i den automatiserte tilpasningen, for det andre kan det påpekes at det blant annet for den demokratiske deltagelsen er viktig å få informasjon om andre tjenester, for det tredje vil mange være på søken etter informasjon på vegne av andre, og for det fjerde kan det være viktig å orientere seg om rettigheter man kan få ved å gjøre endringer i sin egen situasjon (for eksempel ved å gifte seg fremfor å forbli samboende, eller gå opp eller ned i inntekt).

- K-10. Brukeren skal kunne se hvorfor de tjenestene som er markert som relevante er vurdert som relevante
- K-11. Brukeren skal ha kunne velge en visning som viser hvilke tjenester som er skjult, inkludert (lenke til) informasjon om hvorfor.
- K-12. Brukeren skal ha mulighet til å deaktivere denne automatiske tilpasningen av informasjon dersom vedkommende ønsker det

Funksjonaliteten beskrevet over innebærer at opplysninger om brukeren skal gjøres tilgjengelige i en form for "arbeidsminne" for bruk av S.M.S. Hver bruker har sin adskilte del av dette arbeidsminnet, slik at kun brukeren har tilgang til sine opplysninger. Det er viktig at det ikke lages et nytt "superregister" som inneholder alle opplysninger om alle brukere. Dette sikres på samme måte som med dagens Minside, ved at opplysningene ikke hentes med mindre brukeren er innlogget og har autorisert det, og at opplysningene slettes fra arbeidsminnet så fort brukeren logger ut igjen (eller blir logget ut etter en viss tid med inaktivitet).

- K-13. S.M.S. henting av opplysninger om brukeren må kun skje når brukeren er innlogget og har gitt sitt samtykke til det.
- K-14. Når brukeren er logget ut, skal ingen opplysninger om brukeren lenger være lagret.

6.2.2 B-5. Det bør være mulig å senke kravet til egeninnsats for prioriterte grupper

Gitt at det er mulig å identifisere de prioriterte gruppene ut ifra informasjon det offentlige allerede har, vil det være mulig å senke kravet til egeninnsats for disse ved å bruke funksjonaliteten for å tilpasse tjenesten til brukerens konkrete situasjon, se kravene knyttet til behov B-1 over.

Prioriteringsmekanismen blir dermed først og fremst knyttet til hvilke rettighetsordninger man først velger å bygge inn støtte for i S.M.S. Dersom småbarnsfamilier med lav inntekt blir definert som en prioritert gruppe, vil en måte å senke kravet til egeninnsats for disse være å prioritere å gjøre tilgjengelig de nødvendige opplysningene i S.M.S. og formalisere de delene av regelverket som må til for at S.M.S. skal kunne gi automatisert veiledning om muligheten for for eksempel bostøtte og eventuelt rentefritak til denne gruppen.

En annen måte å senke kravet til egeninnsats for prioriterte grupper, vil være gjennom utforming av informasjonen slik at den informasjonen man mener er viktigst for dem, kommer tydelig fram. For en nettside kan dette gjøres gjennom plassering på siden, samt gjennom valg av ord og språkføring. Hva som er den beste løsningen her, vil det typisk være behov for brukertester for å komme frem til, med andre ord er dette en jobb for blant annet fagområdet ”informasjonsdesign”.

6.2.2.1 Konklusjon

Utover behovet for å tilpasse tjenesten til brukerens konkrete situasjon, og som er dekket av kravene under behov B-1 over, mener vi behovet om å senke kravet til egeninnsats for prioriterte grupper først og fremst har ikke-tekniske løsninger, og velger derfor å ikke ta med krav herfra i det videre arbeidet.

6.2.3 B-6. Det bør være mulig å få tilgang til ulike målgruppers situasjon og behov gjennom tjenesten

For å kunne gi best mulig informasjon til sine målgrupper trenger etatene kunnskap om målgruppene. En mulighet for å få slik informasjon er gjennom spørreundersøkelser, en annen er gjennom analyse av den informasjonen de har om sine eksisterende brukere/kunder. For nett-tjenester kan det også ligge informasjon i å analysere bruksmønsteret av nettstedet gjennom å analysere nettstedets logger. Problemet med tolkning av logger for bruksmønster er at det er vanskelig å avgjøre om bruken avdekker hva brukeren opplever som nyttig, eller om de avslører hva brukeren blir ”lurt” til å gjøre.

S.M.S. kan tenkes å bidra til bedre beskrivelse av brukernes situasjon ettersom logininformasjon i S.M.S. kan gi et mer helhetlig bilde av situasjonen enn ved bruk av enkeltetatens nettsteder. I S.M.S. kan det fremgå om samme bruker benytter seg av tjenester fra to eller flere etater. Loggene fra hver etats nettsted derimot, kan kun si noe om hva en bruker gjør innenfor rammen av deres tjenester. Men dette er isåfall logininformasjon som dagens Minside også vil kunne gi.

S.M.S.’ automatiserte veiledning vil til gjengjeld gi økt sannsynlighet for at en tjeneste som blir brukt også representerer en rettighet, og dette kan gjøre det aktuelt å tolke loggene mer i retning av informasjon om hvilken situasjon brukeren er i, istedenfor informasjon om hva de gjør på nettstedet.

S.M.S. bør i tillegg ha en mekanisme for å gi enkel brukertilbakemelding. Dette kan være av typen ”var denne informasjonen nyttig”-knapp.

- K-15. S.M.S. bør tilby en feedback-mekanisme hvor brukerne kan angi om tilpasningen av informasjon oppleves relevant.

Uansett vil vi påpeke at etatens behov for å få tilgang til ulike målgruppers situasjon har som *formål* å kunne målrette informasjonen bedre. Gjennom kravene utledet i B-1 kan de treffe målgruppen basert på deres konkrete situasjon, og på den måten også dekker dette behovet.

6.2.4 B-7. Det bør være mulig å bruke tjenesten i samarbeide med private organisasjoner og interessegrupper

Selv om informasjonen om rettigheter gjøres bedre tilgjengelig fra det offentlige side, vil det uansett være viktig å kunne involvere en tredjepart. Det offentlige er i praksis ”motpart” i saker der innbyggere gjør krav på rettigheter, og selv om rettssikkerhetsgarantier skal sikre at folk får det de har krav, på vil det være naivt å tro at det offentlige fullt og helt kan passe på seg selv. Selvbetjeningsløsninger på nettet kan på den ene siden ha den fordelen at den gjør det enklere å be tredjeparter om hjelp til å kreve en rettighet. På den annen side kan selvbetjeningsløsningen være slik at det blir så ”behagelig” å velge den enkleste løsningen, at man i mindre grad tar seg bryet med å søke hjelp fra tredjepart. For eksempel kan man ha planer om å sjekke selvangivelsen hos en fagperson, men ender opp med å velge den raske løsningen (en tekstmelding) når fristen nærmer seg.¹⁷⁴ Dermed kan vi risikere at det offentlige i mindre grad blir ”kvalitetssikret” av kvalifiserte tredjeparter.

Slik vi ser det kan private organisasjoner og interessegrupper involveres på to måter. For det første bør de på eget initiativ kunne få innsyn i hvordan regelverket er transformert til kildekode som så er grunnlaget for den automatisert veiledningen.

- K-16. Kildekoden som inneholder transformert regelverk må være åpent tilgjengelig på nettet.

For å lette muligheten til å kontrollere regelverket må det være lett å identifisere hvilke deler av kildekode som inneholder hvilket regelverk.

- K-17. Hver del av kildekode som inneholder transformert regelverk, må referere til den regelen som den implementerer.
- K-18. Referansen til regelverket må være så spesifikk som mulig. Som et minimum må den referere med URI til riktig paragraf i Lovdata, i tillegg til eventuelt fritekst for å spesifisere ytterligere.

Kildekoden bør videre angi eksplisitt hvilke elementer i transformasjonsmodellen kildekode representerer, f.eks. prosess (aritmetisk, logisk, dataauthenticing), opplysningstype samt eventuelt legaldefinisjon.

- K-19. Kildekoden bør synliggjøre hvilke deler av Schariums transformasjonsmetode den implementerer.

Den andre måten private organisasjoner og interessegrupper bør kunne involveres på, er gjennom å tilby brukere av Minside å benytte det vi kan kalle ”automatiserte rådgivningstjenester”. Som et eksempel på dette kan vi se for oss at Aleneforeldreforeningen,

har en kalkulator for skatteberegning for 2007 à la den som finnes hos privatøkonomibladet Dine Penger¹⁷⁵, som kan brukes for å gi tips om opplysninger som kan gi et mer korrekt bilde av situasjonen til en aleneforsørger. Ulempen med kalkulatoren er at det tar tid å bruke den, fordi det er så mange opplysninger som skal fylles ut.¹⁷⁶ I stedet burde det være mulig for en S.M.S.-bruker å ”importere” den logikken som Aleneforeldreforeningen har lagt inn, og kjøre denne på de opplysningene som er tilgjengelig i S.M.S.¹⁷⁷

K-20. Det må være mulig å anvende regelsett fra private organisasjoner og interessegrupper (tredjepart) på opplysningene i Minside.

For å unngå databehandlingsansvar hos flere parter enn nødvendig skal personopplysningene i S.M.S. ikke måtte sendes til de som tilbyr regelsett.

K-21. Anvendelse av andres regelsett må kunne skje uten at personopplysningene i Minside flyter fra Minside til tredjeparten.

6.2.5 B-11. Det bør være mulig å dekke det faktiske informasjonsbehovet på en måte som er forståelig for målgruppen

Når tjenesten er tilpasset en brukers konkrete situasjon, kan kommunikasjonen gjøres mye mer konkret enn ved generelle tjenester/generell informasjon. Muligheten for å konkretisere informasjon er dekket gjennom kravene i B-1.

Å sikre at informasjonen er *forståelig* for brukeren er på sin side en ikke-teknisk oppgave som bl.a. krever brukerundersøkelser og hører innunder blant annet informasjonsdesign. Som del av informasjonsdesignjobben er bruksmønsterstatistikk som kan si noe om hvordan tjenesten blir brukt nyttig, jfr også B-6 over.

6.2.6 B-14. Det bør være mulig å sette sammen tjenester fra flere virksomheter til en helhet

Per idag kan skjema kun forhåndsutfylles med opplysninger som den ansvarlige etaten selv har tilgang til. Ved å benytte opplysninger tilgjengelige for Minside til forhåndsutfylling, kan et skjema forhåndsutfylles også med data fra andre etater. Dette tilsvarer funksjonaliteten som beskrevet under pkt B-2, krav 7, om forhåndsutfylling av skjema.

En annen form for sammensatte tjenester vil være å sette sammen flere enkelttjenester fra flere etater, til en samlet tjeneste. Vi kan se for oss en samletjeneste for stønader til foreldre, som, avhengig av situasjonen brukeren er i, inkluderer en eller flere av tjenestene til NAV, samt eventuelt Husbankens bostøtte-tjeneste og søknad om rentefritak fra Lånekassen. Dette dekkes isåfall gjennom funksjonaliteten for filtrering av relevante tjenester, kombinert med koblingen av en tjeneste til et tema i LivsIT-strukturen.

En tredje variant kan være at gjennomføringen av en tjeneste ”trigger” en annen tjeneste. Eksempelet fra forrige kapittel viser dette, der flyttemeldingen kan generere en opplysningsendring – endret adresse – som ”trigger” tjenesten for bytte av fastlege.

K-22. Det må være mulig å relatere tjenester til hverandre

6.3 Oppsummering av kravene

Nedenfor er kravene fra kravspesifikasjonen oppsummert

- K-1. S.M.S. må i innloggingsperioden kunne hente opplysninger om den innloggede brukeren
- K-2. S.M.S. må kunne bruke de innhentede opplysningene om brukeren til å tilpasse tjenesten basert på disse opplysningene.
- K-3. Det må finnes en måte å identifisere opplysningstyper på tvers av S.M.S. og tjenestene som er tilknyttet S.M.S.
- K-4. Basert på vurderingen av de innhentede opplysningene om brukeren skal S.M.S. tilpasse informasjonen slik at bare tjenester som er relevante for brukeren vises.
- K-5. Opplysningene i S.M.S. må kunne knyttes til opplysningstyper i informasjonsmodellen i det transformerte regelverket
- K-6. S.M.S. må ha støtte for de operasjonene som er angitt som del av prosessmodellen
- K-7. Når brukeren velger en tjeneste skal skjema være forhåndsutfylt med data som allerede er kjent for Minside
- K-8. Brukeren må ha mulighet til å overskrive de forhåndsutfylte dataene
- K-9. Brukeren må få se hva som er kilden til de forhåndsutfylte dataene
- K-10. Brukeren skal kunne se hvorfor de tjenestene som er markert som relevante er vurdert som relevante
- K-11. Brukeren skal ha kunne velge en visning som viser hvilke tjenester som er skjult, inkludert (link til) informasjon om hvorfor.
- K-12. Brukeren skal ha mulighet til å deaktivere denne automatiske tilpasningen av informasjon dersom vedkommende ønsker det
- K-13. S.M.S. henting av opplysninger om brukeren må kun skje når brukeren er innlogget og har gitt sitt samtykke til det.
- K-14. Når brukeren er logget ut, skal ingen opplysninger om brukeren lenger være lagret.
- K-15. S.M.S. bør tilby en feedback-mekanisme hvor brukerne kan angi om tilpasningen av informasjon oppleves relevant.
- K-16. Kildekoden som inneholder transformert regelverk må være åpent tilgjengelig på nettet.
- K-17. Hver del av kildekode som inneholder transformert regelverk, må referere til den regelen som den implementerer.
- K-18. Referansen til regelverket må være så spesifikk som mulig. Som et minimum må den referere med URI til riktig paragraf i Lovdata, i tillegg til eventuelt fritekst for å spesifisere ytterligere.
- K-19. Kildekoden bør synliggjøre hvilke deler av Schartums transformasjonsmodell den implementerer.
- K-20. Det må være mulig å anvende regelsett fra private organisasjoner og interessegrupper (tredjepart) på opplysningene i Minside.
- K-21. Anvendelse av andres regelsett må kunne skje uten at personopplysningene i Minside flytter fra Minside til tredjeparten.
- K-22. Det må være mulig å relatere tjenester til hverandre

I neste kapittel lager vi en løsningsbeskrivelse for S.M.S. som skal oppfylle så mange av disse kravene som mulig.

7 Løsningsforslag og vurdering

7.1 Innledning

Dette kapittelet inneholder vårt løsningsforslag for en videreutvikling av Minside til å bli en tjeneste som tilpasser informasjonen til brukeren på bakgrunn av hvilke opplysninger det offentlige har om brukeren.

Løsningen baserer seg på bruk av Semantic Web-teknologi, og er derfor kalt ”Semantisk Minside”, forkortet S.M.S. I løsningsforslaget refererer vi til scenarioet om Peder og Marte, og vi forutsetter derfor at dette scenarioet er kjent for leseren. Forøvrig er løsningsbeskrivelsen ment å kunne leses som et selvstendig dokument.

Løsningsforslaget dekker ikke ikke-funksjonelle krav som for eksempel krav til ytelse, skalerbarhet og driftssikkerhet.¹⁷⁸

Vi tar utgangspunkt i dagens Minside-løsning, og beskriver hvordan Minside kan bygges ut med et arbeidsminne, hvor personopplysninger fra offentlige etater (heretter kalt tjenesteleverandører) samles når brukeren er pålogget. I tillegg til personopplysninger kan S.M.S. hente transformert regelverk fra forskjellige kilder. Ved hjelp av en slutningsmekanisme kan S.M.S. anvende regelverket på personopplysningene. Resultatene som slutningsmekanismen produserer, brukes til å utstyre lenkene til tjenesteleverandørens tjenester i S.M.S. med informasjon om hvor sannsynlig det er at brukeren har rettigheter knyttet til den tjenesten.

Som et eksempel skal S.M.S. kunne fastslå at det er stor sannsynlighet for at Peder og Marte har rett til bostøtte fra Husbanken, og lenken til Husbankens skjema for å søke bostøtte skal dermed markeres på en måte som synliggjør dette for Peder som er innlogget.

Muligheten til å hente transformert regelverk hos tjenesteleverandører og andre parter, som åpner for en distribuert modell for å vedlikeholde grunnlaget for den automatiserte rettslige veiledningen, er en sentral del av løsningsforslaget vårt. Utfordringen er å sikre at det transformerte regelverket foreligger på en slik måte at S.M.S. kan gjøre bruk av det. For å få til dette har vi gjort følgende valg:

- Det transformerte regelverket må følge standardene for Semantic Web. Ettersom det ikke er definert en standard for regelverk ennå, bruker vi regelformatet som benyttes i CWM – et lite verktøy for å eksperimentere med Semantic Web standarder (og kommende standarder)
- Det transformerte regelverket må være transformert i henhold til Schartums transformasjonsmetode, slik at det er mulig å identifisere rekkefølgen for utføring av reglene på en generell måte

En nærmere gjennomgang av hvordan S.M.S. kan hente og bruke transformert regelverk, og kravene til regelverket, gir vi i kapitlene 7.3-7.7. Det første viser i detalj hvordan regler som er transformert til CWM-format, kan anvendes på konkrete personopplysninger. Det andre ser nærmere på hvordan S.M.S. kan finne frem i regelverk generelt. For å demonstrere hvordan dette kan fungere lager vi en prototype for funksjonaliteten ved hjelp av regelverk og testdata fra scenarioet om Peder og Marte, samt verktøyet CWM.

2. S.M.S. ber om samtykke til å hente opplysninger om brukeren (personopplysninger) fra tjenesteleverandørene (TL) og lagre disse i arbeidsminnet så lenge brukeren har en aktiv sesjon
3. S.M.S. forespør tjenesteleverandørene om personopplysninger basert på brukerens fødselsnummer
4. S.M.S. henter transformert regelverk fra eget lager over transformert regelverk samt eventuelle tilgjengelige transformerte regelverk fra nettstedene til tjenesteleverandører og plasserer det i arbeidsminnet
5. S.M.S. henter transformert regelverk fra nettstedene til de tredjepartene/interesseorganisasjonene brukeren tidligere har valgt å lagre lenker til i sin brukerprofil, og plasserer det i arbeidsminnet
6. S.M.S. resonnerer ved hjelp av slutningsmekanismen, basert på personopplysningene og transformert regelverk som er tilgjengelige
7. S.M.S. tilpasser presentasjonen av informasjon i henhold til resultatene fra slutningsmekanismen
8. Brukeren ser gjennom informasjonen om hva og hvorfor tjenester er markert som relevante/ikke-relevante
9. Brukeren velger å endre/oppdatere opplysninger om seg selv
10. Arbeidsminnet oppdateres. Slutningsmekanismene resonnerer på nytt basert på oppdatert(e) opplysning(er), tilpasser informasjon på nytt – med markering av hva som er endret
11. Brukeren velger å endre/utvide regelverksgrunnlaget fra nye tredjeparter/interesseorganisasjoner
12. S.M.S. henter valgt regelverk – påpeker eventuelt avvik fra eksisterende, og slutningsmekanismen resonnerer på nytt, tilpasser informasjon på nytt og markerer hva som eventuelt er endret.
13. Brukeren velger å utføre en tjeneste (åpne et skjema)
14. Skjema åpnes
15. Opplysninger som er tilgjengelige i arbeidsminnet forhåndsutfylles
16. I det forhåndsutfylte skjema fremgår opplysningens opphav, om den er hentet fra tjenesteleverandør eller endret/lagt til av brukeren.
17. Brukeren fyller ut manglende opplysninger, skjemaet gir om mulig kontinuerlig tilbakemelding om antatt utfall
18. Brukeren trykker "send", og skjemaet sendes til mottakeren
19. Brukeren får kvittering i sin meldingsboks med referanse til saksnummer og saksbehandlingsregler
20. Bruker trykker "logg ut"
21. S.M.S.' arbeidsminne tømmes for alle personopplysninger og regler

Nedenfor er en gjennomgang av elementer og operasjoner fra bruksscenarioet over.

7.2.1.1 Brukeren

Brukeren tilsvare bruker i dagens Minside.

Brukeren er en sluttbruker som har rollen som innbygger, og som har behov for rettighetsinformasjon.¹⁷⁹ Brukeren representerer primær-målgruppen for S.M.S.

7.2.1.2 S.M.S.

S.M.S. står for "Semantisk Minside" og er betegnelsen på løsningen. For brukeren fremstår S.M.S. i samspill med de andre aktørene som ett system. S.M.S. betegner den delen av

systemet som har brukerens profil, slutningsmekanisme, arbeidsminne samt presentasjon av informasjon om og lenker til nettbaserte tjenester. S.M.S. bygger på Minside.

7.2.1.3 Den felles innloggingsløsningen

Innloggingsløsningen tilsvarende innloggingsløsningen i dagens Minside.

Det er tre mulige begrunnelser for at S.M.S. eventuelt kan trenge høyere sikkerhetsnivå enn dagens Minside:

- fordi flere etater velger å gjøre flere personopplysninger tilgjengelige, slik at omfanget av personopplysninger i seg selv krever høyere sikkerhetsnivå,
- fordi muligheten for å lagre pekere til foretrukne tredjeparters regelverk i brukerprofilen kan resultere i personopplysninger som oppfattes som sensitive,¹⁸⁰
- fordi tilgang til S.M.S. også kan gi tilgang til slutningsmekanismen og de personopplysningene som genereres når slutningsmekanismen anvender regler på de innhentete personopplysningene

7.2.1.4 Samtykke

Tilsvarende i hovedsak samtykket som Minside i dag baserer sin databehandling på. Brukeren må på samme måte som med dagens Minside akseptere at det blir opprettet en brukerprofil hvor brukerens fødselsnummer brukes som identifikator. I tillegg må brukeren gi samtykke til at Minside (og tilsvarende S.M.S.) kan forespørre tjenesteleverandørene om personopplysninger knyttet til brukeren. Forskjellene er:

- I brukerprofilen i S.M.S. kan det lagres hvilke regelverk fra tredjeparter/interesseorganisasjoner brukeren ønsker å benytte
- I S.M.S. blir personopplysningene fra tjenesteleverandørene lagret i et arbeidsminne for prosessering basert på transformert regelverk, slik at det kan gis automatisert veiledning. I dagens Minside blir personopplysningen bare *vist*.

7.2.1.5 Personopplysninger

S.M.S. henter, behandler og skaper personopplysninger på flere måter.

S.M.S. forespør tjenesteleverandørene på samme måte som dagens Minside etter opplysninger som tjenesteleverandøren har om en bruker. Dette foregår gjennom registertjenestene. Forskjellen i S.M.S. er at svaret fra tjenesteleverandørene vil være i form av RDF-data. Dersom etaten har harmonisert sine datamodeller gjennom ”Semantikkregisteret for elektronisk samhandling”, SERES, bør dette være grunnlaget for RDF-dataene.¹⁸¹ Dette innebærer at de er utstyrt med kraftigere semantikk.

S.M.S. behandler innhentede personopplysninger i henhold til importert, transformert regelverk, ved hjelp av en slutningsmekanisme. Dette resulterer i nye, avledede personopplysninger.

I tillegg til det innsynet brukeren får i opplysningene om seg selv i dagens Minside, kan S.M.S. gi brukeren mer informasjon om de enkelte personopplysningene – for eksempel gjennom å synliggjøre hvor de benyttes i et regelverk, hvilke vedtak de er grunnlag for etc. På den måten kan S.M.S. bidra til å bedre forståelsen av opplysningene og med det forbedre innsynet for brukeren. Dette forsterker det personvernøkende elementet som innsynsmulighetene i dagens Minside har.

I tillegg til personopplysningene som S.M.S. får fra tjenesteleverandørene og de opplysningene som slutningsmekanismen genererer, vil det også oppstå personopplysninger i forbindelse med bruken av S.M.S., både i brukerens profil og gjennom eventuell logging av brukerens aktivitet.

7.2.1.6 Tjenesteleverandørene

Tjenesteleverandørene tilsvarer begrepet tjenesteleverandører i dagens Minside, og er offentlige etater som tilbyr tjenester – inkludert registertjenester som gir tilgang til personopplysninger.

7.2.1.7 Aktiv sesjon

Når brukeren logger seg inn på S.M.S. etableres det en brukersesjon. Denne varer helt til brukeren logger seg ut, eventuelt til S.M.S. avslutter sesjonen etter en viss tid med inaktivitet. Dette tilsvarer dagens Minside. Opplysninger om brukeren i S.M.S.' "arbeidsminne" slettes når brukeren ikke lenger har en aktiv sesjon.

