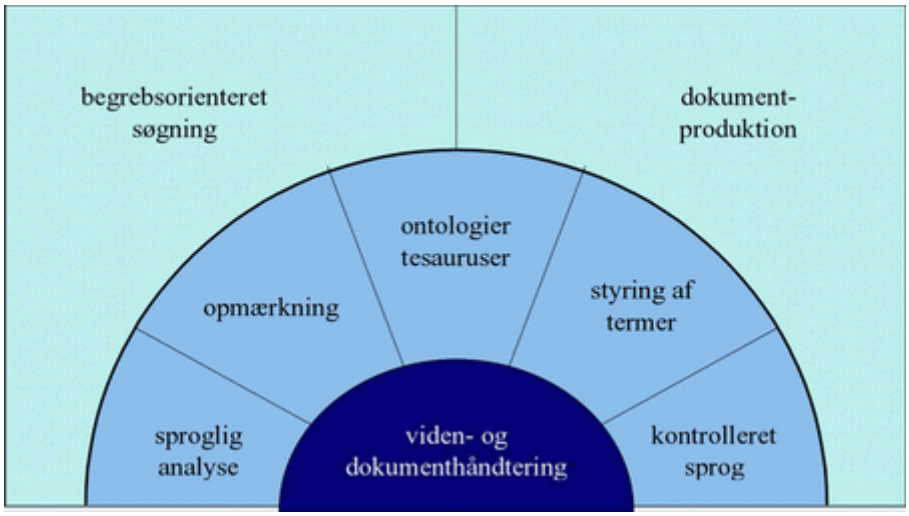


Ontologier og metadata i relation til søgning i tekster

Bolette S. Pedersen, Costanza Navarretta, Dorte Haltrup Hansen

VID-rapport nr. 2

Center for Sprogteknologi
Oktober 2003

Rapporten kan fås ved henvendelse til CST, cst@cst.dk, eller hentes fra CST's hjemmeside www.cst.dk.

VID-projektet er støttet af Center for IT-forskning (nu overgået til Forskningsstyrelsen).

Om VID: Viden- og Dokumenthåndtering med sprogteknologi

Der er et udtalt behov hos danske virksomheder for at kunne supplere deres eksisterende sproglige kompetence og viden med sprogteknologiske IT-værktøjer og metoder som dels kan støtte medarbejderne, dels forankre viden og processer i virksomhedens IT-systemer, dels danne grundlag for den udvikling der kræves hvis virksomhederne skal overleve og vokse i den stadigt mere globaliserede økonomi.

VID-projektet er et forsknings- og udviklingsprojekt der har til formål at udforske de forskellige muligheder som sprogteknologi frembyder inden for informationssøgning og dokumentproduktion, og at understøtte de deltagende virksomheder i at udvikle værktøjer til bedre udnyttelse af egen viden, samt til bedre og mere effektiv produktion af dokumentation, herunder flersproget dokumentation. Foruden CST omfatter projektet på den ene side virksomhederne Bang & Olufsen A/S, Zacco A/S og Nordea A/S, som i dette projekt udgør teknologiens brugere, på den anden Navigo Systems A/S og Ankiro, som er teknologiproducenter. Projektet omfatter følgende forskningsopgaver:

- analyse af de tekstuelle data virksomhederne skal kunne håndtere for at kunne fastlægge tesauruser/ontologier for de relevante semantiske domæner, undersøgelse af den bedst egnede formalisme/teknologi til at udtrykke disse;

- afdækning og videreudvikling af sprogteknologiske komponenter til brug for automatisk tekstklassifikation og begrebsorienteret informationssøgning, indbefattende tilpasning af sprogteknologiske 'basismoduler' til opmærkning af tekst;

- udforskning af flertydighed i tekstuelle data som kan vanskeliggøre informationssøgning; ligeledes den omvendte problematik: at samme indhold kan udformes forskelligt rent sprogligt og derfor kan være svær at fremfinde i