Opplysninger som brukeren ønsker å lagre mellom hver sesjon, som oppdaterte personopplysninger eller lenker til tredjeparters regelverk, kan lagres i brukerens profil.

7.2.1.8 Fødselsnummer

S.M.S. bruker fødselsnummer på samme måte som dagens Minside, som en unik identifikator for å identifisere en person på tvers av systemer (Minside og tjenesteleverandørene). Bare innbyggere med fødselsnummer eller D-nummer kan logge seg inn på Minside.

7.2.1.9 Transformert regelverk

Transformert regelverk er regelverk som er fortolket og oversatt til et format som er forståelig for S.M.S.' slutningsmekanisme. Regelverket kan være en kombinasjon av en ontologi og konkrete regler som gjør det mulig å utlede eller fastsette nye opplysninger basert på eksisterende opplysninger.

S.M.S. bør støtte en fremtidig standard for utvekslingsformat for slike regler som kommer gjennom arbeidet med "Rule Interchange Format". Vi tar utgangspunkt i det regelformatet som støttes av det verktøyet vi har brukt for å prototype funksjonalitet knyttet til regler, det vil si CWM.

I det transformerte regelverket angis opplysningstypene ved hjelp av RDF inkludert bruk av navnerom, slik at det er mulig å koble opplysningene i arbeidsminnet med opplysningstypene som brukes i regelverket. Dette bør baseres på harmonisering av datamodeller på tvers av etatene, f.eks. gjennom SERES.

7.2.1.10 S.M.S.' eget lager for transformert regelverk

Forvaltningsansvarlig for S.M.S. kan sørge for å transformere (deler av) regelverk på eget initiativ, og gjøre dette tilgjengelig for S.M.S. Dette lageret bør være åpent, slik at andre kan (gjen)bruke det og kontrollere innholdet i det, med andre ord bør det ligge tilgjengelig på en åpen del av nettstedet til S.M.S., eventuelt nettstedet til etaten med forvaltningsansvaret.

7.2.1.11 Tredjeparter/interesserorganisasjoner

I rettighetsspørsmål er forvaltningen part i saken, og det er uheldig om bare forvaltningens fortolkning av regelverk skal være tilgjengelig for brukerne i form av transformert regelverk.

Derfor bør deres fortolkning kunne utfordres og/eller suppleres med fortolkninger fra tredjeparter.

Noen eksempler på tredjeparter kan være en interesseorganisasjon for en spesiell befolkningsgruppe (trygdede, pasienter), arbeidstakerorganisasjoner, Skattebetalerforeningen etc. Politiske partier kan også tenkes å ville gjøre bruk av den samme mekanismen for å hjelpe brukeren med å simulere hva konsekvensen av deres politikk ville være.

Betalte rådgivere som f.eks. advokater, kan også benytte den samme funksjonaliteten. I vår løsningsbeskrivelse er det forutsatt at det transformerte regelverket ligger åpent på et nettsted. Betalte rådgivere vil muligens foretrekke å kunne benytte mekanismen uten å måtte gi alle gratis tilgang til den kunnskapen som ligger i transformasjonsarbeidet. Det kan i så fall løses ved å bygge ut mulighet for å importere regelverk med tilgangskontroll, men det er det ikke tatt høyde for her.

7.2.1.12 Transformert regelverk på nettstedet til tjenesteleverandører og tredjeparter/interesseorganisasjoner

Transformert regelverk fra tjenesteleverandører og tredjeparter/interesseorganisasjoner (og S.M.S. selv) må ligge åpent tilgjengelig på den ansvarliges nettsted i en form for tekstlig representasjon av reglene.

På samme måte som tjenesteleverandørene melder inn tjenester til forvaltningsorganisasjonen for Minside, må de også opplyse om hvor, det vil si i form av en URL til deres nettsted, etatens transformerte regelverk kan hentes fra.

Eieren av det transformerte regelverket bør legge inn mekanismer for å sikre at det ikke kan endres, for eksempel gjennom "sjekksummer"¹⁸² eller andre former for signeringsløsninger. Dette går vi ikke nærmere inn på.

7.2.1.13 Arbeidsminne

Arbeidsminne i S.M.S. er et RDF-lager der alle personopplysningene, alle tilhørende datamodeller/ontologier og alle regler som hentes inn, plasseres. Slutningsmekanismen kan interagere med dette datagrunnlaget ved å forespørre enkelt-tripler eller sett av tripler. Personopplysninger som utledes ved hjelp av slutningsmekanismen plasseres også her.

Arbeidsminnet er knyttet til en brukersesjon. Opplysninger i arbeidsminnet knyttet til en brukersesjon er ikke tilgjengelige for andre brukere.

Arbeidsminnet tømmes når brukerens sesjon avsluttes ved utlogging.

7.2.1.14 Brukerprofil

I likhet med dagens Minside blir det opprettet en brukerprofil første gang man logger seg inn i S.M.S, og samtykker til S.M.S.' behandling av personopplysninger. I denne profilen lagres fødselsnummer, navn, bostedskommune og språkvalg. I S.M.S. vil det i tillegg være mulig å legge til pekere til transformert regelverk hos tredjeparter som brukeren har valgt å benytte istedenfor eller i tillegg til det regelverket som tilbys via tjenesteleverandører og S.M.S. selv.

S.M.S. skal kunne markere endringer i informasjonen som vises dersom brukeren enten endrer personopplysninger eller bytter/utvider det transformerte regelverket som den

automatiserte tilpasningen av informasjon baserer seg på. For å kunne få til det må det lagres et ”bilde” av før-situasjonen. Dette bildet bør kunne lagres fra en sesjon til en annen, slik at endringer i informasjonen som skyldes endringer i personopplysningen mellom to besøk på S.M.S. – for eksempel dersom en bruker i mellomtiden har fått barn – kan markeres spesielt.

Dersom brukeren velger å endre noen av personopplysningene som er hentet inn, f.eks. inntekt dersom vedkommende har gått vesentlig opp eller ned i forhold til det tallet S.M.S. får fra Skatteetaten, kan brukeren be om at endringen lagres i brukerprofilen. På den måten slipper brukeren å gjøre den samme endringen hver gang vedkommende logger inn på S.M.S.

Det er en vurdering vi ikke har konkludert med her, om lagring av de to typene opplysninger som beskrevet over (øyeblikksbilde av informasjonen på S.M.S. og endrede verdier på personopplysninger) kan gjøres med et sikkerhetsnivå tilsvarende dagens Minside, eller om verdiene i så fall bør krypteres eller sikres på andre måter.

7.2.1.15 Resonnerer

Med ”resonnerer” mener vi her bruk av en slutningsmekanisme (”inference engine”) som kan utlede implisitte opplysninger fra eksisterende opplysninger, og anvende regler på informasjonen.

7.2.1.16 S.M.S.’ slutningsmekanisme

S.M.S. slutningsmekanisme er den delen av S.M.S. som kan resonnerer over opplysninger. Slutningsmekanismen må støtte W3Cs standarder for data og regler. Vi har ikke tatt stilling til hva slags form for logikk som er minimum for slutningsmekanismen.¹⁸³

7.2.1.17 Tilpasset presentasjon av informasjon

Tilpasset presentasjon av informasjonen i S.M.S. er kjernefunksjonaliteten i løsningen. Basert på transformert regelverk skal S.M.S. kunne presentere lenker til tjenester på en måte som synliggjør hvor relevant tjenesten er – i betydningen om det er en tjeneste brukeren har rett til eller ei. Fremheving av relevante tjenester og nedtoning av ikke-relevante tjenester vil gjøre det lettere for brukeren å gjøre krav på sine rettigheter.

7.2.1.18 Tjenester markert som relevante/ikke-relevante

Dette kan eksemplifiseres med dagens Minside, og tjenesten ”Informasjon om internasjonal tjeneste”. Via denne tjenesten kan brukeren melde sin interesse for utenlandstjeneste i regi av Forsvaret.¹⁸⁴ På Forsvarets informasjonsside om internasjonal tjeneste¹⁸⁵ finner vi mer informasjon om krav til søkere, og et av kravene er at søkeren er over 18 år.¹⁸⁶ Det er med andre ord en tjeneste som ikke er relevant – i den forstand at det er noe poeng i å søke – for yngre brukere.

Samtidig er det en tjeneste som det kan være verdifullt for yngre personer å vite om, med tanke på å kunne vurdere hva de ønsker å gjøre på et senere tidspunkt i livet, og hvordan de kan forberede seg. Dette understreker at tjenester ikke bør *skjules*, men derimot markeres på en måte som gir brukeren informasjon om årsaken. Brukeren skal selv kunne styre visningen og eventuell filtrering slik at ikke-relevante tjenester ikke vises.

Ikke-relevante tjenester kan deles i to grupper. De som *kan bli* relevante og de som *ikke kan bli* relevante. For en som er 16 år er utenlandstjeneste en tjeneste som *vil kunne bli* relevant senest om to år. Dersom det finnes en øvre aldersgrense for utenlandstjeneste, og en bruker er

over den aldersgrensen, vil tjenesten *aldri bli* relevant for denne brukeren slik vi definerer relevans. Dette skal også synliggjøres.

En god utforming av slik synliggjøring som nevnt her forutsetter brukertesting for å forsikre seg om at informasjonen som formidles blir oppfattet i tråd med intensjonen, og at brukerne forstår eventuelle filtreringsmekanismer. I denne løsningsbeskrivelsen nøyer vi oss med å påpeke prinsippet for synliggjøringen.

Grunnlaget for å vurdere relevansen er resultatet av testingen av inngangskrav for de forskjellige partskategoriene i det eller de regelverkene som er knyttet til en tjeneste. For en partskategori vil for eksempel tre av fire inngangskrav være oppfylt. Se eksempel på denne typen resultater i kapittel 7.4 og 7.5.

7.2.1.19 Brukeren endrer/oppdaterer opplysninger

Personopplysningene S.M.S. får fra tjenesteleverandørene kan være gale eller utdaterte. For at S.M.S. skal kunne foreta automatisert veiledning basert på opplysninger om innbyggeren kan det være aktuelt for brukeren å endre opplysninger. Dette vil også gi mulighet til å bruke S.M.S. for å simulere alternativer.

Når en bruker velger å endre en opplysning må det gå klart frem at endringen ikke innebærer en oppdatering av kilden for opplysningene. Dersom det skal skje, må brukeren sende en dertil egnet forespørsel til etaten opplysningene kommer fra, for eksempel ved bruk av den eksisterende meldingstjenesten i Minside.¹⁸⁷ Videre må brukeren tilbys valg for å lagre den endrede opplysningen i brukerprofilen. Brukeren vil i så fall, ved neste innlogging, bli spurt om vedkommende ønsker veiledning basert på opplysningene brukeren har endret, eller opplysningene de får fra kilden.

7.2.1.20 S.M.S. tilpasser informasjon på nytt

Når det skjer endringer i datagrunnlaget i arbeidsminnet, enten ved endringer i personopplysningene eller ved tillegg eller endring i regelverksgrunnlaget, vil S.M.S.' slutningsmekanisme aktiveres på nytt, og alle regler prosesseres på nytt for å se om endringene genererer nye resultater.

7.2.1.21 S.M.S. markerer hva som er endret

S.M.S. må synliggjøre for brukeren hvilke endringer som har skjedd mellom hver gang informasjonen er endret som følge av at slutningsmekanismen har re-prosessert reglene, for eksempel fordi det har skjedd endringer i datagrunnlaget eller i regelverket. Dette gjelder også mellom to innlogginger (se 7.2.1.14 om brukerprofil over).

7.2.1.22 Brukeren endrer/utvider regelverksgrunnlaget

Dersom brukeren ønsker det kan vedkommende supplere eller erstatte regelverksgrunnlaget som S.M.S. bruker ved å hente regelverk fra foretrukne tredjeparter, for eksempel en arbeidstagerorganisasjon eller pasientorganisasjon. Opplysningen om hvilke regelverk brukeren har valgt å hente og bruke må betraktes som en potensielt sensitiv personopplysning, og det er derfor viktig at S.M.S. kan hente regelverket uten at det knyttes til brukerens identitet. Muligheten til å "pseudonymisere" kilden for regelverket er også en forutsetning for å kunne lagre en peker til regelverket i brukerprofilen for fremtidig bruk.

Når regelverksgrunnlaget er lagt til må slutningsmekanismen først brukes til å identifisere eventuelt avvik fra eksisterende regelverk slik at det kan synliggjøres for brukeren. Deretter kan brukeren be slutningsmekanismen gjøre en ny prosessering av alle regler i regelverksegrunnlaget. S.M.S. må deretter på samme måte som nevnt over, markere eventuelle endringer i informasjonen på siden.

7.2.1.23 Utføre en tjeneste

Når brukeren velger en lenke til en tjeneste hos en tjenesteleverandør vil det åpnes et skjema hos tjenesteleverandøren. S.M.S. kan tilby brukeren å forhåndsutfylle feltene med opplysninger fra arbeidsminnet forutsatt at skjemafeltene hos tjenesteleverandøren identifiserer opplysningstypen på samme måte som personopplysningene som ligger i S.M.S.' arbeidsminne (RDF-data og navnerom).

I det forhåndsutfylte skjemaet fremgår opplysningens opphav, om den er hentet fra tjenesteleverandør eller endret/lagt til av brukeren. Brukeren fyller ut manglende opplysninger, og overskriver eventuelt eksisterende opplysninger. Skjemaet gir om mulig kontinuerlig tilbakemelding om antatt utfall (simulering). Brukeren trykker "send", og skjemaet sendes til mottakeren. Brukeren får kvittering i sin meldingsboks med referanse til saksnummer og saksbehandlingsregler.

7.2.1.24 Logg ut

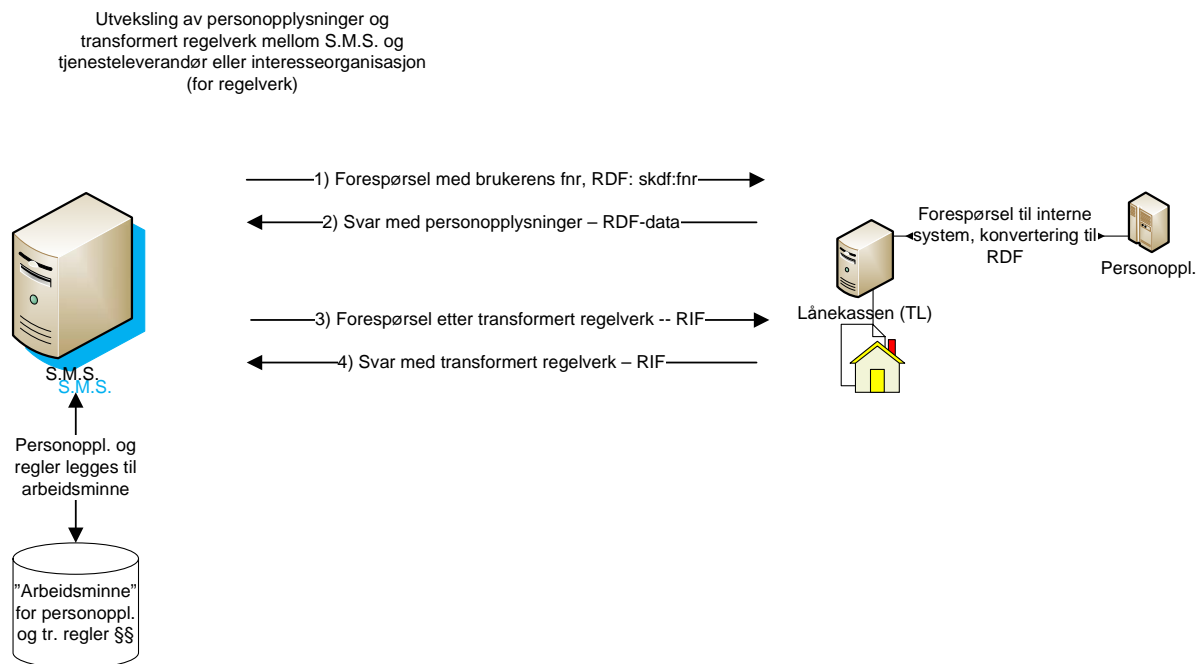
Brukeren kan logge ut ved å velge "logg ut" enten på S.M.S. eller på en av de tilknyttete tjenesteleverandørenes tjenester (single logout). Dersom brukeren er inaktiv over en viss tid, skal brukeren bli logget ut automatisk.

7.2.1.25 Arbeidsminnet tømmes

Når brukeren logger ut (eller blir logget ut på grunn av for lang inaktivitet) skal alle opplysninger i arbeidsminnet tømmes.

7.2.2 Utveksling av personopplysninger og regelverk mellom S.M.S. og andre aktører

Skissen nedenfor viser hvordan S.M.S. kan forespørre tjenesteleverandører etter 1) personopplysninger med referanse til en persons fødselsnummer. Både forespørsel og svar er i form av RDF-data, representert som tekst gjennom RDF/XML-formatet.



Figur 16 -- Detaljering av utveksling av personopplysninger og/eller regelverk mellom S.M.S. og tjenesteleverandører/tredjeparter

7.3 S.M.S.-prototyp basert på CWM – oversikt

Over har vi gått igjennom S.M.S. på et overordnet nivå. I dette og de påfølgende kapitlene skal vi se nærmere på funksjonaliteten knyttet til automatisert rettslig veiledning. Hensikten med kapitlet er å dokumentere at Semantic Web-teknologi gir oss de mulighetene som er nødvendige for at S.M.S. skal kunne fungere. Vi har kalt dette en prototyp, fordi den demonstrerer hvordan vi kan bruke Semantic Web-teknologi for gi datamaskinene *kunnskap* slik vi har definert det i kapittel 3. Prototypen består fra vår side i at vi definerer et sett med testdata i RDF-format, og et sett med maskinforståelige regler basert på regelverk vi har transformert. Dette bruker vi til å kontrollere om Semantic Web vil være i stand til å anvende reglene vi definerer på de dataene vi har.

Vi har ikke programmert selve slutningsmekanismen som utfører reglene. Her bruker vi isteden verktøyet CWM, som kan sees på som et Semantic Web i miniatyr; det kan håndtere RDF-data (på både N3 og RDF/XML-format), og har en slutningsmekanisme som kan "resonnere" over dataene, blant annet ved hjelp av regler. I eksemplene viser vi konkret hvordan vi har brukt CWM. Dette er gjort for å åpne for at andre skal kunne etterprøve resultatene. Men det er også gjort med et håp om at det vil gi en nyttig utvidelse av den tilgjengelige dokumentasjonen av CWM. For en kort innføring i hvordan CWM brukes, viser vi til vedlegg.¹⁸⁸

Det viktige for oss å vise er at det finnes måter å transformere regelverk til datamaskinspråk på, som gjør det mulig å hente inn transformerte regler fra flere kilder, og anvende dem på – og utvide – det settet av personopplysninger som ligger i S.M.S. arbeidsminne for brukeren.

Når det gjelder brukergrensesnitt har vi nøydt oss med å antyde hva slags informasjon som bør presenteres for brukeren, og hvilke mekanismer som kan brukes for å vurdere relevansen av tjenester som S.M.S. lenker til. Forøvrig konstaterer vi at slutningsmekanismen i S.M.S. genererer nye opplysninger om en bruker som plasseres i arbeidsminnet, og at det i dag ikke

er mangel på kunnskap, teknologier eller metoder for hvordan man kan lage nettsider som presenterer opplysninger fra et datalager for en bruker.¹⁸⁹

Når vi demonstrerer prototypen med eksempelreglene, konsentrerer vi oss om å demonstrere den kunnskapen som vi formulerer ved hjelp av regler på det formatet som CWM forstår – og som forutsetter at personopplysningene er tilgjengelige som RDF-data med eksplisitt angitte opplysningstyper (gjennom bruk av navnerom). S.M.S.-løsningen forutsetter imidlertid et nivå av kunnskap som angir den generelle fremgangsmåten som skal benyttes for hvert regelverk. Dette ser vi nærmere på i kapittel 7.7, etter at vi har sett på de konkrete eksemplene. Etter å ha demonstrert muligheter og utfordringer ved hjelp av to konkrete regelverk, for henholdsvis rentefritak og bostøtte, løfter vi med andre ord blikket opp på den generelle delen av løsningen.

7.4 Eksempel – automatisert veiledning basert på Lånekassens regler for rentefritak

I dette kapittelet skal vi ved hjelp av CWM prototype den sentrale delen av veiledningsfunksjonen i S.M.S., nemlig muligheten til å anvende et transformert regelverk hentet fra en tjenesteleverandør på et konkret regelverk og for en konkret bruker, Peder Ås. Vi bruker Lånekassens regelverk for rentefritak, og konsentrerer oss om den delen som gjelder for Peder og Marte. I neste kapittel ser vi på Husbankens regelverk for bostøtte, og konsentrerer oss om forskjellene mellom dette og Lånekassens regelverk.

Vi understreker at transformeringen vi har gjort er basert på en modifisert rettsdogmatisk metode.¹⁹⁰ Formålet er her å illustrere hvordan S.M.S. kan håndtere regelverk, og vi har forsøkt å balansere bruken av regler mellom slik at det på den ene siden er så enkelt som mulig med tanke på å forenkle fremstillingen, samtidig som det på den annen side er realistisk nok til at erfaringene vi gjør fortsatt er relevante.

Som vi har sagt innledningsvis forutsetter S.M.S. at regelverk som S.M.S. skal gi automatisert veiledning innenfor, må være transformert på en standardisert måte, både i form av det konkrete formatet på reglene, og i form av struktur. Strukturen er basert på Schartums transformasjonsmetode. Første del av eksempelet består derfor i å transformere de aktuelle reglene i tråd med denne metoden. Deretter går vi videre til å etablere en informasjons- og prosessmodell, før vi tester de transformerte reglene i praksis.

7.4.1 Sortering av regelverket

Første del av transformasjonsmetoden er å sortere regelverket etter partskategori og etter regelkategori.

Første steg er å identifisere en partskategori. Vi ønsker å identifisere en partskategori som i størst mulig grad identifiserer et unikt sett av materielle inngangskriterier. I Lånekassens rentefritaksregler i forskrift om forrentning og tilbakebetaling av utdanningslån og tap av rettigheter¹⁹¹ finnes det flere hovedregler for rentefritak, som delvis er egnet til å identifisere partskategori. Disse gjenspeiles i titlene på paragrafene:¹⁹²

§ 12-1. Rett til rentefritak av sosiale og økonomiske årsaker

§ 12-2. Rett til rentefritak ved økonomisk stønad etter sosialtjenesteloven

§ 12-3. Rett til rentefritak ved fulltidsarbeid og lav bruttoinntekt

§ 12-4. Rett til rentefritak ved soning og lav bruttoinntekt

”Vår” partskategori er her de som har rett til rentefritak ”av sosiale og økonomiske årsaker”, jfr § 12-1. Men denne paragrafen lister selv opp forskjellige tilfeller, som også er aktuelle som partskategorier. Valg av partskategori er i henhold til transformasjonsmetoden et pragmatisk valg, der det kan være en vurdering om man ønsker få kategorier og flere inngangskrav, eller flere kategorier. Valget styres av formålet, som er å få sortert reglene på en hensiktsmessig måte. For S.M.S. er det derimot viktigere å finne et strengere skille, slik at partskategori identifiserer ett sett regler som er slik at *alle* reglene gjelder for denne partskategorien.

Hvis vi ser for oss at vi velger partskategorien ”rett til rentefritak av sosiale og økonomiske årsaker”, kan vi sette opp titlene på bokstavene a-d for henholdsvis sykdom og arbeidsledighet, verneplikt/militær skole, fødsel eller adopsjon, og omsorgsarbeid, som materielle inngangskriterier. Under hver av disse finner vi flere regler, som hver bare er aktuelle for den regelen den sorterer under.

For å gjenspeile det har vi heller valgt å operere med flere nivåer av partskategorier. Dermed følger det også at ”felles regler” kan være felles for et gitt nivå av partskategorier. Tabellstrukturen som er foreslått i Schartum (2005:60) har kun ett nivå.

Vi har heller ikke tatt med formelle inngangskrav eller innholdsbeskrivelser. Formelle inngangskrav gjelder regler som bare kommer til anvendelse ved en faktisk søknad, som krav til format og dokumentasjon. De er med andre ord ikke relevante i en veiledningssituasjon. Innholdsbeskrivelser gir regler for hva en avgjørelse i siste omgang vil inneholde, for eksempel utregningsregler. Disse kan være en viktig del av en simulator-tjeneste, men vi begrenser oss til veiledningsdelen, og ser bort fra dem her.

Resultatet er satt opp i tabellen nedenfor. I tillegg til regelverket som gjelder for Peder og Marte – det vil si reglene om fødsel – har vi tatt med de ”omkringliggende partskategoriene”, for bedre å få frem sorteringen. Disse reglene er satt med grå bakgrunn. De reglene vi skal gå videre med har vi navngitt for lettere å kunne referere til dem senere.¹⁹³

Tabell 1 -- Sortering av regelverket for rentefritak i partskategorier og materielle inngangskrav

Partskategori1	Partskategori2	Partskategori3	Materielle inngangskrav	Ref.	
<i>Felles regler</i>					
PK1: sosiale eller økonomiske årsaker	<i>Felles regler</i>		PK1-FM1: en brutto månedsinntekt på kr 14.980 eller lavere, har rett til rentefritak. Inntektsgrensen blir økt med kr 1.500 for hvert barn under 16 år som låntakeren forsørger.	§ 12-1, 1. ledd	
	sykdom og arbeidsledighet	<i>Felles regler</i>			
		[...]			
	verneplikt/militær skole	<i>Felles regler</i>			
		[...]			
	PK2: Fødsel eller adopsjon	<i>Felles regler</i>			
		PK3: Fødsel		M1: Låntaker som dokumenterer fødsel og omsorg etter fødsel	§ 12-1, c
			M2: Den av foreldrene som har omsorgen for barnet, M3: får rentefritak i inntil tolv måneder fra fødselen		
Adopsjon					
Omsorgsarbeid	<i>Felles regler</i>				
	[...]				
økonomisk stønad etter sosialtjenesteloven					

7.4.2 Etablering av rettslig informasjonsmodell

Neste steg i transformasjonsmetoden er å etablere en rettslig informasjonsmodell med oversikt over opplysningstypene i regelverket som vi har sortert i tabellen over.

For partskategoriene ”sosiale eller økonomiske årsaker” og ”fødsel eller adopsjon” har vi ikke identifisert noen egne opplysningstyper. Disse kategoriene blir isteden definert gjennom det regelverket som gjelder for partskategoriene. Det vil med andre ord være varierende om det til en konkret partskategori hører en faktisk operasjoniserbar regel eller ei, eller om partskategorien kun fungerer som ”merkelapp”, en ren sorteringsmekanisme, for et sett regler som definerer én mulig vei gjennom regelverket som definerer en rettighet.

Basert på den delen av regelverket som gjelder for partskategori ”Fødsel” (nivå 3), har vi laget informasjonsmodellen nedenfor. Vi har identifisert opplysningstypene, og dessuten sett på hvordan hver opplysningstype eventuelt kan representeres ved hjelp av personopplysninger som S.M.S. kan antas å ha tilgang til om en bruker, basert på hva vi vet om opplysningene som er tilgjengelige for Minside gjennom dagens registertjenester. I tabellen har vi også tatt med betegnelsen på opplysningen som RDF-data, slik vi vil bruke den senere.

Vi minner om at vår fortolkning av reglene med tanke på hvilke opplysninger vi kan bruke for å representere opplysningstypene i regelverket, er gjort for å eksemplifisere. For å sikre rettsriktige fortolkninger må man anvende en fullverdig rettsdogmatisk metode for å avdekke og løse alle tolknings spørsmålene. I et reelt systemutviklingsarbeid ville man trukket inn

jurister og saksbehandlere for å konkretisere hvilke opplysninger regelverket krever. Et eksempel på vår forenkling er opplysningstypen "brutto månedsinntekt". Inntektsbegrepet er et av de mest brukte eksemplene på begreper med forskjellig innhold i forskjellige sammenhenger - siktes det til brutto- eller nettoinntekt, alminnelig inntekt eller personinntekt? Vårt valg, i denne situasjonen, er å ta et inntektstall fra likningen som gjelder for et år, og dele dette på 12 måneder. Dette er ikke nødvendigvis korrekt, men for vår fremstilling er det tilstrekkelig å vise at vi kan "fylle" begrepet inntekt med innhold.

Selv om vår tolkning av inntektsbegrepet ikke nødvendigvis er korrekt, er det en god illustrasjon på en opplysningstype som det er mulig å finne automatiserbare fortolkninger av. Dette er i kontrast til opplysningstyper som er *vage*.¹⁹⁴ Det kan med hensikt være lagt inn vaghet i regelverket fordi det ikke på forhånd er mulig for lovgiver å si hva som vil være et riktig resultat - i betydningen rettferdig eller rimelig.¹⁹⁵ I våre eksempler har vi ingen klare(!) tilfeller av vaghet. Dersom vi ville hatt det, og vi ikke mente det var forsvarlig å automatisere vagheten ved tilnærminger gjennom en eller flere andre opplysninger, så ville resultatet bli at S.M.S. ikke kan bekrefte om den aktuelle regelen er oppfylt eller ei. Det i seg selv kan være en nyttig opplysning, og gi innspill til innbyggeren om at dette må inn til videre saksbehandling eller lignende.

Regel	Kommentar	Opplysningstype	RDF-data
M1: Låntaker som dokumenterer fødsel [og omsorg etter fødsel]	Regelen angir et formelt inngangskrav (krav til dokumentasjon), men i kravet fremgår to opplysningstyper, <i>låntaker</i> , som vi setter lik en som har kundenummer i lånekassen, og <i>fødsel</i> , som vi representerer ved barnets fødselsdato. Vi bruker fødselsdato fra folkeregisteret. "Omsorg etter fødsel" håndteres nedenfor.	Låntaker Barnets_fødselsdato	lanekassen: kundenummer skdf:fdato
M2: Den av foreldrene som har omsorgen for barnet	Opplysningstypen <i>forelder</i> er en person som har et barn. Vi bruker opplysning om foreldre-barn-relasjon fra folkeregisteret. Opplysningstypen <i>omsorgen for barnet</i> er et ja/nei-spørsmål. ¹⁹⁶ Dette er en opplysning som det antagelig bare vil finnes indikasjoner på i S.M.S. om er oppfylt eller ei. Opplysning om at en bruker har et nyfødt barn er i seg selv en indikasjon. En opplysning fra NAV om at vedkommende mottar fødselspenger, vil understreke dette.	Forelder Har_omsorgen	skdf:barn rentefritak: harOmsorgenFor
M3: Inntil 12 måneder etter fødselen	Opplysningstypen <i>inntil 12 måneder etter fødselen</i> er tall for tiden som er gått. Istedenfor å lage en egen opplysning for denne bruker vi en test for hvorvidt dagens dato minus et år er mindre enn <i>Barnets_fødselsdato</i> (over)	Dagens_dato	sms:today
PK1-FM1: har en brutto månedsinntekt på kr 14.980 eller lavere. Inntektsgrensen blir økt med kr 1.500 for hvert barn under 16 år som låntakeren forsørger.	Opplysningstypen "brutto månedsinntekt" er et tall. Vi bruker likningstall fra Skatteetaten for alminnelig inntekt, og deler på 12. <i>Inntektsgrense</i> er en variabel verdi (økes pr barn). Vi legger denne i <i>max_inntekt</i> . <i>Antall barn under 16 år</i> er en opplysningstype som angis som et tall, og som vi bruker opplysningene fra folkeregisteret til å fastslå.	Brutto_måneds- inntekt inntektsgrense Antall_barn_under_ 16	rentefritak: brutto_maaned- inntekt rentefritak: max_inntekt

7.4.3 Etablering av rettslig prosessmodell

Neste steg er å lage prosessmodellen. For det enkelte regelverket må tjenesteleverandørene – eller eventuelle tredjeparter eller S.M.S.'s forvaltningsorganisasjon – detaljere de konkrete operasjonene som beskrives i de enkelte reglene. Operasjonene må nødvendigvis beskrives som operasjoner på de opplysningstypene vi har identifisert i den rettslige informasjonsmodellen over.

I Lånekasseregulverket har vi reglene og de opplysningstypene som gjelder for hvert nivå av regelkategorier og de materielle inngangskriteriene. For hver konkrete regel har vi beskrevet nødvendige operasjoner i en form for pseudokode.¹⁹⁷ En slik prosessmodell vil utgjøre en viktig del av spesifikasjonen som programmereren bruker for å fullføre transformeringen til det maskinlesbare språket, som for eksempel COBOL eller Java.¹⁹⁸ I S.M.S. ønsker vi å benytte Semantic Web-teknologi i så stor grad som mulig istedenfor tradisjonelle programmeringsspråk. Vi har derfor utvidet prosessmodellen med regler på det formatet som vårt Semantic Web-verktøy, CWM, forstår. For enkelte av CWM-reglene har vi gjort forenklinger. Disse er angitt ved hakeparenteser, "[forenkling]".

Merk at vi formulerer alle reglene slik at de skal generere ja- eller nei-svar. Om nødvendig lagrer vi et beregnet resultat i en midlertidig variabel, før vi legger til en ny test på denne variabelen, som kan besvares med ja eller nei.

Merk også at vi ikke sorterer reglene etter type operasjon (logisk, aritmetisk eller dataauthenting) slik det anbefales i Schartums transformasjonsmetode. I vårt tilfelle er utvalget av regler såpass lite, samtidig som formålet ikke er å gi en fullstendig, rettsriktig modell, at en ekstra sorteringsmekanisme muligens ville gitt mindre, snarere enn mer, oversikt. Behovet for å identifisere hvilke prosesser som er aritmetiske har også noe å gjøre med hvordan vi skal lagre resultatet. I de fleste systemer i dag brukes det en relasjonsdatabase for å ta vare på opplysninger. Dette er i praksis et sett med tabeller. Dersom en ny opplysning skal lagres må det sikres at det finnes en tabell som har en kolonne avsatt for den nye opplysningstypen. Dette må gjøres på forhånd. For RDF-data derimot, er det ikke behov for å definere en slik tabellstruktur på forhånd. En hver ny opplysning er en ny trippel, som kan legges til når som helst.¹⁹⁹

Bruken av prefiks for å angi navnerom for verdiene er basert på følgende prinsipper:

- Opplysningstyper som representerer personopplysninger tjenesteleverandøren gjør tilgjengelig for S.M.S.-brukerne, har prefiks tilsvarende etatens navn. Eksempel: laanekassen:kundenummer
- Opplysningstyper som er hentet fra et regelverk en tjenesteleverandør har transformert, og som *ikke* har en identisk opplysningstype blant personopplysningene nevnt i punktet over, får prefiks som indikerer navnet på regelverket. Eksempel: rentefritak:brutto_maanedsinntekt. Som URI for navnerom for regelverk bruker vi referanse til Lovdata, ettersom det er stabile URLer.
- Opplysningstyper som representerer opplysninger som er interne for S.M.S. har prefiks sms. Eksempel: sms:resultat (som angir resultatet av prøvingen av en regel).
- I tillegg brukes prefikset math: for navnerommet <http://www.w3.org/2000/10/swap/math#> som gir tilgang til de aritmetiske operasjonene i CWM.²⁰⁰

Steg	Operasjoner (pseudokode)	Transformert regel (CWM-format)
PK1: Sosiale eller økonomiske årsaker		

Steg	Operasjoner (pseudokode)	Transformert regel (CWM-format)
PK2: Fødsel eller adopsjon		
PK3: Fødsel		
M1: Låntaker som dokumenterer fødsel	Bruker (x) har kundenummer (y) i Lånekassen OG Bruker (x) har barn (z) → ja/nei	{ ?x lanekassen:kundenummer ?y . ?x skdf:barn ?z . } => { rentefritak:M1 sms:resultat "true" } .
M2: Den av foreldrene som har omsorgen for barnet	Bruker (x) har omsorgen for barnet (z) → ja/nei	{ ?x rentefritak:harOmsorgenFor ?y . } => { rentefritak:M2 sms:resultat "true" } .
M3: Inntil 12 måneder etter fødselen	Barnets fødselsdato (dato) større enn [dagens dato minus ett år] → ja/nei	{ ?x skdf:forelderTil ?z . ?z skdf:fdato ?dato. ?dato math:greaterThan [today – 365] . } => { rentefritak:M3 sms:resultat "true" } .
Felles regler		
PK1-FM1: en brutto månedsinntekt på kr 14.980 eller lavere. Inntektsgrensen blir økt med kr 1.500 for hvert barn under 16 år som låntakeren forsørger.	<p>Liste alle barn, og telle dem, og multiplisere med 1500 og få tillegget til inntektsgrensen. → tillegg</p> <p>Legg tillegget til inntektsgrensen på 14980 → max_inntekt</p> <p>Brukerens brutto årsinntekt dividert på 12 → brutto_månedsinntekt</p> <p>Brutto_månedsinntekt mindre eller lik max_inntekt → ja/nei</p>	<p>[Bruker spørrespråk for å få antall barn]</p> <pre>{ ?x skdf:fnr "03077512345" . ([ant_barn] "1500") math:product ?tillegg . } => { ?x rentefritak:tillegg ?tillegg . } .</pre> <pre>{ ?x skdf:fnr "03077512345" . ?x rentefritak:tillegg ?y . (?y "14980") math:sum ?max_inntekt . } => { sms:peder rentefritak:max_inntekt ?max_inntekt . } .</pre> <pre>{ ?x skdf:fnr "03077512345". ?x skds:almInntekt ?i . (?i "12") math:quotient ?mi . } => { ?x rentefritak:brutto_maanedsinntekt ?mi . } .</pre> <pre>{ ?x skdf:fnr "03077512345". ?x rentefritak:brutto_maanedsinntekt ?bmi . ?x rentefritak:max_inntekt ?max_inntekt . ?bmi math:notGreaterThan ?max_inntekt . } => { rentefritak:PK1-FM1 sms:resultat "true" . } .</pre>

7.4.4 Eksempel på kjøring

Vi har plukket ut tre av reglene i prosessmodellen over, M1, M2 og PK1-FM1, for å demonstrere anvendelse av reglene på personopplysninger i S.M.S. Det første eksempelet forklarer generelt hvordan vi bruker CWM for å simulere S.M.S.' slutningsmekanisme. I det andre eksempelet ser vi på konsekvensen av å ha regler som bruker opplysninger som ikke er tilgjengelige. Det siste eksempelet demonstrerer hvordan S.M.S. slutningsmekanisme er uavhengig av rekkefølgen på regler, slik at opplysninger som blir utledet i forbindelse med en regel, vil komme til nytte i eventuelle andre regler som bruker samme opplysning uavhengig av hvilken rekkefølge reglene presenteres i for slutningsmekanismen.

For å kunne kjøre eksemplene trenger vi eksempeldata som illustrere hvilke opplysninger som er tilgjengelige for S.M.S. om Peder og Marte.²⁰¹ I den forbindelse har vi gjort noen forutsetninger om hvilke personopplysninger som vil være tilgjengelige for S.M.S.

For det første forutsetter vi at inntekts- og skatteopplysninger fra Skattedirektoratet gjøres tilgjengelige. De aktuelle tallene for dette er allerede tilgjengelige for innbyggerene på Internett gjennom blant annet skattekort-tjenesten og selvangivelsen. Videre forutsetter vi at opplysningene om kundeforholdet hos Lånkassen er tilgjengelige som registertjenester. I dagens Minside må man velge transaksjonstjenester for Lånkassen for å få se for eksempel kundenummer. Videre forutsetter vi at Skattedirektoratet øker antallet opplysninger i oppslaget i folkeregisteret ("Min adresse"), slik at også barn inntil en viss alder som har samme adresse som foreldrene, vises, og da med fødselsdato og personnummer.²⁰²

Vi understreker også at vi nå snakker om S.M.S.-versjonen av registertjenester, som leverer opplysningene som RDF-data.

7.4.4.1 Opplysninger om brukeren 'Peder Ås' i S.M.S.²⁰³

Nedenfor har vi gjengitt de personopplysningene S.M.S. har om Peder Ås, hentet fra tre forskjellige registertjenester som RDF-data, som vi for anledningen simulerer gjennom rene tekstfiler i N3-formatet:

- 1) Fra Skatteetaten – folkeregisteret, med navnerommet "skdf:" for <http://sms.skatteetaten.no/freg#> -- filnavn *Registeropplysninger-Peder-skdf.data*
- 2) Fra Skatteetaten – inntektsopplysninger, med navnerommet "skds:" for <http://sms.skatteetaten.no/skatt#> -- filnavn *Registeropplysninger-Peder-skds.data*
- 3) Fra Lånkassen – med navnerommet "lanekassen:" for <http://sms.lanekassen.no/#> -- filnavn *Registeropplysninger-Peder-lanekassen.data*

<p>Fra Skatteetaten – folkeregister</p> <pre> sms:peder a skdf:Person; skdf:fnr "03077512345"; skdf:navn "Peder Ås"; skdf:adresse "Konkretveien 12"; skdf:bostedskommune "Oslo"; skdf:barn [a skdf:Person; skdf:fnr "15010712345"; skdf:fdato "150107"; skdf:navn "Synnøve Kirkerud Ås"; skdf:adresse "Konkretveien 12"; skdf:bostedskommune "Oslo"]; skdf:samboer [a skdf:Person; skdf:fnr "08087512345"; skdf:navn "Marte Kirkerud"; skdf:adresse "Konkretveien 12"; skdf:bostedskommune "Oslo". </pre>	<p>Fra Skatteetaten – inntektsopplysninger</p> <pre> sms:peder a skdf:Person; skdf:fnr "03077512345"; skds:personinntekt "130000"; skds:almInntekt "168000" . </pre> <p>Fra Lånkassen</p> <pre> sms:peder a skdf:Person; skdf:fnr "03077512345"; lanekassen:kundenummer "13579"; lanekassen:totalGjeld "300000" . </pre>
---	--

7.4.4.2 Regel M1: Låntaker som dokumenterer fødsel

Fra prosessmodellen har vi regelen for det materielle inngangskravet M1 formulert i pseudokode og transformert til CWM-format:

M1: Låntaker som dokumenterer fødsel	Bruker (x) har kundenummer (y) i Lånkassen OG Bruker (x) har barn (z) → ja/nei	{?x lanekassen:kundenummer ?y . ?x skdf:barn ?z .} => { rentefritak:M1 sms:resultat "true" } .
--------------------------------------	--	--

For å teste reglene fôrer vi CWM med de tre datafilene med personopplysningene som vi laget over for å representere registertjenestene, samt en regelfil som inneholder regelen slik vi har formulert den i prosessmodellen. Denne filen har vi kalt *Rentefritaksregel-M1.regel*.

Med disse filene som input, gir vi CWM parameteren ”—think”. Dette aktiverer slutningsmekanismen i CWM og fører til at CWM resonnerer på de opplysningene – inkludert reglene – den har fått.

De opplysningene vi har om Peder i de tre datafilene og regelen fra regelfilen går inn i CWMs RDF-lager. Dersom regelen slår til, genereres det en ny opplysning, nemlig at regelen M1 har sms:resultat lik ”true”. Denne opplysningen blir lagt til RDF-lageret. Dersom CWM ikke får andre beskjeder er standard oppførsel at den skriver ut alle opplysningene i RDF-lageret. Dette er uoversiktlig for vår del, vi ønsker kun å få bekreftet eller avkreftet om Peder passerer inngangskravet M1.

Måten å styre hva som kommer ut av CWM er ved hjelp av en filtreringsfunksjon. Filtreringsfunksjonen skrives som en regel. Vi vil bruke følgende filtreringsregel, som vi lagrer i filen *Rapportregel-rentefritak.rapp*:

```
{ ?x sms:resultat "true" . } => { ?x sms:resultat "Regelen er testet med positivt resultat" . } .
```

Denne rapportregelen kan oversettes som følger: For alle elementer som har resultatet ”true”, skriv ut elementet samt teksten ”Regelen er testet med positivt resultat”. Dersom det ikke kommer noe ut, vet vi at testen feilet.

Den komplette kommandoen og resultatet fra CWM blir følgende:

Kommando:

```
C:\CWM\Hovedoppgaven>python %cwm%\cwm.py Registeropplysninger-Peder-skdf.data
Registeropplysninger-Peder-skds.data Registeropplysninger-Peder-lanekassen.data
Rentefritaksregel-M1.regel --think --filter="Rapportregel-rentefritak.rapp"
```

Resultat:

```
#Processed by Id: cwm.py,v 1.185 2006/05/30 21:08:05 syosi Exp
# using base file:/CWM/Hovedoppgaven/
```

```
# Notation3 generation by
# notation3.py,v 1.187 2006/01/13 14:48:54 syosi Exp
```

```
# Base was: file:/CWM/Hovedoppgaven/
@prefix : <http://sms.norge.no/#> .
@prefix rentefritak: <http://www.lovddata.no/for/sf/kd/xd-20061220-1645.html#> .
```

```
rentefritak:M1 :resultat "Regelen er testet med positivt resultat" .
```

```
#ENDS
```

Som vi ser av den uthevede skriften, konkluderer CWM – basert på opplysningene om Peder og de transformerte rentefritaksreglene – med at regelen M1 er oppfylt. I S.M.S. ville denne opplysningen gått inn i arbeidsminnet som en ny opplysning om Peder, sammen med resultatene for kjøringen av de øvrige reglene.

Legg merke til at CWM ikke nødvendigvis gir samme prefiks til de navnerommene som brukes som de vi har brukt i våre data- og regel-filer. I resultatet over ser vi at navnerommet

<http://sms.norge.no/#> er definert som *default* navnerom, det vil si det navnerommet som gjelder dersom man bare angir et tomt prefiks (":"). I våre filer bruker vi prefikset "sms:" for det samme navnerommet. Denne endringen kan være forvirrende for oss som leser resultatet, men det skaper ingen forvirring for datamaskinen, ettersom CWM kun bryr seg om det faktiske navnerommet, og ikke hvilke prefiks vi måtte finne på å tilordne dem.²⁰⁴

7.4.4.3 Regel M2: Den av foreldrene som har omsorgen for barnet

Fra prosessmodellen har vi følgende pseudokode og transformert regel på CWM-format for regel M2:

M2: Den av foreldrene som har omsorgen for barnet	Bruker (x) har omsorgen for barnet (z) → ja/nei	{ ?x rentefritak:harOmsorgenFor ?y . } => { rentefritak:M2 sms:resultat "true" } .
---	---	--

Vi følger samme opplegget som angitt i eksempelet over, men i listen med filer vi fører CWM med, legger vi til regelen vi har laget for regel M2. Dette betyr at vi beholder M1-regelen, slik at vi tester begge reglene samtidig. Rapporten vår skal dermed ikke lenger nøye seg med én oppfylt regel, men liste både regel M1 og M2 i listen over regler som er testet med positivt resultat. Regelen M2 plasserer vi i filen *Rentefritaksregel-M2.regel*.

Denne gangen tar vi med bare selve kommandoen vi har kjørt, og resultatet.

```
C:\CWM\Hovedoppgaven>python %cwm%\cwm.py Registeropplysninger-Peder-skdf.data
Registeropplysninger-Peder-skds.data Registeropplysninger-Peder-lanekassen.data Re
ntefritaksregel-M1.regel rentefritaksregel-m2.regel --think --filter="Rapportregel-rentefritak.rapp"
```

```
rentefritak:M1 :resultat "Regelen er testet med positivt resultat" .
```

Som vi ser er det fortsatt bare M1 som er oppfylt. Årsaken er, som vi var inne på i informasjonsmodellen, at vi mangler opplysninger om hvorvidt Peder "harOmsorgenFor" barnet eller ei. Vi ser ingen umiddelbar mulighet for at det skal finnes en opplysning som kan gi en indikasjon på dette, annet enn eventuelt en opplysning om at Peder mottar foreldrepenger fra NAV. Men det vet vi at han ikke gjør, og det bringer oss ikke nærmere en løsning her. En mulighet kan være å rett og slett å spørre brukeren om å fylle inn manglende opplysninger, og be S.M.S. bruke disse. For S.M.S. vil ikke dette være annerledes enn muligheten brukeren bør ha for å endre opplysninger S.M.S. har fått fra registertjenestene.

For å simulere dette legger vi til en fil – *Registeropplysninger-Peder-egne.data* – som representerer brukerens profil med opplysninger brukeren har gitt og som inneholder opplysningen om at Peder har omsorg for datteren sin, i henhold til betydningen i Lånekassens regelverk for rentefritak:

```
sms:peder
rentefritak:harOmsorgenFor [skdf:fnr "15010712345"] .205
```

Kommandoen og resultatet blir dermed:

Kommando:

```
C:\CWM\Hovedoppgaven>python %cwm%\cwm.py Registeropplysninger-Peder-skdf.data
Registeropplysninger-Peder-skds.data
Registeropplysninger-Peder-lanekassen.data Registeropplysninger-Peder-egne.data Rentefritaksregel-
M1.regel Rentefritaksregel-m2.regel --think --filter="Rapportregel-rentefritak.rapp"
```

Resultat:

rentefritak:M1 :resultat "Regelen er testet med positivt resultat" .
 rentefritak:M2 :resultat "Regelen er testet med positivt resultat" .

Med den ekstra opplysningen om omsorg, ser vi nå at testing av to av inngangskravene, M1 og M2, gir positivt resultat.

7.4.4.4 Felles regel PK1-FM1: Inntektsgrense som økes pr barn

Fra prosessmodellen har vi følgende pseudokode og regel transformert til CWM-format for det første (og eneste) felles materielle inngangskravet for alle underparts-kategorier av partskategori nivå 1.

<p>PK1-FM1: en brutto månedsinntekt på kr 14.980 eller lavere. Inntektsgrensen blir økt med kr 1.500 for hvert barn under 16 år som låntakeren forsørger.</p>	<p>Liste alle barn, og telle dem, og multiplisere med 1500 og få tillegget til inntektsgrensen. → tillegg</p> <p>Legg tillegget til inntektsgrensen på 14980 → max_inntekt</p> <p>Brukerens brutto årsinntekt dividert på 12 → brutto_månedsinntekt</p> <p>Brutto_månedsinntekt mindre eller lik max_inntekt → ja/nei</p>	<p>[Bruker spørrespråk for å få antall barn]</p> <pre>{ ?x skdf:fnr "03077512345" . ([ant_barn] "1500") math:product ?tillegg . } => { ?x rentefritak:tillegg ?tillegg . } . { ?x skdf:fnr "03077512345" . ?x rentefritak:tillegg ?y . (?y "14980") math:sum ?max_inntekt . } => { sms:peder rentefritak:max_inntekt ?max_inntekt. } . { ?x skdf:fnr "03077512345". ?x skds:almInntekt ?i . (?i "12") math:quotient ?mi. } => { ?x rentefritak:brutto_maanedsinntekt ?mi . } . { ?x skdf:fnr "03077512345". ?x rentefritak:brutto_maanedsinntekt ?bmi . ?x rentefritak:max_inntekt ?max_inntekt . ?bmi math:notGreaterThan ?max_inntekt . } => { rentefritak:PK1-FM1 sms:resultat "true" . } .</pre>
---	---	---

Den første delen av regelen PK1-FM1 har vi forenklet. For å finne ut hvor mange barn som er knyttet til brukeren er det enkleste trolig å gjøre en spørring ved hjelp av spørrespråket SPARQL i personopplysningene i S.M.S.' arbeidsminne.²⁰⁶ Vi forutsetter derfor at vi har en verdi for "?antall_barn", og bruker tallet "1" for å representere det i vårt tilfelle.

Vi trenger i tillegg to ekstra opplysninger for å gjennomføre regelen. Den ene er basert på antallet barn, og gir max_inntekt. Denne beregningen består i seg selv av to beregninger, en for å beregne tillegget – 1.500 kr per barn, og en for å legge tillegget til inntektsgrensen på 14.980 kr. Den andre ekstra opplysningen er tallet for månedsinntekt, brutto_maanedsinntekt, som vi beregner ut fra opplysningen om Peders alminnelige inntekt, skdf:almInntekt.

Til sammen gir dette fire nye regler som vi har lagret i filen *Rentefritaksregel-PK1-FM1.regel*. Umiddelbart virker det nødvendig å kontrollere rekkefølgen på hvordan disse reglene blir utført, slik at vi vet at det er beregnet en brutto_maanedsinntekt og en max_inntekt for Peder før disse opplysningene sammenlignes. Isåfall vil det være nødvendig å foreta flere kjøringar av CWM, en for hver regel, og sørge for å legge til de nye verdiene til

datagrunnlaget vårt for hver kjøring – for eksempel i en egen fil med resultatene fra hver kjøring.

Når vi bruker parameteret ”-think” på CWM, derimot, er ikke rekkefølgen på reglene viktig. CWM vil forsøke alle reglene på nytt så lenge minst en av reglene i hver kjøring genererer et resultat, i tilfelle dette resultatet i neste omgang kan være input som gjør at en av de øvrige reglene skal slå til. En konsekvens av dette er at dersom det genereres et resultat i hver omgang reglene anvendes, vil det kjøringen aldri avslutte.²⁰⁷ For å eksemplifisere denne funksjonen har vi laget regelfilen for testing av månedsinntekt mot maksimal inntekt, slik at de tre første reglene, for å beregne maks_inntekt og brutto_månedsinntekt, kommer sist. Den første regelen som CWM møter vil dermed ikke generere noe resultat, fordi de opplysningene som den baserer seg på, ikke finnes. Men når alle reglene har kjørt en gang, er disse beregnet, og testen på om brutto_månedsinntekt er mindre enn maks_inntekt, kan gjennomføres.

Nedenfor er den komplette kommandoen for å teste regelen PK1-FM1 samt reglene M1 og M2, samt resultatet:

```
C:\CWM\Hovedoppgaven>python %CWM%\cwm.py Registeropplysninger-Peder-skdf.data
Registeropplysninger-Peder-skds.data Registeropplysninger-Peder-lanekassen.data
Registeropplysninger-Peder-egne.data Rentefritaksregel-M1.regel Rentefritaksregel-M2.regel
Rentefritaksregel-PK1-FM1.regel --think --filter=Rapportregel-rentefritak.rapp
```

```
rentefritak:M1      :resultat "Regelen er testet med positivt resultat" .
rentefritak:M2      :resultat "Regelen er testet med positivt resultat" .
rentefritak:PK1-FM1 :resultat "Regelen er testet med positivt resultat" .
```

7.4.5 Konklusjon

Som tidligere nevnt vil resultatet av testene legges til arbeidsminnet i S.M.S., og vi kan nå slå fast at arbeidsminnet i S.M.S. ville ha inneholdt informasjon om at Peder passerer de tre inngangskravene M1, M2 og PK1-FM1 (vel og merke hvis han selv legger til opplysningen om at han er hjemme med omsorg for Synnøve). Dersom vi hadde kjørt regelen M3 også, ville den også gitt positivt resultat. Totalt ville dermed fire av de fire reglene som gjelder for partskategorien PK3 ”Fødsel” vært testet positivt. Dette gir et grunnlag for S.M.S. til å sørge for at lenken til Lånekassens søknadsskjema for rentefritak får en prioritert plassering på nettsiden. Dersom Peder ønsker å vite hvorfor dette skjemaet vises så tydelig på S.M.S. kan han velge å få se en begrunnelse, som viser til de fire inngangskravene og at de er testet positivt for ham. Med andre ord kan vi konkludere med at S.M.S. har vært til hjelp for Peder med å finne frem til Lånekassens rentefritaksordning.

7.5 Eksempel – automatisert veiledning basert på Husbankens regler for bostøtte

I forrige kapittel så vi på Lånekassens regler og prototypet deler av funksjonaliteten til S.M.S. ved hjelp av CWM. I dette kapitlet ser vi på regelverket for Husbankens bostøtte. Vi nøyer oss med å vise sorteringen av regelverket, og lager ikke en egen informasjons- eller prosessmodell. Vær oppmerksom på at navnene M1-M3 nå gjelder for Husbankens regler, som angitt i tabellen nedenfor.

Fra scenarioet vet vi at Peder og Marte også har krav på bostøtte fra Husbanken. Nedenfor har vi satt opp en oversikt over sortering av de reglene som er relevante for Peder og Marte med tanke på at de er en husstand med et medlem som er under 18 år. Alle reglene er hentet fra

forskriften om bostøtte²⁰⁸ unntatt definisjonen av hvem som regnes som medlem av husstand, som er hentet fra Husbankens retningslinjer § 3.²⁰⁹

Partskategori	Materielle inngangskrav	Regelreferanse
<i>Felles</i>	FM1: Folkeregistrert og lovlig bosatt i Norge	§ 1, 2. ledd
	FM2: husstandsmedlem [..] alle personer som er registrert bosatt på adressen i folkeregisteret	Retn.l. § 3
PK: Husstandsmedlem under 18 år		
	M1: Husstandsmedlem under 18 år	§ 2, 1. ledd, punkt 1
	M2: Boligen skal ha kjøkken, bad, oppholdsrom, soverom og egen inngang.	§ 4, 1. ledd, 1. setning
	M3: For alle typer boliger gjelder krav om at det skal være helårsbolig.	§ 4, 3. ledd, 1. setning

Merk at tabellen ikke inneholder *unntaksregler*. Blant annet inneholder bostøttereglene krav til hvordan boliger skal være finansiert²¹⁰ for at husstanden skal ha rett til bostøtte. Men i § 3, 3. ledd, punkt 1 gis det unntak for ”Husstander med barn under 18 år”. Dette er en regel som gjelder for vår partskategori. Samtidig er det eneste innholdet i regelen at den sier noe om hvilke regler som *ikke* gjelder for denne partskategorien. Dette håndteres derfor ved at verken unntaksregelen selv, eller reglene den unntar, kommer med blant materielle inngangskrav for vår partskategori.

For M2 og M3 er det tvilsomt om det finnes opplysninger som kan brukes direkte for å teste dette. Riktignok gir tjenesten ”Mine eiendommer” i Minside idag opplysninger om en eiendom for eksempel er en fritidseiendom. Selv om enkelte av verdiene for eiendomstype i ”Mine eiendommer” antagelig kan eliminere muligheten for at en eiendom er en helårsbolig, er det til liten nytte for Peder og Marte, som leier bolig.

For reglene FM1 og FM2 er det interessant at det henvises eksplisitt til *folkeregisteret*. Dette kravet gjelder også implisitt for kravet M1, ettersom husstandsmedlem er definert gjennom FM2. Det gjør at opplysningene fra folkeregisteret i Minside kan brukes direkte for å teste disse. Vi har forutsatt at en S.M.S.-bruker får opplysninger om egne barn for barn under 25 år og som har samme folkeregistrerte adresse som brukeren. Nå utvider vi denne forutsetningen til også å gjelde ektefelle eller samboer man har felles barn med.²¹¹

Siden vi har forutsatt at S.M.S. bare får vite om familiemedlemmer som har samme folkeregistrerte adresse som brukeren selv, kan vi bruke dette til å konkludere med at en husstand består av brukeren samt barn og ektefelle/samboer. Nedenfor viser vi hvordan spørringer ved hjelp av SPARQL kan brukes for å hente ut opplysninger. I det første eksempelet har vi bedt om å få listet opp alle x i arbeidsminnet som har adresse y. Det viser at S.M.S. for brukeren Peder kjenner til tre personer, og alle har samme adresse. I det andre eksempelet endrer vi spørringen til å *kreve* at de som ramses opp *skal ha* samme adresse.²¹²

Tabell 2 -- Eksempel på SPARQL-spørringer for å finne husstandsmedlemmer

Query: PREFIX skdf: <http://sms.skatteetaten.no/freg#> PREFIX sms: <http://sms.norge.no/#> SELECT ?x ?y	Query: PREFIX skdf: <http://sms.skatteetaten.no/freg#> PREFIX sms: <http://sms.norge.no/#> SELECT ?x
--	---

<p>WHERE { ?x skdf:adresse ?y; }</p> <p>-----</p> <p>Query Results (3 answers):</p> <table border="1"> <thead> <tr> <th>x</th> <th>y</th> </tr> </thead> <tbody> <tr> <td>sms:peder</td> <td>"Konkretveien 12"</td> </tr> <tr> <td>e58334d:-7fff</td> <td>"Konkretveien 12"</td> </tr> <tr> <td>e58334d:-8000</td> <td>"Konkretveien 12"</td> </tr> </tbody> </table>	x	y	sms:peder	"Konkretveien 12"	e58334d:-7fff	"Konkretveien 12"	e58334d:-8000	"Konkretveien 12"	<p>WHERE { ?x skdf:adresse "Konkretveien 12" . }</p> <p>-----</p> <p>Query Results (3 answers):</p> <table border="1"> <thead> <tr> <th>x</th> </tr> </thead> <tbody> <tr> <td>sms:peder</td> </tr> <tr> <td>e6159f7:-8000</td> </tr> <tr> <td>e6159f7:-7fff</td> </tr> </tbody> </table>	x	sms:peder	e6159f7:-8000	e6159f7:-7fff
x	y												
sms:peder	"Konkretveien 12"												
e58334d:-7fff	"Konkretveien 12"												
e58334d:-8000	"Konkretveien 12"												
x													
sms:peder													
e6159f7:-8000													
e6159f7:-7fff													

De ubegripelige tegnkombinasjonene i kolonne x er automatisk genererte navn for anonyme subjekter, subjekter vi vet noe om (f.eks. adressen deres), men som vi ikke har gitt en egen URI.

Dataene vi får ut fra spørringene er egnet til å brukes med de andre reglene, for eksempel regel M1 for å teste om et av husstandsmedlemmene er under 18 år.

Med de opplysningene S.M.S. har, kan vi konkludere positivt på FM1 og FM2 (vel og merke uten at vi kontrollerer om husstandsmedlemmene er lovlig bosatt i landet, og heller ikke om noen i husstanden er leietakere hos de andre). Tilsvarende kan vi konkludere positivt for M1. For M2 og M3 kan det til gjengjeld ikke konkluderes positivt, fordi vi ikke har egnede opplysninger. Dette må eventuelt omgås ved å be innbyggeren legge inn disse opplysningene, eller eventuelt bekrefte det vi antar er mest sannsynlig; at husstanden holder hus i en helårsbolig med egen inngang, kjøkken, bad, oppholdsrom og soverom.²¹³ Et siste alternativ er å regne det for så sannsynlig at boligen oppfyller kravene, at regelen som stiller krav til boligen kuttes ut, eller settes til alltid å gi et positivt resultat når transformasjonen av regelverket kun er ment for veiledning.

Behovet for å lage egen transformasjon av regelverket for veiledningsformål, kan vi også begrunne hvis vi ser på bostøttereglenes krav til behovsprøving. Med de reglene vi har sett på til nå – de materielle inngangskriteriene – vil S.M.S. antagelig vurdere "Søknadsskjema for bostøtte" som en relevant tjeneste for en gruppe som i praksis omfatter nesten *alle* familier som har barn. Vi antar at de fleste familiene i Norge er lovlig bosatt her og bor i boliger som tilfredsstillende minstekravene. Dermed er det et spørsmål om den veiledningen S.M.S. kan gi for dette regelverket har noen nytteverdi. Det som mangler er en utsilingsmekanisme for alle de som har for høy *inntekt*.

Problemet er at regelverket er utformet på en måte som gjør det naturlig å kategorisere reglene som inneholder behovsprøving av inntekt, under kategorien for regler om *avgjørelsens nærmere innhold*. Isteden for å fastslå gjennom en inntektsgrense at en person ikke har rett til bostøtte, er reglene lagt opp slik at man må begynne å beregne bostøtten, og først når den er ferdig beregnet, vil det for de fleste komme frem at bostøtte blir 0,-.

Dette skillet er vel et av de vi kan klassifisere som å være av "akademisk interesse". Og ser vi på Husbankens bostøttekalkulator som er tilgjengelig på deres hjemmesider, ser vi at den informerer om at den gir brukeren tilbakemelding om at vedkommende ikke har rett til bostøtte dersom inntekten til husstanden er for høy. Det er med andre ord ikke umulig å omformulere reglene, og "regne bakover", for å kunne operere med en inntektsgrense i praksis. Dette er et ekstra argument for å lage egne transformeringer av regelverket med tanke på *veiledning*, som vi så et mulig behov for under spørsmålet om krav til boligen over.²¹⁴

7.5.1 Konklusjon

På grunn av utformingen av regelverket for Husbankens bostøtte-ordning, og de opplysningene som (ikke) er tilgjengelige for S.M.S., kreves det ekstra tiltak i form av spesial-transformerte regler eller standardsvar som at alle boliger oppfyller minstekravene, for at S.M.S. skal kunne gjøre en relevansvurdering av Husbankens bostøtte-ordning.

7.6 Vurdering – automatisert rettslig veiledning

I de to foregående kapitlene har vi sett hvordan S.M.S. under visse forutsetninger kan brukes til å fastslå om de enkelte inngangskravene til et regelsett er oppfylt eller ei for en S.M.S.-bruker ved hjelp av en slutningsmekanisme tilsvarende den som finnes i CWM.

Opplysningene om hvorvidt inngangskrav for et gitt regelverk er oppfylt eller ei, utgjør nye opplysninger, og disse kan brukes til å vurdere relevansen til den tjenesten som regelverket gjelder for. Denne vurderingen – og opplysningene den bygger på – er grunnlaget for å tilpasse informasjonen på S.M.S. til brukerens konkrete situasjon.

I eksempelet med Lånekassen så vi at det var sannsynlig at S.M.S. ville kunne gi Peder informasjon om at rentefritaksordningen med stor sannsynlighet var relevant for ham, mens vi i eksempelet med Husbanken så at det ville vært behov for tiltak for å sikre at S.M.S. kunne gi meningsfull veiledning for denne ordningen. I den forbindelse så vi at det ikke nødvendigvis er tilstrekkelig med bruk av materielle inngangskrav for å gi veiledning – men at det kan være nødvendig å omformulere reglene om vedtakets nærmere innhold for å få egnete mekanismer for å teste om en tjeneste er relevant. I dette avsnittet ser vi på andre mulige strategier for å gjøre S.M.S. bedre egnet til å foreta automatisert rettslig veiledning.

Vi har sett at det er knyttet utfordringer både til hvilke personopplysninger S.M.S. har tilgang til for en bruker, og utformingen av regelverket. En fare dersom S.M.S. skal realiseres, er at det transformerte regelverket som gjøres tilgjengelig for S.M.S., ikke gjør bruk av de opplysningene S.M.S. har tilgang til. Det vil isåfall gjøre det umulig for S.M.S. å foreta en vurdering av tjenestenes relevans. For å motvirke en sårn situasjon, mener vi en eller flere av følgende strategier kan være aktuelle:

1) Brukeren kan supplere med egne opplysninger

I eksempelet med regel M2 over viste vi hvordan brukeren kan bli forespurt og fylle ut en verdi for en opplysningstype som S.M.S. mangler. Ved gjennomgang og transformering av regelverk med tanke på bruk i S.M.S. er det mulig å se igjennom og vurdere om det er visse opplysningstyper som går igjen i flere regelverk, men som ikke er tilgjengelige i S.M.S., og som det er verdt å be brukeren oppgi selv ved førstegangs innlogging, samt oppdatere med jevne mellomrom.

2) Egne regler for å fastslå at en opplysning eller en kombinasjon av opplysninger tilsvarer en annen opplysning

Dette har vi gitt et eksempel på i forbindelse med regel PK1-FM1. Regelverket for rentefritak opererer med inntektsbegrep per måned, mens S.M.S. kun har tilgang på opplysninger om innbyggerens årsinntekt. Her sa vi det så enkelt som at månedsinntekten var årsinntekt delt på 12. Men virkeligheten er at inntektsbegrepet er et av de mest tydelige eksempler på at forskjellige opplysningstyper har samme betegnelser.

Semantic Web gir gjennom bruken av navnerom en måte å tydeliggjøre hvilket inntektsbegrep det siktes til – for eksempel gir "rentefritak:inntekt" en presisering av at det siktes til inntekt i henhold til reglene for rentefritak, dersom "rentefritak:" er bundet til URIen for regelverket

for rentefritak på Lovdata. Men gjennom bruk av regler og ontologier, er det også mulig å erklære at to inntektsbegrep i en viss sammenheng skal vurderes som like, for eksempel gjennom tripler som ”inntekt1 owl:sameAs inntekt2”. En annen mulighet er å sette sammen flere opplysninger til en, for eksempel ved å gi en regel som sier at inntekt i en sammenheng er summen av fødselspenger og stipend fra lånekassen:

$$\{ (\text{nav:fodselspenger laanekassen:stipend}) \text{ math:sum } ?i \} \Rightarrow \{ \text{sms:peder :inntekt1 } ?i \} .$$

3) Skille mellom transformering av regler for beslutningssystemer og regler for veiledning

En av de største fordelene ved å basere en automatisert veiledningstjeneste som S.M.S. på bruk av Semantic Web-teknologi er at den gjør det mulig for andre å gjenbruke den kunnskapen som for eksempel kommer til uttrykk i et regelverk som er transformert til Semantic Web regelformat.²¹⁵ Derfor ser vi det som formålstjenlig at det transformerte regelverket en tjenesteleverandør gjør tilgjengelig, representerer den transformasjonen av regelverket som brukes av etatens egne beslutningssystemer.

Begrunnelsen for dette er for det første at regelverket isåfall allerede er transformert til ett datamaskinspråk, noe som trolig vil gjøre det relativt lett å transformere det til ett nytt datamaskinspråk, som Semantic Web. For det andre vil det gi en viktig forbedring i den faktiske innsynsmuligheten i beslutningssystemene i forvaltningen.²¹⁶ Etter vår mening kan det siste poenget innebære en viktig kilde til kunnskap om reglene. Vi vil ikke påstå at det er lettere å lese transformert regelverk enn de autentiske tekstene. Men vi vil heller ikke utelukke at systematikken og strukturen som regelverket vil (bør) ha når det er transformert til datamaskinspråk, for noen kan være til hjelp for å forstå den autentiske teksten.

Samtidig som det er fordeler ved at det transformerte regelverket som tjenesteleverandørene gjør tilgjengelig for S.M.S. samsvarer så langt som mulig med det som brukes i tjenesteleverandørenes beslutningssystemer, vil det helt klart være fordeler ved å velge en ”løsere” transformering. Avveiningen kan være at noen få får veldig rettsriktig veiledning, i motsetning til å gi mange flere nokså rettsriktig veiledning.

En løsning kan være å velge en todelt løsning, der utgangspunktet er et nøyaktig og rettsriktig transformert regelverk. Til å supplere dette lages det så egne veiledningsregler som for eksempel kan slå fast at en opplysningstype i det nøyaktige regelverket *for veiledningsformål* kan settes til å være lik en opplysning som S.M.S. har. En forutsetning for en slik delt løsning er at de forskjellige elementene i det transformerte regelverket blir utstyrt med (meta)informasjon som gjør det klart hvilken type regel det er snakk om, for eksempel om regelen er en *tilnærmet riktig* regel, laget for veiledningsformål.

7.6.1 Konklusjon

Vi har sett eksempel på at S.M.S. sannsynligvis ville ha en god mulighet for å vurdere hvor relevant en tjeneste er (rentefritak) og et eksempel på det motsatte (bostøtte). I tillegg har vi gått igjennom mulige strategier for å øke sannsynligheten for at S.M.S. kan gjøre en bedre jobb i flere tilfeller. Dersom det på forhånd sikres at regelverket som transformeres er dekkende for de krav som stilles for de ulike partskategoriene, og det samtidig er sannsynlig at S.M.S. har de opplysningstypene som regelverket gjør bruk av – eller det er mulig å gi supplerende regler som gjør at disse eller tilstrekkelig like opplysninger skaffes til veie – vet vi at S.M.S. kan gi rettslig veiledning til brukerne.

De nærmere avklaringene om *hvordan* det skal formidles til en S.M.S.-bruker at vedkommende for eksempel i henhold til regelverk X som gjelder for tjeneste Y som lenken Z peker til, har oppfylt fire av seks inngangskrav, forutsetter designforslag og brukertesting. Det er også behov for transformering av flere regelverk for å kunne fastslå om det normalt er slik at en høyere prosentvis oppfyllelse av inngangskrav er en god indikasjon på at regelverket er relevant for innbyggeren enn dersom den prosentvise oppfyllelsen av inngangskrav er lavere.

I eksemplene over har vi forholdt oss til ett konkret regelverk, som vi har transformert til CWM-regler basert på strukturen i Schartums transformasjonsmetode, med en liten presisering knyttet til bruken av partskategorier. I det neste kapitlet ser vi nærmere på hvordan vi bruker de opplysningene vi får om regelverket gjennom transformasjonsprosessen, til å utstyre regelverket med tilstrekkelig (meta)informasjon slik at S.M.S. kan styres med en generell oppskrift for hvordan transformert regelverk fra forskjellige kilder om forskjellige tjenester, skal benyttes.

7.7 Nærmere om S.M.S. kontrollstruktur

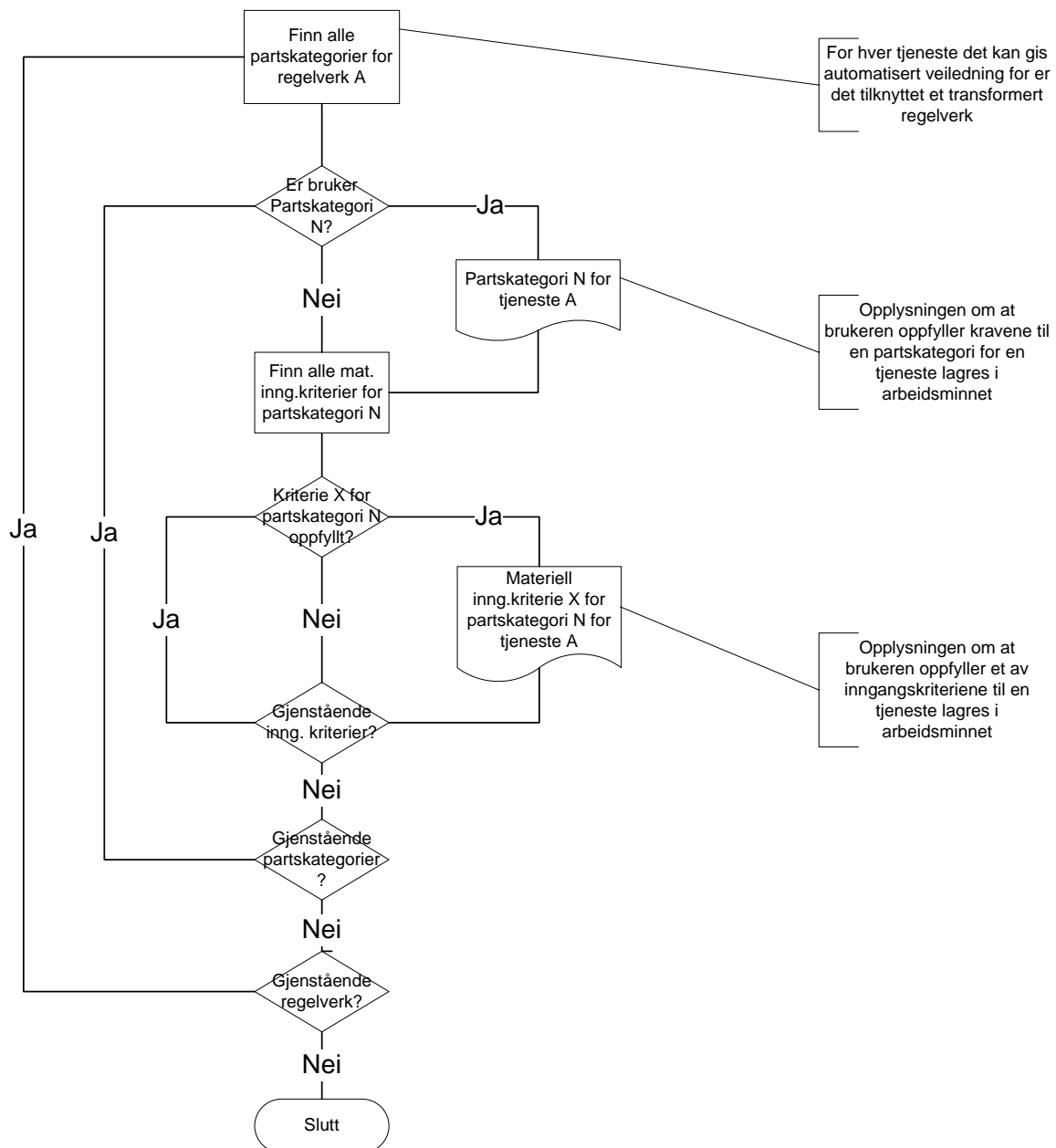
I kapitlene over har vi demonstrert ved hjelp av CWM hvordan S.M.S. kan gjøre bruk av regler som er transformert til CWM-format. Det kan formuleres som at vi har sett hvordan S.M.S. kan ha kunnskap om enkeltregelverk. I dette kapitlet skal vi se på hvordan vi etablerer den kunnskapen som er nødvendig for at S.M.S. skal kunne håndtere regelverk generelt.

I eksemplene over sorterte vi først reglene for rentefritak i partskategorier – på flere nivåer – og i materielle inngangskriterier. Deretter laget vi en rettslig informasjonsmodell og rettslig prosessmodell. For begge disse modellene har vi vist hvordan opplysningstyper og operasjoner kan transformeres til Semantic Web-standardene for data og regler. Som vi har sett gjennom eksempelet på kjøring av reglene på eksempeldata fra scenarioet, har vi langt på vei laget en løsning for veiledning innenfor en del av reglene for rentefritak.

Men S.M.S. er ment å være en løsning for generell veiledning basert på regelverk fra flere aktører, og gjerne flere regelverk per aktør. Nedenfor beskriver vi en generell måte for hvordan transformert regelverk må beskrives, slik at S.M.S. skal kunne håndtere det transformerte regelverket uten at det er behov for spesialtilpasning. Løsningen er basert på å gjøre den sorteringen og inndelingen av regelverket som Schartums transformasjonsmetode anviser, til maskinforståelige opplysninger om regelverket selv, som gjør det mulig å knytte enkeltregler til en generell arbeidsflyt i S.M.S.

I kapittel 7.4.1 valgte vi å ha et hierarki av partskategorier som defineres slik at vi for det laveste partskategorinivået vet at *alle* reglene som ligger under denne partskategorien, samt alle *felles* regler for partskategoriene på høyere nivå, kommer til anvendelse. Dette kan vi bruke til å gi en generell beskrivelse av kontrollstrukturen i S.M.S.

For et gitt regelverk må det testes for alle partskategorier på det øverste nivået. For hver partskategori må det på samme måte testes for hver eventuelle under-partskategori. Slik fortsetter det videre til de materielle inngangskriteriene. Nedenfor har vi laget et flytskjema som illustrerer denne overordnede prosess-strukturen:



Figur 1 -- Flytskjema som illustrerer S.M.S. kontrollstruktur ved håndtering av regelverk

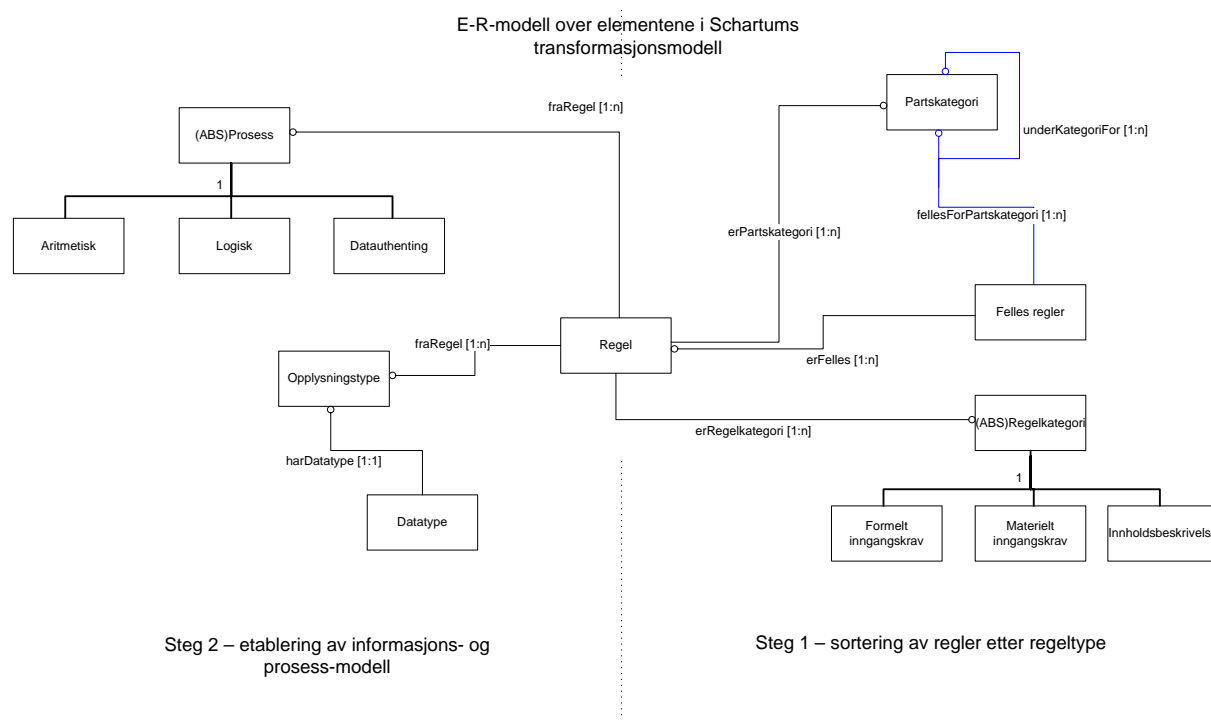
Flytskjemaet over illustrerer den overordnede kontrollstrukturen i S.M.S.' håndtering av regelverk knyttet til tjenestene som vises på S.M.S.

I flytskjemaet har vi ikke tatt med traverseringen gjennom flere nivåer av partskategorier. En annen utvidelse som det er naturlig å tenke seg er å skille mellom inngangskrav som testes og som gir negativt resultat på grunn av opplysningene om brukeren, og negative svar på grunn av at det *mangler* opplysninger. I stedet for kun "ja" eller "nei" på testingen av inngangskrav, åpner en slik nyansering for at resultatet kan bli "kanskje" der hvor S.M.S. ikke har opplysninger til å vurdere kravet.

Den kontrollstrukturen flytskjemaet illustrerer forutsetter at det finnes en standardisert måte – som kan brukes på tvers av tjenesteleverandører og på tvers av regelverk – for å identifisere de forskjellige delene av det transformerte regelverket. Som flytskjemaet over viser er det i første omgang behov for å identifisere regler knyttet til partskategorier og de materielle

inngangskriteriene. En slik identifikasjonsmekanisme vil gi S.M.S. en mulighet for å automatisk gå igjennom reglene slik flytskjemaet viser.

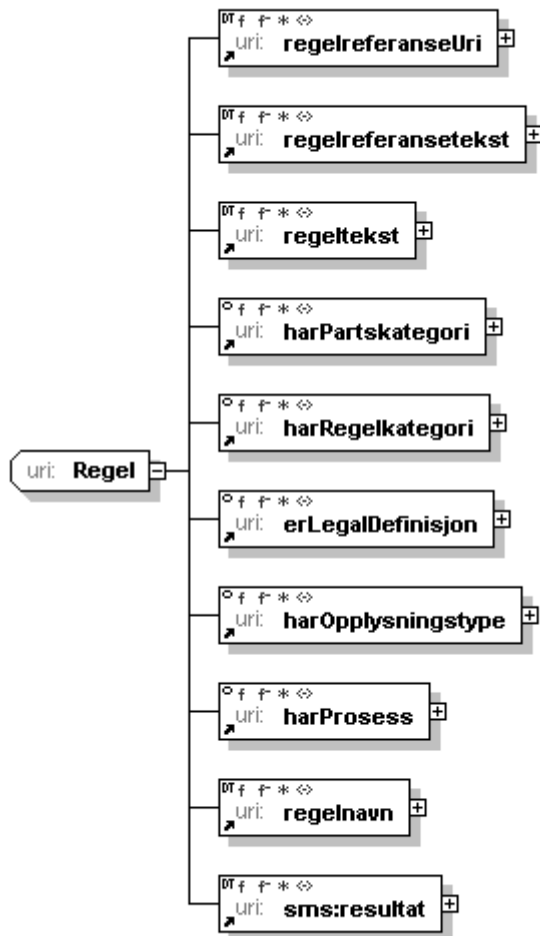
Nedenfor har vi laget et forslag til en datamodell for de opplysningene vi trenger å ta vare på for hver regel. Datamodellen er beskrevet som en ER-modell (Entitet-relasjons-modell) og viser forholdet mellom de sentrale begrepene i Schartums modeller for transformasjonsmetoden. I tillegg har vi lagt inn relasjonen som gjør det mulig å ha flere nivåer av partskategorier, gjennom "underKategoriFor"-relasjonen som forbinder entiteten "Partskategori" med seg selv. Tilsvarende har vi lagt inn muligheten for "Felles regler" å være knyttet til en partskategori, via relasjonen "fellesForPartskategori".



Figur 18 -- Utkast til ER-modell for begrepene i Schartums transformasjonsmetode

Denne modellen må sees på som et utkast som må forfines gjennom bruk. For eksempel har vi ikke tatt med skillet mellom opplysningstyper med faste og variable verdier som brukes for å avgjøre hvorvidt noe trenger å undersøkes, eller om de er faste variabler som kan hentes utenfra, som for eksempel størrelsen på grunnbeløpet (G). (Schartum 2005:64). Et annet aspekt som kan være relevant her er å vurdere behovet for modellering av en opplysningstypes status. For eksempel kan det være et behov for å skille mellom en inntektsopplysning mens den er en påstand og når den er vedtatt i et likningsvedtak.²¹⁷

I tillegg til entitetene og relasjonene som er med i modellen over, må attributter for de forskjellige entitetene og relasjonene detaljeres. Regel-entiteten er spesielt viktig. Vi har brukt et verktøy for arbeid med RDF og OWL datamodeller og ontologier til å skissere en mulig detaljering av regel-entiteten. Nedenfor er en illustrasjon som viser hvilke relasjoner og attributter vi ser for oss.²¹⁸



Generated by SemanticWorks

www.altova.com

Figur 19 -- Utkast til relasjoner og attributter for entiteten "regel"

De attributtene som begynner med "er" eller "har" tilsvarer relasjonene i ER-modellen, bortsett fra "erLegalDefinisjon", som ikke er med i modellen over. Attributtet "sms:resultat" tilsvarer den opplysningstypen vi har brukt for å lagre opplysningen om et krav er testet positivt eller ei. De resterende attributtene er informasjon vi ser for oss bør spesifiseres for hver regel.

Attributtet "Regelnavn" kan være en menneskevennlig måte å identifisere regelen som en del av det transformerte regelverket, for eksempel "PK1-FM1" for å indikere materiell regel 1 for Partskategori 1. Attributtet regelreferansetekst er den menneskevennlige måten å identifisere kilden til regelen i lovverket på, for eksempel med kortnavn på loven, paragrafnummer og punkt. Attributtet regeltekst inneholder den autentiske teksten som regelen representerer. Attributtet regelreferanseUri er en så nøyaktig peker som mulig til regelen i Lovdata.

En bedre løsning enn kombinasjonen av en URL som peker til en hel paragraf, og et attributt som gir den delen av paragrafens tekst vi faktisk snakker om, kunne vært bruk av XPointer.²¹⁹ Dette er en standard for å kunne peke til hvilken som helst del av et XML-dokument, uavhengig av hvor forfatteren har lagt inn ankere eller identiteter som det kan pekes til med vanlige URLer. En XPointer kan også angi både start og slutt, og på den måten trekke ut en del av en setning i en lovtekst. En fordel av mer nøyaktig peking, er at vi vil kunne analysere disse pekerene og finne relasjonene motsatt vei; med utgangspunkt i en regeltekst på Lovdatas

nettsted, vil vi kunne nøste oss tilbake til alle steder denne regelteksten er omtalt. En slik funksjon vil trolig øke muligheten for gjenbruk av allerede gjennomførte transformasjoner av regelverket.²²⁰

Det som ikke er med i figuren er hvordan den ferdige transformerte regelen, i et passende regelformat, skal knyttes til regel-entiteten. Det vil avhenge av hva som blir standarden for regelformat i Semantic Web. Men det er grunn til å anta at regler vil kunne uttrykkes som RDF-data, og da vil det være mulig å koble dette direkte sammen, om nødvendig ved hjelp av lister dersom det er flere enkeltoperasjoner knyttet til en regel. Alternativet er å knytte den transformerte regelen til regel-entiteten som et tekstattributt.

Uavhengig av de detaljerte avklaringene rundt hvordan vi modellerer de opplysningene vi trenger for å kunne referere til transformert regelverk i forhold til kontrollstrukturen i S.M.S., er det viktigste at denne typen nye opplysninger om regelverket – som er grunnlaget for datamaskinenes kunnskap om det transformerte regelverket – gjøres tilgjengelig som RDF-data. Dermed kan vi bruke nøyaktig de samme mekanismene som vi har sett i eksemplene på det konkrete regelverket, til å hente ut det transformerte regelverket på den strukturen S.M.S. er avhengig av å få det.

Nedenfor illustrerer vi det ved hjelp av et eksempel på en regel som vi har representert som RDF-data i henhold til modellen over:

```
<rentefritak:M1>
  a :Regel;
  :harRegelkategori <rentefritak:fødsel>;
  :regelnavn "partskategori-fødsel"^^XML:string;
  :regelreferanseUri ""@http://www.lovdatab.no/for/sf/kd/xd-20061220-1645.html#12-1;
  :regelreferansetekst "§12-1, bokstav c, tittelen";
  :regeltekst ""Låntaker som dokumenterer fødsel [og omsorg etter fødsel.];
  sms:resultat "FALSE";
  rdfs:label "12-1-c-første setning" .
```

Hvis vi hadde hele regelverket i RDF på denne formen, ville vi kunnet bruke spørrings- og slutningsfunksjonen i CWM til å hente enkeltregler avhengig av situasjonen. Nedenfor har vi noen eksempler på forespørsler vi har testet ved hjelp av rapportregler i CWM:

```
#Hente ut alle partskategorier fra et RDF-lager
#{ ?x a trf:Partskategori . } => { ?x a trf:Partskategori . } .

#Hente ut en Partskategori og alle subPartskategorier -- forutsetter regler om motsatt221
{ ?x a trf:Partskategori . ?x trf:harSubPartskategori ?y } =>
{ ?x trf:harSubPartskategori ?y . } .

#Hente ut en regel av en spesiell type og som er testet med negativt resultat
{ ?x trf:harRegelkategori ?y . ?y a trf:Partskategori . ?y trf:erSubPartskategoriAv ?z . ?x sms:resultat
"FALSE" . }
=> { ?x a trf:Regel . ?x trf:harPartskategori ?y . ?y trf:erSubPartskategoriAv ?z . } .
```

7.7.1 Konklusjon

I dette kapitlet har vi sett nærmere på hva som må til for at S.M.S. skal ha tilstrekkelig kunnskap til å kunne behandle et hvilket som helst regelverk. Løsningen vi har skissert ligger i å strukturere det transformerte regelverket på en forhåndsdefinert måte, og sørge for at alle

opplysninger om regelverket er tilgjengelige på samme måte som det transformerte regelverket selv – som RDF-data. En verdifull bi-effekt av dette er at den kunnskapen som gjøres eksplisitt ved for eksempel å angi hvilken partskategori en regel tilhører, ikke bare er tilgjengelig for maskiner, men kan leses og analyseres av mennesker. Dette åpner både for demokratisk kontroll, og spredning av kunnskap om reglene.

Vi har skissert et utkast til en datamodell, og en begynnende ontologi, for bruk med Schartums transformasjonsmetode. Uavhengig av om man ønsker å gå videre med S.M.S. eller ei, er det etter vår mening verdt å videreutvikle en slik ontologi. Dette vil trolig øke sannsynligheten for at Schartums transformasjonsmetode blir tatt i bruk, samtidig som det gjør det trolig at den kunnskapen som i realiteten kommer til uttrykk ved modellering av regelverket i henhold til denne metoden, får et format – gjennom RDF – som er gjenbrukbart og maskinforståelig. Et slikt arbeid kan sees i sammenheng med andre pågående arbeider for å etablere rettslige ontologier i RDF og OWL. Et eksempel på et slikt pågående arbeid er beskrevet i ”An OWL Ontology of Fundamental Legal Concepts”.²²² Disse ontologiene kan utfylle hverandre, ettersom Schartums metode er et utgangspunkt for å modellere *prosessen* i et regelsett – hvordan vise veien fra start til mål i reglene.

7.8 Kravtabell og konklusjon

I tabellen nedenfor går vi igjennom kravene fra den funksjonelle kravspesifikasjonen, og viser hvor i løsningsbeskrivelsen hvert krav er håndtert. Alle krav unntatt to (K-15 og K-22) er beskrevet.

Krav	Kommentar og referanse til løsning
K-23. S.M.S. må i innloggingsperioden kunne hente opplysninger om den innloggede brukeren.	7.2.1.5 – Personopplysninger, 7.2.2 – Utsveksling av personopplysninger [...]
K-24. S.M.S. må kunne bruke de innhentede opplysningene om brukeren til å tilpasse tjenesten basert på disse opplysningene.	7.2.1.15 – Resonnerer, og 7.2.1.16 – S.M.S.’ slutningsmekanisme. Detaljert i prototype-beskrivelse og eksempler (7.3-7.7)
K-25. Det må finnes en måte å identifisere opplysningstyper på tvers av S.M.S. og tjenestene som er tilknyttet S.M.S.	7.2.1.5 – Personopplysninger, forutsetter personopplysninger som RDF-data, som igjen forutsetter bruk av navnerom. Duplisering av opplysningstyper i forskjellige navnerom kan unngås ved harmonisering av datamodeller gjennom SERES.
K-26. Basert på vurderingen av de innhentede opplysningene om brukeren skal S.M.S. tilpasse informasjonen slik at bare tjenester som er relevante for brukeren vises.	7.2.1.17 – Tilpasset presentasjon, og 7.2.1.18 – Tjenester markert som relevante/ikke-relevante, samt videre henvisninger til eksempler fra prototypen.
K-27. Opplysningene i S.M.S. må kunne knyttes til opplysningstyper i informasjonsmodellen i det transformerte regelverket.	7.2.1.9 – Transformert regelverk. Se også svar på krav K3, og K19.
K-28. S.M.S. må ha støtte for de operasjonene som er angitt som del av prosessmodellen.	7.7 – Nærmere om S.M.S.’ kontrollstruktur
K-29. Når brukeren velger en tjeneste skal skjema være forhåndsutfylt med data som allerede er kjent for Minside.	7.2.1.23 – Utføre en tjeneste
K-30. Brukeren må ha mulighet til å overskrive de forhåndsutfylte dataene.	7.2.1.19 – Brukeren endrer/oppdaterer opplysninger, og 7.2.1.23 – Utføre en tjeneste
K-31. Brukeren må få se hva som er kilden til de forhåndsutfylte dataene.	7.2.1.23 – Utføre en tjeneste
K-32. Brukeren skal kunne se hvorfor de tjenestene som er markert som relevante er vurdert som relevante.	7.2.1.19 – Tjenester markert som relevante/ikke-relevante
K-33. Brukeren skal ha kunne velge en visning som viser hvilke tjenester som er skjult, inkludert	7.2.1.19 – Tjenester markert som relevante/ikke-relevante

(lenke til) informasjon om hvorfor.	
K-34. Brukeren skal ha mulighet til å deaktivere denne automatiske tilpasningen av informasjon dersom vedkommende ønsker det.	7.2.1.19 – Tjenester markert som relevante/ikke-relevante
K-35. S.M.S.' henting av opplysninger om brukeren må kun skje når brukeren er innlogget og har gitt sitt samtykke til det.	7.2.1 – Use-case-beskrivelse av S.M.S., punkt 2, samt 7.2.1.4. – Samtykke
K-36. Når brukeren er logget ut, skal ingen opplysninger om brukeren lenger være lagret.	7.2.1.13 – Arbeidsminne, og 7.2.1.25 – Arbeidsminnet tømmes Unntak: Fødselsnummer og eventuelle opplysninger lagret i brukerens profil, se 7.2.1.14
K-37. S.M.S. bør tilby en feedback-mekanisme hvor brukerne kan angi om tilpasningen av informasjon oppleves relevant.	Ikke beskrevet.
K-38. Kildekoden som inneholder transformert regelverk må være åpent tilgjengelig på nettet.	7.2.1.9 – Transformert regelverk, og 7.2.1.12 – Transformert regelverk på nettstedet til tjenesteleverandører [...]
K-39. Hver del av kildekode som inneholder transformert regelverk, må referere til den regelen som den implementerer.	Se forslag til datamodell for opplysninger om regler, 7.7 – Nærmere om S.M.S. kontrollstruktur
K-40. Referansen til regelverket må være så spesifikk som mulig. Som et minimum må den referere med URI til riktig paragraf i Lovdata, i tillegg til eventuelt fritekst for å spesifisere ytterligere.	Se forslag til datamodell for opplysninger om regler, 7.7 – Nærmere om S.M.S. kontrollstruktur
K-41. Kildekoden bør synliggjøre hvilke deler av Schartums transformasjonsmodell den implementerer.	Se forslag til datamodell for opplysninger om regler, 7.7 – Nærmere om S.M.S. kontrollstruktur
K-42. Det må være mulig å anvende regelsett fra private organisasjoner og interessegrupper (tredjepart) på opplysningene i Minside.	7.2.1.11 – Tredjeparter/interesseorganisasjoner, og 7.2.1.12 – Transformert regelverk på nettstedet til tjenesteleverandører [...]
K-43. Anvendelse av andres regelsett må kunne skje uten at personopplysningene i Minside flyter fra Minside til tredjeparten.	Reglene importeres til S.M.S., jfr 7.2.2 – Utsveksling av personopplysninger og regelverk [...]. Se også 7.2.1.22 – Brukeren endrer/utvider regelverksgrunnlaget
K-44. Det må være mulig å relatere tjenester til hverandre.	Kan skje indirekte gjennom at regelverk referer til opplysninger fra andre etater. Forøvrig ikke beskrevet.

Tabellen over viser at løsningsbeskriven for S.M.S. med to unntak dekker de kravene vi utledet basert på behovene vi utledet fra den statlige informasjonspolitikken. I tillegg har vi ved hjelp av eksemplene sett at S.M.S. med stor sannsynlighet ville gjort Peder og Marte oppmerksomme på rentefritaksordningen. Resultatet var ikke like klart for bostøtteordningen, men vi har foreslått strategier som ville øket sannsynligheten for at S.M.S. kunne veiledet Peder og Marte også til den tjenesten. Vi konkluderer derfor med at vi gjennom løsningsforslaget for S.M.S. har vist at Semantic Web-teknologi kan bidra til å gjøre rettighetsinformasjon tilgjengelig på en bedre måte.

8 Oppsummering

I dette kapittelet vil vi først oppsummere de viktigste resultatene av det konkrete arbeidet vi har gjennomført for å se om Semantic Web kan bidra til å gjøre rettighetsinformasjon bedre tilgjengelig. Deretter vil vi oppsummere våre erfaringer med selve arbeidet, inkludert en vurdering av metodevalget.

8.1 Kan Semantic Web gi bedre rettighetsinformasjon?

I vurderingen av løsningsforslaget vårt konkluderte vi positivt på problemstillingen vår; Semantic Web kan bidra til å gjøre rettighetsinformasjon tilgjengelig på en bedre måte. Men det er samtidig noen forutsetninger som må være til stede, og vi anbefalte noen mulige strategier for å sørge for at S.M.S. kan gi best mulige resultater.

Vi mener at det er gode argumenter for å realisere en løsning à la S.M.S. Som vi påpekte i kapittel 3.1 mener vi at det kan vurderes en veiledningsplikt som gjelder for personer som ikke har vært i kontakt med det offentlige, men som det offentlige likevel har nok opplysninger om til å kunne gi målrettet informasjon. Med dagens teknologi forutsetter en slik veiledningsplikt at de ulike forvaltningsorganene deler informasjon om innbyggerne med hverandre. Men kombinasjonen av en Minside-basert løsning, der innbyggeren selv får innsyn i og oversikt over de opplysningene forvaltningen har, og muligheten for å gi automatisert rettslig veiledning basert på disse opplysningene, eliminerer dette problemet. For første gang vil forvaltningen kunne gi konkret og målrettet informasjon til noen de ikke en gang trenger å vite hvem er.

Denne modellen har også den store fordelen at den lar seg kombinere med tredjeparters ønske om og vilje til å veilede. I likhet med dagens web-teknologi, som har gitt enorm tilgang til informasjon, har Semantic Web-teknologi den effekten at ”kunnskapen” som bygges ved hjelp av Semantic Web *akkumuleres*. Det er med andre ord fullt mulig å få nytte av Semantic Web selv om man begynner i det små. En frase som ofte blir brukt for å understreke dette poenget, og som vi slutter oss til, er følgende: ”a little semantics goes a long way”.

Som et siste argument synes vi det er viktig å understreke at S.M.S. ikke nødvendigvis fører til at etater får nye oppgaver i tillegg til de oppgavene de allerede har. De etatene som forvalter velferdstjenester som rentefritak og bostøtte, arbeider uansett med transformasjon av rettsregler fra lovtekst til programkode. Den viktigste endringen for å realisere S.M.S. er at disse etatene må velge å gjøre denne transformasjonen tilgjengelig på et annet *format*. Dersom de gjør det, og bygger videre på våre forslag til modellering, vil det dessuten innebære et forbedret innsyn i hvordan etatene fortolker regelverket når det automatiseres.

Selv om vi har tro på at en løsning basert på vårt løsningsforslag vil være et positivt bidrag, ser vi også at det byr på flere utfordringer.

På det rent praktiske dreier det seg om en delvis umoden teknologi, der viktige deler – som for eksempel regelformatet – ikke ennå er standardisert. Dette gir usikkerhet blant annet om ytelsesmessige forhold og mulige feil. Vi synes samtidig å ha sett en stor utvikling bare iløpet av det siste året. Flere industriaktører ser ut til å omfavne den grunnleggende Semantic Web-standarder RDF, noe som øker sannsynligheten for at eventuelle praktiske hindre blir håndtert i nær fremtid.

Når det gjelder S.M.S. konkret, er utfordringene særlig knyttet til organisatoriske forhold. Hva må til for å sikre at de ulike offentlige instansene harmoniserer sine datamodeller? Semantikkregisteret for elektronisk samhandling (SERES) kan spille en viktig rolle her, men vi har inntrykk av at etatene først og fremst er opptatt av å få orden på sine egne data, før vi får se samhandling på tvers som S.M.S. er avhengig av.

En annen viktig utfordring ved S.M.S. vil være å avklare ansvarsforholdene knyttet til veiledningen – og ikke minst eventuell *villedning* eller mangelfull veiledning. S.M.S. er på mange måter en distribuert løsning og ansvarsforhold kan være vanskelig å identifisere.

Til slutt må vi selvsagt påpeke at spørsmålet om forbedret informasjon om rettigheter kanskje ikke bør knyttes til en diskusjon om hva som er den mest fremtidsrettete teknologien, ettersom de som har de største behovene for rettighetsinformasjon, trolig ikke vil være de første til å bruke teknologien. En eventuell satsing på S.M.S. bør i så fall også ta høyde for kompensierende tiltak for brukergrupper det er lite trolig at vil få nytte av løsningen.

8.1.1 Schartums transformasjonsmetode som grunnlag for veiledning

Schartums transformasjonsmetode kan sees på som et forslag til et sett med rettssikkerhetsgarantier til anvendelse ved utvikling av rettslige beslutningssystemer. Vi har brukt metoden som grunnlag for å analysere eksempelloverket vi har testet S.M.S. med, og i tillegg brukt det som et sentralt element i kontrollstrukturen til S.M.S. I den forbindelsen har vi gjort noen presiseringer. Vi definerer partskategorier strengere, og åpner for et hierarki av partskategorier, slik at det for hver partskategori på laveste nivå sitter igjen med en uttømmende liste over inngangskrav.

Vi har anvendt metoden på *veiledning*, og har sett at den – med de tilpasningen vi har gjort kan fungere godt. Et forbehold gjelder for regelverk som tilsvarende husbankens regler for bostøtte. Her så vi at det først var når vi kom til regeltypen innholdsregler, at vi fant det som i praksis var et viktig inngangskrav, nemlig inntektsgrensen. Uansett vil det etter vår mening være interessant å modellere og teste mer regelverk der Schartums transformasjonsmetode brukes med tanke på å transformere regelverk for veiledningsformål.

8.2 Våre erfaringer med systemutvikling som forskningsmetode

For å kunne gjennomføre denne oppgaven var vi ikke bare avhengige av å forstå hva Semantic Web er, men også hvordan det kan brukes. Det er fordeler og ulemper ved en fremgangsmåte som gjør en avhengig av å sette seg inn i ny teknologi på denne måten.

Som nevnt i kapittel 3 blir systemutvikling som forskningsmetode blant annet sammenlignet med utviklingen innen luftfart. Og på samme måte som at det var mange som fikk en hard landing før brødrene Wright endelig fikk luft under vingene, har vi stadig måtte konstatere at Semantic Web ikke virker. I hvert fall virket det ikke før vi kom på at det manglet et punktum, eller før vi oppdaget vår egen logiske feilslutning. Men for hver feil, lærte vi litt mer, og som belønning for innsatsen kan vi nå konstatere at vi forstår betydelig mer av det som for ikke lenge siden var en endeløs rekke uforståelige tekniske spesifikasjoner.

Vårt viktigste verktøy for å arbeide med Semantic Web har vært CWM. Og vår viktigste kilde til å forstå hva CWM var, hva det kunne brukes til, og hvordan det skulle brukes var *eksempler*. Informasjonen vi fant på CWMs nettsider var til liten nytte før vi gjennom prøving og feiling basert på eksempler i dokumentasjonen, hadde lært de grunnleggende funksjonene. Som en del av dokumentasjonen for denne oppgaven følger det eksempler på datafiler og

regelfiler. Forhåpentligvis kan de komme til nytte for andre som ønsker å bruke CWM for å få en praktisk erfaring med Semantic Web.

Som en avslutning på vår oppsummering av vår erfaring med systemutvikling som forskningsmetode på et fag som forvaltningsinformatikk, har vi sett nærmere på en beskrivelse av hva faget er ment å være.

I sin artikkel om hva forvaltningsinformatikk er og kan bli, skriver Schartum (1994) blant annet om at det bør være et kjennetegn ved faget at det har en ”orientering mot aktuelle og påviste problemer”. En presisering er at faget om nødvendig må la de antatt mest fremtidsrettede løsningene ligge, og isteden konsentrere seg om ”systemløsninger og arbeidsmåter som kan oppfattes som foreldede [...] når disse er praktisk viktige.”

Vår problemstilling – om tilgang til rettighetsinformasjon – har utvilsomt utgangspunkt i et aktuelt og påvist problem. Derimot kan det se ut som at vi har gått utenom den anviste stien og i entusiasme kastet oss over en antatt fremtidsrettet løsning – Semantic Web. Heldigvis finner vi også at forvaltningsinformatikk er tenkt å ha rom for denne typen oppgaver, ved at det ”[l]ikevel [...] selvsagt [må] være rom for forsøk på å komme utviklingen i forkjøpet f.eks ved, på basis av den aktuelle situasjon, å gjøre forutsetninger om samfunnsmessig og teknologisk utvikling.”

Vi har hatt stor glede av å benytte oss av dette rommet.

9 Litteraturliste

Andreessen, M. (14.03.1993), "NCSA Mosaic for X 0.10 released", [online]. Tilgjengelig fra: <http://1997.webhistory.org/www.lists/www-talk.1993q1/0262.html>

Arbeids- og administrasjonsdepartementet 2001, *Informasjonspolitik for statsforvaltningen*, [online]. Tilgjengelig fra: http://www.regjeringen.no/en/dep/fad/Documents/rapporter_planer/Rapporter/2001/Informasjonspolitik-for-statsforvaltningen.html?id=438513

Bench-Capon, T. og M. Sergot 1988, "Towards a Rule-Based Representation of Open-Texture in Law", i Walter, C. (red.), *Computer Power and Legal Language*, Quorum Books, New York.

Berners-Lee, T. (14.10.1998), "Semantic Web Roadmap", *Design Issues*, [online]. Tilgjengelig fra: <http://www.w3.org/DesignIssues/Semantic.html>

-----, (09.03.2006), "Notation3 (N3) A readable RDF syntax", *Design Issues*, [online]. Tilgjengelig fra: <http://www.w3.org/DesignIssues/Notation3>

----- (20.05.2006), "Relational Databases and the Semantic Web", *Design Issues*, [online]. Tilgjengelig fra: <http://www.w3.org/DesignIssues/RDB-RDF.html>

----- (18.07.2006), *AI and Semantic Web*, [online]. Tilgjengelig fra: <http://www.w3.org/2006/Talks/0718-aaai-tbl/Overview.html>

----- (31.10.2006), *CWM*, [online]. Tilgjengelig fra: <http://www.w3.org/2000/10/swap/doc/cwm>

Berners-Lee, T. og M. Fischetti 1999, *Weaving the Web – The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*, Harper, San Francisco.

Berners-Lee, T., J. Hendler og O. Lassila 2001, "Semantic Web", *Scientific American*, [online]. Tilgjengelig fra: <http://www.sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21>

Bing, J. 1982, *Rettslige kommunikasjonsprosesser*, Universitetsforlaget, Oslo. Tilgjengelig fra: <http://www.lovdata.no/litt/rkp/bing-1982-0.html>

----- 2004, "Copymarks: A suggestion for simple management of copyrighted material", *International Review of Law, Computers & Technology*, Volume 18, Issue 3 November 2004, sidene 347 – 374. Tilgjengelig fra: <http://efn.no/copyright/copymarks.html>

Brickley, D. (19.11.2006), *Resource Description Framework (RDF)*, [online], Mozilla Foundation. Tilgjengelig fra: <http://www.mozilla.org/rdf/doc/>

Connolly, D. W. (01.12.1994), e-post tilgjengelig på nettsted med arkiv fra diskusjoner i arbeidsgruppen for HTML (HTML-WG), [online]. Tilgjengelig fra: <https://listserv.heanet.ie/cgi-bin/wa?A2=ind9412&L=html-wg&P=582>

”Description logic”, 2007, *Wikipedia*. [online], Wikipedia. Tilgjengelig fra: http://en.wikipedia.org/wiki/Description_logic

Eckhoff, T og E. Smith 1999, *Forvaltningsrett*, Tano Aschehoug, Oslo.
FOR-2006-12-20-1645 (KD), Forskrift om forrentning og tilbakebetaling av utdanningslån og tap av rettigheter 2007, [online]. Tilgjengelig fra: <http://www.lovdato.no/for/sf/kd/kd-20061220-1645.html>

”Entity Relationship Modell”, 2007, *Wikipedia*, [online], Wikipedia. Tilgjengelig fra: http://en.wikipedia.org/wiki/Entity-relationship_model

Forskrift 2004-12-22 nr 1755 (KD) Forskrift om bostøtte fra Den Norske Stats Husbank, [online]. Tilgjengelig fra: <http://www.lovdato.no/for/sf/kr/xr-20041222-1755.html>

Frihagen, A. 1986, *Forvaltningsloven, kommentarutgave*, bind I, Forlaget A. Frihagen, Oslo

Herman, I. 2006, *State of the Semantic Web*, [online]. Tilgjengelig fra: <http://www.w3.org/2006/Talks/1016-Beijing-IH/>

-----, 2007, *State of the Semantic Web*, [online]. Tilgjengelig fra: <http://www.w3.org/2007/Talks/0223-Bangalore-IH/>

Husbanken 2006, *Håndbok i bostøtte 9b1-1e*

----- 2007a, *Bostøtte 2007 - Håndbok i regelverk*, [online], Husbanken. Tilgjengelig fra: <http://www.husbanken.as/regelvrk/hb-nr.nsf/webRegelverk/14C600F1368D0F0CC1256FB6003E8A1E?OpenDocument>

----- 2007b, *Retningslinjer for bostøtte, HB 9.B.10*, [online], Husbanken. Tilgjengelig fra: <http://www.husbanken.as/regelvrk/hb-nr.nsf/webRegelverk/BD7BF0587C7C9CFCC1257134002BC440?OpenDocument>

IETF, (1998), *A MIME Content-Type for Directory Information*, [online], IETF. Tilgjengelig fra: <http://tools.ietf.org/html/rfc2425>

-----, (01.2005), *Uniform Resource Identifier (URI): Generic Syntax*, RFC 3986, [online], IETF. Tilgjengelig fra: <http://tools.ietf.org/html/rfc3986>

Justis- og politidepartementet 1999-2000, *Stortingsmelding nr. 25: Om fri rettshjelp*, [online], Justis- og politidepartementet. Tilgjengelig fra: <http://www.regjeringen.no/nb/dep/jd/dok/regpubl/stmeld/19992000/Stmeld-nr-25-1999-2000-.html?id=331821>

Kommunikasjonsforeningen (24.01.2005), ”Offentlig informasjonspolitik”, *Syskrinet*, [online], Kommunikasjonsforeningen. Tilgjengelig fra: http://www.syskrinet.no/view_article.asp?id=140

Kuechler, B. og V. Vaishnavi (red.), (18.01.2006), *Design Research in Information Systems*, [online], Association for Information Systems. Tilgjengelig fra: <http://www.isworld.org/Researchdesign/drisISworld.htm>

LOV 1814-05-17 nr 00: *Kongeriget Norges Grundlov, given i Rigsforsamlingen paa Eidsvold den 17de Mai 1814*, [online]. Tilgjengelig fra: <http://www.lovdatab.no/all/hl-18140517-000.html>

LOV 1946-03-01 nr 03 om *Den Norske Stats Husbank*, [online]. Tilgjengelig fra: <http://www.lovdatab.no/all/hl-19460301-003.html>

LOV 1967-02-10 nr 00: *Lov om behandlingsmåten i forvaltningssaker (forvaltningsloven)*, [online]. Tilgjengelig fra: <http://www.lovdatab.no/all/hl-19670210-000.html>

LOV 1969-06-19 nr 53: *Lov om Norsk Lovtidend m.v.*, [online]. Tilgjengelig fra: <http://www.lovdatab.no/all/hl-19690619-053.html>

LOV 2005-06-03 nr 37: *Lov om utdanningsstøtte*, [online]. Tilgjengelig fra: <http://www.lovdatab.no/all/hl-20050603-037.html>

Mahler, T. og T. Olsen 2007, "Identity Management and Privacy Protection Law", Computer Law and Security Report (under publisering)

Miazine, K. 2006, *Rettskilder og hyperstrukturer – om alternative grensesnitt til rettslige informasjonssystemer*, spesialoppgave ved NRCCL, [online], Universitetet i Oslo. Tilgjengelig fra: <http://wo.uio.no/as/WebObjects/theses.woa/wa/these?WORKID=38662>

Norge.no [online].Tilgjengelig fra: www.norge.no

----- 2006, *Implementasjonsguide for registerinformasjon v 0506, teknisk dokumentasjon for tjenesteleverandører til Minside*, [online], Norge.no. Tilgjengelig fra: <http://www.norge.no/minside/teknisk.asp>

Palmer, S. B. *CWM - Closed World Machine*, [online]. Tilgjengelig fra: <http://infomesh.net/2001/cwm/>

Passin, T. B. 2004, *Explorers guide to the Semantic Web*, Manning Publications Co, Greenwich.

Rubino, R., A. Rotolo og G. Sartor 2006, "An OWL Ontology of Fundamental Legal Concepts", i van Engers, T.M. (red.), *Legal Knowledge and Information Systems*, IOS Press, s. 101-110.

Schartum, D. W. 1993, *Rettsikkerhet og systemutvikling i offentlig forvaltning*, Universitetsforlaget, Oslo.

----- 2005a, "Utvikling av beslutningssystemer – fra lovtekst til programkode", *Forvaltningsinformatisk notatserie 1/05*, [online]. Tilgjengelig fra: http://www.afin.uio.no/forskning/notater/utvikling_av_beslutningssystemer.pdf

-----, 2005b. *Juridisk metode innen forvaltningsinformatikken*, [online]. Tilgjengelig fra: <http://www.uio.no/studier/emner/jus/afin/FINF5001/v05/undervisningsmateriale/Juridisk%20metode%20innen%20forvaltningsinformatikken.ppt>

-----, *Offentlig innsyn i forvaltningens saksdokumenter i lys av informasjonsteknologiske muligheter*, [online], Avdeling for forvaltningsinformatikk. Tilgjengelig fra: http://www.afin.uio.no/forskning/andre_publicasjoner/sem260598/dws.html#ref3

Schneider, G. og J. Winters 1998, *Applying use cases*, Addison-Wesley Longman Publishing Co., Inc. Boston

”Semiotic Triangle”, 2007, *Wikipedia*, [online], Wikipedia. Tilgjengelig fra: http://en.wikipedia.org/wiki/Semiotic_triangle

Shadbolt, N., T. Berners-Lee og W. Hall (2006), ”The Semantic Web Revisited”, *IEEE Intelligent Systems* 21(3) sidene. 96-101, [online]. Tilgjengelig fra: http://eprints.ecs.soton.ac.uk/12614/01/Semantic_Web_Revisited.pdf

Skattedirektoratet, (09.2005), *Generelt om folkeregistrering*, [online], Skattedirektoratet. Tilgjengelig fra: <http://www.skatteetaten.no/Templates/Artikkel.aspx?id=6640&epslanguage=EN>

Statens informasjonstjeneste 1998, *LivsIT - elektronisk informasjon i livssituasjoner*

Swick, R. R. og H. S. Thompson (red.) (07.10.1999), ”The Cambridge Communiqué”, W3C. Tilgjengelig fra: <http://www.w3.org/TR/schema-arch>

Teknologirådet 2006, *Offentlige tjenester på Internett*, [online], Teknologirådet. Tilgjengelig fra: http://www.teknologiradet.no/rapport_endelig_isxfU.pdf.file

”vCard”, 2007, *Wikipedia*. [online], Wikipedia. Tilgjengelig fra: <http://en.wikipedia.org/wiki/VCard>

Vedtak om endringer i Stortingsvedtak for Kommunal- og arbeidsdepartementets bostøtteordning og Sosial- og helsedepartementets botilskuddsordning, vedtak av 12. juni 1972 med senere endringer, senest ved vedtak av 23. januar 1997, [online]. Tilgjengelig fra: <http://www.husbanken.as/regelvrk/hb-nr.nsf/webRegelverk/47A783993F270A91C1256488002BA500?OpenDocument>

W3C, [online]. Tilgjengelig fra: www.w3.org

-----, (10.02.2004), *RDF Primer*, [online], W3C. Tilgjengelig fra: <http://www.w3.org/TR/rdf-primer/>

-----, (10.02.2004), *RDF Test Cases*, [online], W3C. Tilgjengelig fra: <http://www.w3.org/TR/rdf-testcases/>

-----, (10.02.2004), *RDF Vocabulary Description Language 1.0: RDF Schema*, [online], W3C. Tilgjengelig fra: <http://www.w3.org/TR/rdf-schema/>

-----, (10.02.2004), *RDF/XML Syntax Specification (Revised)*, [online], W3C. Tilgjengelig fra: <http://www.w3.org/TR/rdf-syntax-grammar/>

-----, (10.07.2006), *RIF Use Cases and Requirements*, [online], W3C. Tilgjengelig fra: <http://www.w3.org/TR/rif-ucr/>

-----, (16.08.2006), *Namespaces in XML 1.0 (Second Edition)*, [online], W3C. Tilgjengelig fra: <http://www.w3.org/TR/REC-xml-names/>

-----, (29.09.2006), *Extensible Markup Language (XML) 1.0 (Fourth Edition)*, [online], W3C. Tilgjengelig fra: <http://www.w3.org/TR/xml/>

-----, (02.03.2007), *Gleaning Resource Descriptions from Dialects of Languages (GRDDL)*, [online], W3C. Tilgjengelig fra: <http://www.w3.org/TR/grddl/>

Ølnes, S. og T. Aaberge 2003, *Enklare tilgang på offentlig informasjon. Evaluering av LivsIT*, VF-rapport nr 7/2003, [online], Vestlandsforskning. Tilgjengelig fra: <http://www.vestforsk.no/www/show.do?page=12&articleid=1231>

----- 2004, *Håndbok for innføring av LivsIT*, VF-rapport nr. 4/2004, [online], Vestlandsforskning. Tilgjengelig fra: <http://www.vestforsk.no/www/show.do?page=12&articleid=1283>

Aasve, Ø. 2007, *Semantiske teknologier – ulike teknologier til ulik bruk*, [online]. Tilgjengelig fra: <http://www.norstella.no/semantiske-teknologier-ulike-teknologier-til-ulik-bruk-presentasjonene-er-lagt-ut.431794-1177.html>

”Ikke helt på nett”, artikkel i Aftenposten 11. april 2007, side 17.

Ikke bare ord... Statlig informasjon mot år 2000, NOU 1992:21 (Viksveen-utvalget)

10 Vedlegg 1: Oppsummering av økonomien til Peder og Marte

Nedenfor er en oppsummering av situasjonen for Peder og Marte fra sommeren 2006 t.o.m. sommeren 2007:

Tid	Hva	Martes inntekt	Peders inntekt	Annet	Samlete midler
Juli	Marte og Peder avslutter studiene	0	0		0
August	1. august: Marte og Peder begynner å jobbe hhv 100 % og 60 %.	23 341	14 005		37 346
Sept.	Marte og Peder flytter inn i leilighet, 7.000,- eks strøm.	23 341	14 005		37 346
Okt.-des.		23 341	14 005		37 346
Januar	Synnøve blir født 15. januar – tre uker før termin	?	?	33 584	33 584
Februar	Marte er hjemme med Synnøve. Peder jobber. Forfall studielån	0	14 005	-12 000	2 005
Mars		0	14 005		14 005
April		0	14 005		14 005
Mai	Forfall studielån	0	14 005	-12 000	2 005
Juni		0	14 005		14 005
Juli		0	14 005		14 005
August	Bytter: Marte jobber, Peder er hjemme	23 341	0		23 341
				Sum	228 993

11 Vedlegg 2: Oppsummering av økonomien dersom Marte og Peder hadde visst

Tid	Hva	Martes inntekt	Peders inntekt	Annet	Bostøtte	Samlete midler
Juli	Marte og Peder avslutter studiene	0	0			0
August	1. august: Marte og Peder begynner å jobbe hhv 100 % og 60 %.	23 341	14 005			37 346
Sept.	Marte og Peder flytter inn i leilighet, 7.000,- eks strøm.	23 341	14 005			37 346
Okt.-des.		23 341	14 005			37 346
Januar	Synnøve blir født 15. januar – tre uker før termin	?	?	33 584		33 584
Februar	Marte er hjemme med Synnøve. Peder jobber. Forfall studielån	0	14 005	-6 000	2 057	10 062
Mars		0	14 005		2 057	16 062
April		0	14 005		2 057	16 062
Mai	Forfall studielån	0	14 005	-6 000	2 057	10 062
Juni		0	14 005		2 057	16 062
Juli		0	14 005		2 057	16 062
August	Bytter: Marte jobber, Peder er hjemme	23 341	0			23 341
					Sum	253 335
					<i>Økning</i>	<i>24 342</i>

12 Vedlegg 3: Beskrivelse av eksempler i kapittel 3

- Originale data skrevet og vedlikeholdt i N3-notasjon. Deretter konvertert til RDF/XML ved hjelp av CWM
- Illustrasjon generert med: <http://www.w3.org/RDF/Validator/direct> som PNG-fil, basert på RDF/XML-data limt inn i skjemaet på nettsiden.
- Standardvalg for generering av figur (farver, font-størrelse, retning)
- Filene med testdata er i vedlegg "Kildekode"

Eksempel på CWM-kommando for å konvertere N3-fil til RDF/XML-fil:

```
C:\CWM>python %cwm%\cwm.py Husbanken\barepeder.n3 --rdf >
Husbanken\RDF\barepeder.rdf
```

12.1 Eksempel – kapittel 3.2.2 – litt om Peder



12.1.1 N3-notasjon

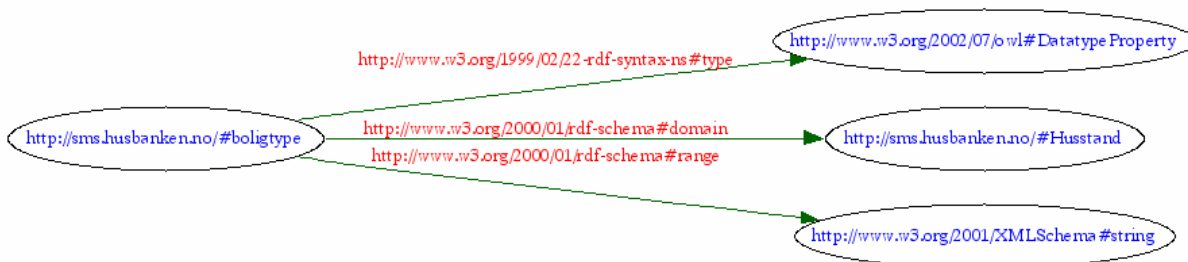
```
sms:peder skdf:navn "Peder Ås".
sms:peder skdf:barn sms:synnove .
```

12.1.2 Filnavn

Original: barepeder.n3

RDF/XML: rdf\barepeder.rdf

12.2 Eksempel – kapittel 3.2.3 – opplysningstyper vs opplysninger



12.2.1 Filnavn

Original: Eksempel2-opplysningstyper-bolig.n3

RDF/XML: rdf\ Eksempel2-opplysningstyper-bolig.rdf

12.3 Eksempel – kapittel 3.3.5 – spørring med SPARQL

12.3.1 Filnavn

Dataene: barepeder.n3

Spørringen: Eksempel4-sparql-query-peder.sparql

12.3.2 Kjøring

Brukte Pellets online demo på <http://www.mindswap.org/2003/pellet/demo.shtml>. La inn dataene fra barepeder.n3 i datafeltet, og spørringen i fila Eksempel4-sparql-query-peder.sparql i spørringsfeltet. Trykket submit.

12.3.3 Resultat

Query Results (1 answers):

Person	Barn
sms:peder	sms:synnove

12.4 Eksempel – kapittel 3.3.6.1 – regler og slutningsmekanismer

Originale data	Resultatdata
sms:peder a skdf:Person; skdf:adresse "Konkretveien 12" .	sms:peder a skdf:Person; skdf:adresse "Konkretveien 12". a husbank:Husstandsmedlem .

12.4.1 Filnavn

Eksempel5-Peder-Husstandsmedlem.n3

12.4.2 CWM-kommando

```
C:\CWM>python %cwm%\cwm.py Husbanken\Eksempel5-Peder-Husstandsmedlem.n3 – think
```

12.4.3 Resultat av kjøring i CWM

```
#Processed by Id: cwm.py,v 1.185 2006/05/30 21:08:05 syosi Exp
# using base file:/CWM/Husbanken\Eksempel5-Peder-
Husstandsmedlem.n3

# Notation3 generation by
# notation3.py,v 1.187 2006/01/13 14:48:54 syosi Exp

# Base was: file:/CWM/Husbanken\Eksempel5-Peder-Husstandsmedlem.n3
@prefix : <http://sms.skatteetaten.no/freg#> .
@prefix Hus: <#> .
@prefix husbank: <http://sms.husbanken.no/#> .
@prefix sms: <http://sms.norge.no/#> .

@forAll Hus:x .

sms:peder a husbank:Husstandsmedlem,
 :Person;
 :adresse "Konkretveien 12" .
{
  Hus:x a :Person;
  :adresse "Konkretveien 12" .
} <http://www.w3.org/2000/10/swap/log#implies> {Hus:x a
husbank
Husstandsmedlem .
} .

#ENDS
```


12.4.3.1 Kommentar

CWM har satt skdf: til ”default” namespace, slik at det ikke fremgår her. Lagt til manuelt i tabellen.

13 Vedlegg 4: Kildekode

Dette vedlegget inneholder innholdet i alle kildekodefilene vi har brukt i oppgaven.

13.1 Kapittel 3 – Eksempeldata

Se Vedlegg "Eksempler kapittel 3" for oversikt over hvilke eksempler disse dataene er brukt i.

13.1.1 barepeder.n3

```
# Instansdefinisjoner for bostøtte-eksempelet i SW-kapitlet,  
# opplysninger hentet fra scenarioet
```

```
# Eksempelet gjelder kun for bostøtte pga barn under 18 år  
# Dersom utvides til f.eks. stønadsmottakere, må det inn et  
# namespace for NAV
```

```
#Forkortelser av "namespace" for våre opplysningstyper:
```

```
@prefix skdf: <http://sms.skatteetaten.no/freg#> .  
@prefix skdl: <http://sms.skatteetaten.no/skatt#> .  
@prefix husbank: <http://sms.husbanken.no/#> .  
@prefix eiendomsinfo: <http://sms.eiendomsinfo.no/#> .  
@prefix sms: <http://sms.norge.no/#> .
```

```
#Forkortelser av namespace for elementer fra W3C
```

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .  
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .  
@prefix owl: <http://www.w3.org/2002/07/owl#> .  
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
```

```
sms:peder
```

```
    skdf:navn "Peder Ås";  
    skdf:barn sms:synnove .
```

13.1.2 Eksempel2-opplysningstyper-bolig.n3

```
# Instansdefinisjoner for bostøtte-eksempelet i SW-kapitlet,  
# opplysninger hentet fra scenarioet
```

```
# Eksempelet gjelder kun for bostøtte pga barn under 18 år  
# Dersom utvides til f.eks. stønadsmottakere, må det inn et  
# namespace for NAV
```

```
#Forkortelser av "namespace" for våre opplysningstyper:
```

```
@prefix skdf: <http://sms.skatteetaten.no/freg#> .  
@prefix skdl: <http://sms.skatteetaten.no/skatt#> .  
@prefix husbank: <http://sms.husbanken.no/#> .  
@prefix eiendomsinfo: <http://sms.eiendomsinfo.no/#> .  
@prefix sms: <http://sms.norge.no/#> .
```

```
#Forkortelser av namespace for elementer fra W3C
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
```

```
husbank:boligtype
  a owl:DatatypeProperty;
  rdfs:domain husbank:Husstand;
  rdfs:range xsd:string .
```

13.1.3 Eksempel4-sparql-query-peder.sparql

```
PREFIX skdf: <http://sms.skatteetaten.no/freg#>
PREFIX sms: <http://sms.norge.no/#>
```

```
SELECT *
WHERE { ?Person skdf:navn "Peder Aas";
        skdf:barn ?Barn }
```

13.1.4 Eksempel5-Peder-Husstandsmedlem.n3

```
# Eksempel til kapitlet "Rules: RIF" i SW-kapitlet
# Viser hvordan regler gir nye data
```

```
#Forkortelser av "namespace" for v re opplysningstyper:
@prefix skdf: <http://sms.skatteetaten.no/freg#> .
@prefix skdl: <http://sms.skatteetaten.no/skatt#> .
@prefix husbank: <http://sms.husbanken.no/#> .
@prefix eiendomsinfo: <http://sms.eiendomsinfo.no/#> .
@prefix sms: <http://sms.norge.no/#> .
```

```
#Forkortelser av namespace for elementer fra W3C
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
```

```
#Originale data
sms:peder
  a skdf:Person;
  skdf:adresse "Konkretveien 12".
```

```
#Regel
{ ?x a skdf:Person. ?x skdf:adresse "Konkretveien 12" } =>
{ ?x a husbank:Husstandsmedlem }.
```

13.2 Kapittel 8 -- Registeropplysninger-filer

13.2.1 Registeropplysninger-Peder-skdf.data

```
# Instansdefinisjoner for Laanekasse-eksempelet i loesningsbeskrivelsen.  
# opplysninger hentet fra scenarioet
```

```
# Eksempelet gjelder kun for bostøtte pga barn under 18 år  
# Dersom utvides til f.eks. stønadsmottakere, må det inn et  
# namespace for NAV
```

```
#Forkortelser av "namespace" for våre opplysningstyper:
```

```
@prefix skdf: <http://sms.skatteetaten.no/freg#> .
```

```
@prefix sms: <http://sms.norge.no/#> .
```

```
#Forkortelser av namespace for elementer fra W3C
```

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
```

```
@prefix owl: <http://www.w3.org/2002/07/owl#> .
```

```
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
```

```
sms:peder
```

```
  a skdf:Person;  
  skdf:fnr "03077512345";  
  skdf:navn "Peder Ås";  
  skdf:adresse "Konkretveien 12";  
  skdf:bostedskommune "Oslo";  
  skdf:barn [  
    a skdf:Person;  
    skdf:fnr "15010712345";  
    skdf:fdato "150107";  
    skdf:navn "Synnøve Kirkerud Ås";  
    skdf:adresse "Konkretveien 12";  
    skdf:bostedskommune "Oslo" ] ;  
  skdf:samboer [  
    a skdf:Person;  
    skdf:fnr "08087512345";  
    skdf:navn "Marte Kirkerud";  
    skdf:adresse "Konkretveien 12";  
    skdf:bostedskommune "Oslo"].
```

13.2.2 Registeropplysninger-Peder-skds.data

```
# Instansdefinisjoner for Laanekasse-eksempelet i loesningsbeskrivelsen.  
# opplysninger hentet fra scenarioet  
# opplysninger om inntekt
```

```
# Eksempelet gjelder kun for bostøtte pga barn under 18 år  
# Dersom utvides til f.eks. stønadsmottakere, må det inn et  
# namespace for NAV
```

#Forkortelser av "namespace" for våre opplysningstyper:

@prefix skdf: <http://sms.skatteetaten.no/freg#> .

@prefix skds: <http://sms.skatteetaten.no/skatt#> .

@prefix sms: <http://sms.norge.no/#> .

#Forkortelser av namespace for elementer fra W3C

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

@prefix owl: <http://www.w3.org/2002/07/owl#> .

@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

sms:peder

a skdf:Person;

skdf:fnr "03077512345";

skds:personinntekt "130000";

skds:almInntekt "168000" .

13.2.3 Registeropplysninger-Peder-lanekassen.data

Instansdefinisjoner for Laanekasse-eksempelet i loesningsbeskrivelsen.

opplysninger hentet fra scenarioet

opplysninger om forholdet til LÅŸnekassen

#Forkortelser av "namespace" for vÅŸre opplysningstyper:

@prefix skdf: <http://sms.skatteetaten.no/freg#> .

@prefix lanekassen: <http://sms.lanekassen.no/#> .

@prefix sms: <http://sms.norge.no/#> .

#Forkortelser av namespace for elementer fra W3C

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

@prefix owl: <http://www.w3.org/2002/07/owl#> .

@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

sms:peder

a skdf:Person;

skdf:fnr "03077512345";

lanekassen:kundenummer "13579";

lanekassen:totalGjeld "300000" .

13.2.4 Registeropplysninger-Peder-egne.data

Instansdefinisjoner for Laanekasse-eksempelet i loesningsbeskrivelsen.

opplysninger hentet fra scenarioet

opplysninger om inntekt

Eksempelet gjelder kun for bostøtte pga barn under 18 år

Dersom utvides til f.eks. stønadsmottakere, må det inn et

namespace for NAV

#Forkortelser av "namespace" for våre opplysningstyper:

@prefix skdf: <http://sms.skatteetaten.no/freg#> .

@prefix skds: <http://sms.skatteetaten.no/skatt#> .

@prefix sms: <http://sms.norge.no/#> .

@prefix rentefritak: <http://www.lovdatab.no/for/sf/kd/xd-20061220-1645.html#> .

#Forkortelser av namespace for elementer fra W3C

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

@prefix owl: <http://www.w3.org/2002/07/owl#> .

@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

sms:peder

rentefritak:harOmsorgenFor [skdf:fnr "15010712345"] .

13.3 Regel-filer

13.3.1 Rentefritaksregel-M1.regel

Regeldefinisjon for Laanekasse-eksempelet i loesningsbeskrivelsen.

#Forkortelser av "namespace" for vÃ¥re opplysningstyper:

@prefix skdf: <http://sms.skatteetaten.no/freg#> .

@prefix sms: <http://sms.norge.no/#> .

@prefix lanekassen: <http://sms.lanekassen.no/#> .

@prefix rentefritak: <http://www.lovdatab.no/for/sf/kd/xd-20061220-1645.html#> .

#Forkortelser av namespace for elementer fra W3C

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

@prefix owl: <http://www.w3.org/2002/07/owl#> .

@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

{

?x lanekassen:kundenummer ?y .

?x skdf:barn ?z .

} =>

{ rentefritak:M1 sms:resultat "true" } .

13.3.2 Rentefritaksregel-M2.regel

Regeldefinisjon for Laanekasse-eksempelet i loesningsbeskrivelsen.

#Forkortelser av "namespace" for vÃ¥re opplysningstyper:

@prefix skdf: <http://sms.skatteetaten.no/freg#> .

```
@prefix sms: <http://sms.norge.no/#> .
@prefix lanekassen: <http://sms.lanekassen.no/#> .
@prefix rentefritak: <http://www.lovdatab.no/for/sf/kd/xd-20061220-1645.html#> .
```

```
#Forkortelser av namespace for elementer fra W3C
```

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
```

```
{ ?x rentefritak:harOmsorgenFor ?y . }
=>
{ rentefritak:M2 sms:resultat "true" } .
```

13.3.3 Rentefritaksregel-PK1-FM1.regel

```
# Regeldefinisjon for Laanekasse-eksempelet i loesningsbeskrivelsen.
```

```
#Forkortelser av "namespace" for vaare opplysningstyper:
```

```
@prefix skds: <http://sms.skatteetaten.no/skatt#> .
@prefix skdf: <http://sms.skatteetaten.no/freg#> .
@prefix sms: <http://sms.norge.no/#> .
@prefix lanekassen: <http://sms.lanekassen.no/#> .
@prefix rentefritak: <http://www.lovdatab.no/for/sf/kd/xd-20061220-1645.html#> .
```

```
#Forkortelser av namespace for elementer fra W3C
```

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
```

```
# Bruken av CWMs innebygde mattefunksjoner forutsetter definisjon av namespace for "math"
```

```
@prefix math: <http://www.w3.org/2000/10/swap/math#> .
```

```
# Variabelen "antall_barn" er hardkodet som "1" i dette tilfellet.
```

```
#Steg 3 -- sjekk Peders brutto maanedsinntekt med maks-inntekt
```

```
{ ?x skdf:fnr "03077512345" .
?x rentefritak:brutto_maanedsinntekt ?bmi .
?x rentefritak:max_inntekt ?max_inntekt .
?bmi math:notGreaterThan ?max_inntekt . } =>
{ rentefritak:PK1-FM1 sms:resultat "true" . }
```

```
#Steg 1a -- beregn tillegget i maks inntekt -- 1500 pr barn --> tillegg
```

```
{ ?x skdf:fnr "03077512345" .
("1" "1500") math:product ?tillegg . } =>
```

```
{ ?x rentefritak:tillegg ?tillegg . } .
```

```
#Steg 1b -- legg tillegget fra regelen over til maks-inntekt
```

```
{ ?x skdf:fnr "03077512345" .
```

```
?x rentefritak:tillegg ?y .
```

```
(?y "14980") math:sum ?max_inntekt . } =>
```

```
{ sms:peder rentefritak:max_inntekt ?max_inntekt. } .
```

```
#Steg 2 -- beregn Peders brutto maanedsinntekt
```

```
{ ?x skdf:fnr "03077512345".
```

```
?x skds:almInntekt ?i .
```

```
(?i "12") math:quotient ?mi. } =>
```

```
{ ?x rentefritak:brutto_maanedsinntekt ?mi . } .
```

13.4 Rapportregler

13.4.1 Rapportregel-rentefritak.rapp

```
# Regeldefinisjon for Laanekasse-eksempelet i loesningsbeskrivelsen.
```

```
#Forkortelser av "namespace" for våre opplysningstyper:
```

```
@prefix skdf: <http://sms.skatteetaten.no/freg#> .
```

```
@prefix sms: <http://sms.norge.no/#> .
```

```
@prefix lanekassen: <http://sms.lanekassen.no/#> .
```

```
@prefix rentefritak: <http://www.lovdata.no/for/sf/kd/xd-20061220-1645.html#> .
```

```
#Forkortelser av namespace for elementer fra W3C
```

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
```

```
@prefix owl: <http://www.w3.org/2002/07/owl#> .
```

```
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
```

```
{ ?x sms:resultat "true" . } => { ?x sms:resultat "Regelen er testet med positivt resultat" . } .
```

13.4.2 Rapportregler-transformasjonsmetoden.rapp

```
# Rapportregler for å hente ut informasjon om _regler_
```

```
# forutsatt info om reglene iht transformasjonsmetoden
```

```
#Forkortelser av "namespace" for våre opplysningstyper:
```

```
@prefix skdf: <http://sms.skatteetaten.no/freg#> .
```

```
@prefix skdl: <http://sms.skatteetaten.no/skatt#> .
```

```
@prefix husbank: <http://sms.husbanken.no/#> .
```

```
@prefix eiendomsinfo: <http://sms.eiendomsinfo.no/#> .
```

```
@prefix sms: <http://sms.norge.no/#> .
```

```
@prefix trf: <http://folk.uio.no/dags#> .
```

```
#Forkortelser av namespace for elementer fra W3C
```

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
```

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
```

```
@prefix owl: <http://www.w3.org/2002/07/owl#> .
```



```

@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

#Forkortelser av namespace for elementer fra W3C for reglene
@prefix math: <http://www.w3.org/2000/10/swap/math#> .

# Fjern kommentarer for å aktivere en regel.
# For rapportering kan kun en regel være aktiv om gangen.

#Eksempelregel:
#{ ?x a sms:Person . } => { ?x a sms:Person . } .
#som kun har til hensikt å skrive ut alle som er sms:Person.

# { ?x a husbank:Husstandsmedlem . ?x skdf:alder ?alder . ?alder math:lessThan "18" } => {
?x a sms:Husstandsmedlem . ?x skdf:alder ?alder . } .
#{ ?x a skdf:Person . } => { ?x a skdf:Person . } .

#Hente ut alle partskategorier fra et RDF-lager
#{ ?x a trf:Partskategori . } => { ?x a trf:Partskategori . ?y a } .

#Hente ut en regel av en spesiell type og som er SANN
{ ?x trf:harRegelkategori ?y . ?y a trf:Partskategori . ?y trf:erSubPartskategoriAv ?z . ?x
sms:resultat "FALSE" . }
=> { ?x a trf:Regel . ?x trf:harPartskategori ?y . ?y trf:erSubPartskategoriAv ?z . } .

#Hente ut en Partskategori og alle subPartskategorier -- forutsetter regler om motsatt
#{ ?x a trf:Partskategori . ?x trf:harSubPartskategori ?y } =>
#{ ?x trf:harSubPartskategori ?y . } .

```

13.5 RDF-data på XML-format

13.5.1 Transformasjonsmetoden

Inneholder OWL-modellering av transformasjonsmetoden, samt et eksempel på en regelinstans. Kan brukes som input til online-tjenesten til Pellet, for å teste rapportreglene for transformasjonsmetoden over (Rapportregler-transformasjonsmetoden.rapp)

```

<?xml version="1.0"?>
<rdf:RDF xmlns="http://folk.uio.no/dags#" xmlns:owl="http://www.w3.org/2002/07/owl#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:sms="http://sms.norge.no/#">
  <rdf:Description rdf:about="http://folk.uio.no/dags#Regel">
    <rdf:type>
      <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
    </rdf:type>
  </rdf:Description>
  <rdf:Description rdf:about="http://folk.uio.no/dags#Partskategori">
    <rdf:type>
      <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
    </rdf:type>

```

```

        <rdfs:subClassOf>
            <rdf:Description
rdf:about="http://folk.uio.no/dags#Transformasjonsmodell"/>
            </rdfs:subClassOf>
        </rdf:Description>
        <rdf:Description rdf:about="http://folk.uio.no/dags#Materiell_inngangskriterie">
            <rdf:type>
                <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
                </rdf:type>
                <rdfs:subClassOf>
                    <rdf:Description rdf:about="http://folk.uio.no/dags#Regeltype"/>
                </rdfs:subClassOf>
                <owl:disjointWith>
                    <rdf:Description
rdf:about="http://folk.uio.no/dags#Om_avgjorelsens_innhold"/>
                </owl:disjointWith>
            </rdf:Description>
            <rdf:Description rdf:about="http://folk.uio.no/dags#Formell_inngangskriterie">
                <rdf:type>
                    <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
                    </rdf:type>
                    <rdfs:subClassOf>
                        <rdf:Description rdf:about="http://folk.uio.no/dags#Regeltype"/>
                    </rdfs:subClassOf>
                    <owl:disjointWith>
                        <rdf:Description
rdf:about="http://folk.uio.no/dags#Materiell_inngangskriterie"/>
                    </owl:disjointWith>
                </rdf:Description>
                <rdf:Description rdf:about="http://folk.uio.no/dags#Legaldefinisjon">
                    <rdf:type>
                        <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
                        </rdf:type>
                    </rdf:Description>
                    <rdf:Description rdf:about="http://folk.uio.no/dags#Opplysningstype">
                        <rdf:type>
                            <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
                            </rdf:type>
                            <rdfs:comment>Opplysningstypene beskriver saksforhold som skal eller
kan legges til grunn for enkeltavgjørelser, og som dessuten beskriver selve avgjørelsen.
(Schartum 2005:57)</rdfs:comment>
                        </rdfs:subClassOf>
                    </rdf:Description>
                <rdf:Description
rdf:about="http://folk.uio.no/dags#Rettslig_informasjonsmodell"/>
                </rdfs:subClassOf>
            </rdf:Description>

```

```

    <rdf:Description rdf:about="http://folk.uio.no/dags#Prosess">
      <rdf:type>
        <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
          </rdf:type>
          <rdfs:comment>"Prosesser" betegner regler for hvorledes
opplysningstyper skal behandles (for eksempel i datamaskinsystemet). Hver prosess leder
fram til et bestemt resultat, for eksempel til et enkeltvedtak av en viss type. (Schartum
2005:57)</rdfs:comment>
          <owl:disjointWith>
            <rdf:Description
rdf:about="http://folk.uio.no/dags#Opplysningstype"/>
              </owl:disjointWith>
              <rdfs:subClassOf>
                <rdf:Description
rdf:about="http://folk.uio.no/dags#Rettslig_prosessmodell"/>
                  </rdfs:subClassOf>
                </rdf:Description>
                <rdf:Description rdf:about="http://folk.uio.no/dags#Transformasjonsmodell">
                  <rdf:type>
                    <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
                      </rdf:type>
                      <rdfs:comment>Toppnoden i modellen</rdfs:comment>
                    </rdf:Description>
                    <rdf:Description rdf:about="http://folk.uio.no/dags#Rettslig_informasjonsmodell">
                      <rdf:type>
                        <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
                          </rdf:type>
                          <rdfs:subClassOf>
                            <rdf:Description
rdf:about="http://folk.uio.no/dags#Transformasjonsmodell"/>
                              </rdfs:subClassOf>
                              <owl:disjointWith>
                                <rdf:Description
rdf:about="http://folk.uio.no/dags#Rettslig_prosessmodell"/>
                                  </owl:disjointWith>
                                </rdf:Description>
                                <rdf:Description rdf:about="http://folk.uio.no/dags#Rettslig_prosessmodell">
                                  <rdf:type>
                                    <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
                                      </rdf:type>
                                      <rdfs:subClassOf>
                                        <rdf:Description
rdf:about="http://folk.uio.no/dags#Transformasjonsmodell"/>
                                          </rdfs:subClassOf>
                                          <owl:disjointWith>

```

```

        <rdf:Description
rdf:about="http://folk.uio.no/dags#Rettslig_informasjonsmodell"/>
        </owl:disjointWith>
    </rdf:Description>
    <rdf:Description rdf:about="http://folk.uio.no/dags#regelreferanseUri">
        <rdf:type>
            <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
            </rdf:type>
            <rdfs:domain>
                <rdf:Description rdf:about="http://folk.uio.no/dags#Regel"/>
            </rdfs:domain>
            <rdfs:range>
                <rdf:Description
rdf:about="http://www.w3.org/2001/XMLSchema#anyURI"/>
            </rdfs:range>
        </rdf:Description>
    <rdf:Description rdf:about="http://folk.uio.no/dags#regelreferansetekst">
        <rdf:type>
            <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
            </rdf:type>
            <rdfs:domain>
                <rdf:Description rdf:about="http://folk.uio.no/dags#Regel"/>
            </rdfs:domain>
            <rdfs:range>
                <rdf:Description
rdf:about="http://www.w3.org/2001/XMLSchema#string"/>
            </rdfs:range>
        </rdf:Description>
    <rdf:Description rdf:about="http://folk.uio.no/dags#regeltekst">
        <rdf:type>
            <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
            </rdf:type>
            <rdfs:domain>
                <rdf:Description rdf:about="http://folk.uio.no/dags#Regel"/>
            </rdfs:domain>
            <rdfs:range>
                <rdf:Description
rdf:about="http://www.w3.org/2001/XMLSchema#string"/>
            </rdfs:range>
        </rdf:Description>
    <rdf:Description rdf:about="http://folk.uio.no/dags#Regeltype">
        <rdf:type>
            <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
            </rdf:type>
            <rdfs:subClassOf>

```

```

        <rdf:Description
rdf:about="http://folk.uio.no/dags#Transformasjonsmodell"/>
        </rdfs:subClassOf>
    </rdf:Description>
    <rdf:Description rdf:about="http://folk.uio.no/dags#harPartskategori">
        <rdf:type>
            <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#ObjectProperty"/>
            </rdf:type>
            <rdfs:domain>
                <rdf:Description rdf:about="http://folk.uio.no/dags#Regel"/>
            </rdfs:domain>
            <rdfs:range>
                <rdf:Description
rdf:about="http://folk.uio.no/dags#Partskategori"/>
            </rdfs:range>
        </rdf:Description>
        <rdf:Description rdf:about="http://folk.uio.no/dags#Om_avgjorelsens_innhold">
            <rdf:type>
                <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
                </rdf:type>
                <rdfs:subClassOf>
                    <rdf:Description rdf:about="http://folk.uio.no/dags#Regeltype"/>
                </rdfs:subClassOf>
                <owl:disjointWith>
                    <rdf:Description
rdf:about="http://folk.uio.no/dags#Formell_inngangskriterie"/>
                </owl:disjointWith>
            </rdf:Description>
            <rdf:Description rdf:about="http://folk.uio.no/dags#harRegelkategori">
                <rdf:type>
                    <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#ObjectProperty"/>
                    </rdf:type>
                    <rdfs:domain>
                        <rdf:Description rdf:about="http://folk.uio.no/dags#Regel"/>
                    </rdfs:domain>
                    <rdfs:range>
                        <rdf:Description rdf:about="http://folk.uio.no/dags#Regeltype"/>
                    </rdfs:range>
                </rdf:Description>
                <rdf:Description rdf:about="http://folk.uio.no/dags#erLegalDefinisjon">
                    <rdf:type>
                        <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#ObjectProperty"/>
                        </rdf:type>
                        <rdfs:domain>
                            <rdf:Description rdf:about="http://folk.uio.no/dags#Regel"/>
                        </rdfs:domain>
                    </rdf:Description>

```

```

        <rdfs:range>
            <rdf:Description
rdf:about="http://folk.uio.no/dags#Legaldefinisjon"/>
            </rdfs:range>
        </rdf:Description>
        <rdf:Description rdf:about="http://folk.uio.no/dags#harOpplysningstype">
            <rdf:type>
                <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#ObjectProperty"/>
                </rdf:type>
            <rdfs:domain>
                <rdf:Description rdf:about="http://folk.uio.no/dags#Regel"/>
            </rdfs:domain>
            <rdfs:range>
                <rdf:Description
rdf:about="http://folk.uio.no/dags#Opplysningstype"/>
                </rdfs:range>
            </rdf:Description>
            <rdf:Description rdf:about="http://folk.uio.no/dags#harProsess">
                <rdf:type>
                    <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#ObjectProperty"/>
                    </rdf:type>
                <rdfs:domain>
                    <rdf:Description rdf:about="http://folk.uio.no/dags#Regel"/>
                </rdfs:domain>
                <rdfs:range>
                    <rdf:Description rdf:about="http://folk.uio.no/dags#Prosess"/>
                </rdfs:range>
            </rdf:Description>
            <rdf:Description rdf:about="http://folk.uio.no/dags#Aritmetisk">
                <rdf:type>
                    <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
                    </rdf:type>
                <rdfs:subClassOf>
                    <rdf:Description rdf:about="http://folk.uio.no/dags#Prosess"/>
                </rdfs:subClassOf>
            </rdf:Description>
            <rdf:Description rdf:about="http://folk.uio.no/dags#Logisk">
                <rdf:type>
                    <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
                    </rdf:type>
                <rdfs:subClassOf>
                    <rdf:Description rdf:about="http://folk.uio.no/dags#Prosess"/>
                </rdfs:subClassOf>
            </rdf:Description>
            <rdf:Description rdf:about="http://folk.uio.no/dags#Dataauthenting">
                <rdf:type>

```

```

        <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
        </rdf:type>
        <rdfs:subClassOf>
            <rdf:Description rdf:about="http://folk.uio.no/dags#Prosess"/>
        </rdfs:subClassOf>
    </rdf:Description>
    <rdf:Description rdf:about="rentefritak:M1">
        <regelreferanseUri xml:lang="http://www.lovdata.no/for/sf/kd/xd-
20061220-1645.html#12-1"></regelreferanseUri>
        <regelreferansetekst>§12-1, bokstav c, tittelen</regelreferansetekst>
        <regeltekst xml:lang="Låntaker som dokumenterer fødsel [og omsorg etter
fødsel.]"></regeltekst>
        <rdf:type>
            <rdf:Description rdf:about="http://folk.uio.no/dags#Regel"/>
        </rdf:type>
        <regelnavn
rdf:datatype="http://www.w3.org/2001/XMLSchema#string">partskategori-
fødsel</regelnavn>
        <harRegelkategori>
            <rdf:Description rdf:about="rentefritak:fodselsel">
                <erSubPartskategoriAv>
                    <rdf:Description
rdf:about="rentefritak:fodselsel_adopsjon">
                        <erSubPartskategoriAv>
                            <rdf:Description
rdf:about="rentefritak:sosiale_økonomiske_aarsaker">
                                <rdf:type>
                                    <rdf:Description rdf:about="http://folk.uio.no/dags#Partskategori"/>
                                </rdf:type>
                            </rdf:Description>
                        </erSubPartskategoriAv>
                    </rdf:type>
                </rdf:Description>
            </erSubPartskategoriAv>
        </rdf:type>
    </rdf:Description>
    </rdf:type>
    </rdf:Description>
    </erSubPartskategoriAv>
    </rdf:type>
    <rdf:Description
rdf:about="http://folk.uio.no/dags#Partskategori"/>
        </rdf:type>
    </rdf:Description>
    </erSubPartskategoriAv>
    </rdf:type>
    <rdf:Description
rdf:about="http://folk.uio.no/dags#Partskategori"/>
        </rdf:type>
    </rdf:Description>
    </erSubPartskategoriAv>
    </rdf:type>
    <rdf:Description
rdf:about="http://folk.uio.no/dags#Partskategori"/>
        </rdf:type>
    </rdf:Description>
    </harRegelkategori>
    <rdfs:label>12-1-c-første setning</rdfs:label>
    <sms:resultat>FALSE</sms:resultat>
    <sms:resultat xml:lang="FALSE"></sms:resultat>
</rdf:Description>
<rdf:Description rdf:about="http://folk.uio.no/dags#Datatype">

```

```

        <rdf:type>
            <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#Class"/>
            </rdf:type>
        </rdf:Description>
        <rdf:Description rdf:about="http://folk.uio.no/dags#regelnavn">
            <rdf:type>
                <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
                </rdf:type>
                <rdfs:domain>
                    <rdf:Description rdf:about="http://folk.uio.no/dags#Regel"/>
                </rdfs:domain>
                <rdfs:range>
                    <rdf:Description
rdf:about="http://www.w3.org/2001/XMLSchema#string"/>
                    </rdfs:range>
                </rdf:Description>
                <rdf:Description rdf:about="http://folk.uio.no/dags#erSubPartskategoriAv">
                    <rdf:type>
                        <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#ObjectProperty"/>
                        </rdf:type>
                        <rdfs:domain>
                            <rdf:Description
rdf:about="http://folk.uio.no/dags#Partskategori"/>
                            </rdfs:domain>
                            <rdfs:range>
                                <rdf:Description
rdf:about="http://folk.uio.no/dags#Partskategori"/>
                                </rdfs:range>
                            </rdf:Description>
                            <rdf:Description rdf:about="http://sms.norge.no/#resultat">
                                <rdf:type>
                                    <rdf:Description
rdf:about="http://www.w3.org/2002/07/owl#DatatypeProperty"/>
                                    </rdf:type>
                                    <rdfs:domain>
                                        <rdf:Description rdf:about="http://folk.uio.no/dags#Regel"/>
                                    </rdfs:domain>
                                    <rdfs:range>
                                        <rdf:Description
rdf:about="http://www.w3.org/2001/XMLSchema#boolean"/>
                                        </rdfs:range>
                                    </rdf:Description>
                                </rdf:RDF>

```


Vedlegg 5: Introduksjon til CWM

Et enkelt verktøy for å eksperimentere med Semantic Web

CWM uttales *coom* og står for ”Closed world machine”. Det er ”hacket” sammen av Sir Tim Berners-Lee etter at han lærte Python, og ønsket et verktøy for å eksperimentere med SW-ideen.

Idag er det flere som bidrar til videreutviklingen av CWM, men såvidt jeg kan vurdere er det ikke planer om å gjøre det til noe annet enn et verktøy for eksperimentering og testing.

CWM er kommandolinjebasert og tilbyr i hovedsak følgende funksjonalitet, fra:

<http://infomesh.net/2001/cwm/>

- Parse and pretty-print the following [RDF](#) formats: [XML RDF](#), [Notation3](#), and [NTriples](#)
- Store triples in a queryable triples database
- Perform inferences as a forward chaining [FOPL](#) inference engine
- Perform builtin functions such as comparing strings, retrieving resources, all using an extensible builtins suite

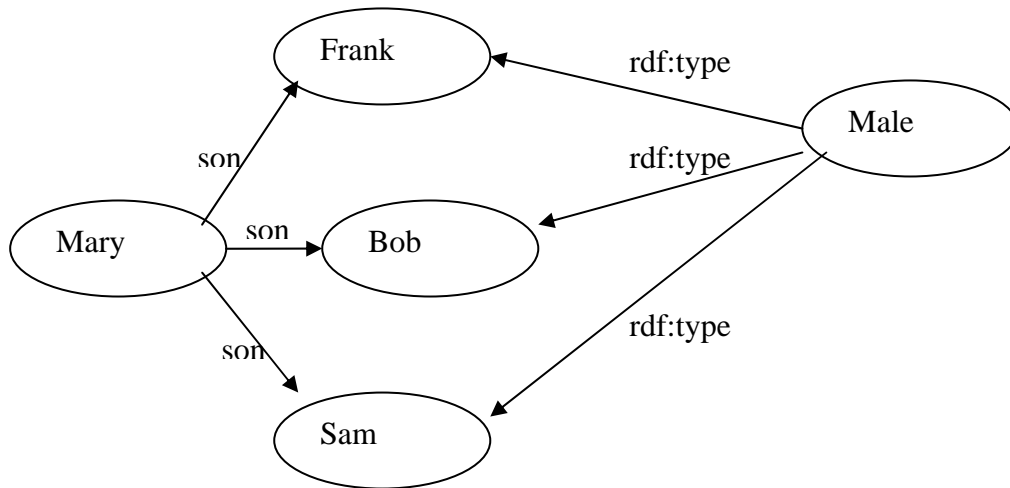
Noe av det nyttige med eksperimentering ved hjelp av CWM er at det tydeliggjør at RDF er en *datamodell* bestående av tripler. Hvilket *format* vi presenterer disse triplene på er valgfritt. CWM takler både XML-representasjonen (som vel ofte er det vi forbinder med RDF, men som er vanskelig å lese) og en forenklet representasjon som er (eller bygger på?) N3 (Notation 3/Turtle?).

Her er en RDF-data uttrykt ved N3:

```
:Mary :son :Frank, a :Male;  
      :son :Bob, a :Male;  
      :son :Sam, a :Male.
```

Hvis man tenker i tripler, er det relativt lett å ”oversette” dette til at *Mary* har tre *sønner*, *Frank*, *Bob* og *Sam*. Hver av disse er av typen *mann*.

Grafisk gjengivelse av dataene, ville se ut omtrent som dette.



(“a” er en “shorthand” for “rdf:type”)

13.6 Egne erfaringer

Det skal være lett å ta i bruk og ”leke” med, men jeg må innrømme at med minimal forhåndskunnskap annet enn på helt overordnet nivå (jeg visste litt om RDF og hadde en drøm om å se ”inference” i praksis ...), var det tungt å trenge igjennom spredt dokumentasjon som ikke presenteres ”for dummies” ... Men iløpet av sommeren føler jeg at jeg kan bruke CWM til å eksperimentere med å

- 1) lage RDF-data
- 2) lage ontologier (OWL)
- 3) skrive enkle regler
- 4) ”smushe” RDF-data fra flere kilder (filer)
- 5) gjøre sparql-spøringer mot RDF-data
- 6) få CWM til å dedusere implisitte data (fra ontologien eller ved hjelp av regler)

Nedenfor er et eksempel på noe av dette hentet fra en tutorial, samt skjermbilder av det samme eksempelet kjørt på min egen maskin ☺

13.7 Eksempler

[Hentet fra: <http://www.w3.org/2003/Talks/0520-www-tf1-b3-rules/all.htm>]

Utgangspunktet er de samme dataene som over:

```
:Mary :son :Frank, a :Male;  
      :son :Bob, a :Male;  
      :son :Sam, a :Male.
```

Men istedenfor å gjenta de samme opplysningene kan vi bruke regelsyntaksen som CWM forstår:

```
{ ?x :son ?y } => { ?y a :Male }.
```

Som leses noe sånn som “for alle x, x er sønn av y, x er en mann”.

Dermed kan vi ha enklere data:

```
# simpler data
```

```
:Mary :son :Frank, :Bob, :Sam.
```

og kombinere det med regelen over.

For å teste det med CWM lagres dataene og reglen i filer, og leses inn. I tillegg aktiveres funksjonen ”think” som forsøker å inferere data:

```
C:\cwm\test> cwm mary-short.n3 son-rules.n3 --think
```

Og ut kommer resultatet:

```
:Mary :son :Frank, a :Male;
```

```
    :son :Bob, a :Male;
```

```
    :son :Sam, a :Male.
```

13.8 Andre eksempler

Over er det rene RDF-data som man kan manipulere med regler. Men CWM kommer også med et sett med regel-filer, inkludert n3-filer (data og regler) som setter CWM istand til å forstå OWL; dermed kan man lage en OWL-ontologi, fore den inn i CWM sammen med regler for OWL-ontologier, og inferere ikke-eksplisitte data fra ontologien. Dette kan for eksempel at en subklasse ikke bare har de attributtene som er definert for selve subklassen, men også de attributtene som er definert for superklassen.

13.9 Skjermbilder fra kjøring av CWM

Merk at her har “son-rules.n3”-fila blitt utvidet til å ha med regelen om at alle sønner er barn av noen.

Merk at CWM på mitt system er satt som følger:

```
C:\CWM>set CWM=C:\Python24\cwm-cvs\2000\10\swap
```

Innholdet i "mary.n3", leses av av CWM, som generer et "triple-store", før innholdet i "triple-store" skrives ut:

```
C:\WINDOWS\system32\cmd.exe
C:\CWM\eksempler>python %CWM%\cwm.py mary.n3
#Processed by Id: cwm.py,v 1.185 2006/05/30 21:08:05 syosi Exp
#   using base file:/CWM/eksempler/mary.n3

# Notation3 generation by
#   notation3.py,v 1.187 2006/01/13 14:48:54 syosi Exp

#   Base was: file:/CWM/eksempler/mary.n3
#   @prefix : <#> .

#   :Mary      :son :Bob,
#             :      :Frank,
#             :      :Sam .

#ENDS
C:\CWM\eksempler>
```

Tilsvarende for innholdet i "mary-short.n3":

```
C:\WINDOWS\system32\cmd.exe
C:\CWM\eksempler>python %CWM%\cwm.py mary-short.n3
#Processed by Id: cwm.py,v 1.185 2006/05/30 21:08:05 syosi Exp
#   using base file:/CWM/eksempler/mary-short.n3

# Notation3 generation by
#   notation3.py,v 1.187 2006/01/13 14:48:54 syosi Exp

#   Base was: file:/CWM/eksempler/mary-short.n3
#   @prefix : <#> .

#   :Mary      :son :Bob,
#             :      :Frank,
#             :      :Sam .

#ENDS
C:\CWM\eksempler>
```

Tilsvarende for innholdet i "son-rules.n3":

```
C:\WINDOWS\system32\cmd.exe
C:\CWM\eksempler>python %CWM%\cwm.py son-rules.n3
#Processed by Id: cwm.py,v 1.185 2006/05/30 21:08:05 syosi Exp
#   using base file:/CWM/eksempler/son-rules.n3

# Notation3 generation by
#   notation3.py,v 1.187 2006/01/13 14:48:54 syosi Exp

#   Base was: file:/CWM/eksempler/son-rules.n3
#   @prefix : <#> .
#   @prefix ma: <mary-short.n3#> .

#   @forall :x,
#     :y .
#   <
#     :x      ma:son :y .
#   >
#   <http://www.w3.org/2000/10/swap/log#implies> <:y      a ma:Child .
#   >,
#   <:y      a ma:Male .
#   > .

#ENDS
C:\CWM\eksempler>
```

Legg merke til at "shorthand"-uttrykket "=>" blir oversatt til full URI for, altså "implies" med namespace <http://www.w3.org/2000/10/swap/log>.

Så kobler vi det sammen, først uten å be CWM tenke:

```

C:\WINDOWS\system32\cmd.exe
C:\CWM\eksempler>python %CWM%\cwm.py mary-short.n3 son-rules.n3
#Processed by Id: cwm.py,v 1.185 2006/05/30 21:08:05 syosi Exp
#   using base file:/CWM/eksempler/

# Notation3 generation by
#   notation3.py,v 1.187 2006/01/13 14:48:54 syosi Exp

#   Base was: file:/CWM/eksempler/
#   @prefix : <mary-short.n3#> .
#   @prefix son: <son-rules.n3#> .

    @forall son:x,
        son:y .

:Mary      :son :Bob,
           :Frank,
           :Sam .

{
  son:x      :son son:y .
}
  <http://www.w3.org/2000/10/swap/log#implies> {son:y      a :Child .
},
  <son:y      a :Male .
} .

#ENDS
C:\CWM\eksempler>_

```

Som vi ser er alt blitt lest inn i samme database, og så skrevet ut, ”urørt”.

Og *endelig*, nå ber vi CWM tenke over innholdet i dataene (samme kommando som over, men med ”think”-parameteret):

```

C:\WINDOWS\system32\cmd.exe
C:\CWM\eksempler>python %CWM%\cwm.py mary-short.n3 son-rules.n3 --think
#Processed by Id: cwm.py,v 1.185 2006/05/30 21:08:05 syosi Exp
#   using base file:/CWM/eksempler/

# Notation3 generation by
#   notation3.py,v 1.187 2006/01/13 14:48:54 syosi Exp

#   Base was: file:/CWM/eksempler/
#   @prefix : <mary-short.n3#> .
#   @prefix son: <son-rules.n3#> .

    @forall son:x,
        son:y .

:Bob      a :Child,
          :Male .

:Frank    a :Child,
          :Male .

:Mary     :son :Bob,
          :Frank,
          :Sam .

:Sam      a :Child,
          :Male .

{
  son:x      :son son:y .
}
  <http://www.w3.org/2000/10/swap/log#implies> {son:y      a :Child .
},
  <son:y      a :Male .
} .

#ENDS
C:\CWM\eksempler>_

```

I tillegg til at Mary har sønnene Bob, Frank og Sam, ser vi i tillegg at de to reglene har gitt CWM mulighet til å dedusere at hver av sønnene er 1) child og 2) male.

13.10 Noen referanser

CWMs hjemmeside: <http://www.w3.org/2000/10/swap/doc/cwm>

A rough guide to N3: <http://infomesh.net/2002/notation3/>

Semantic Web Tutorial using N3: <http://www.w3.org/2000/10/swap/doc/>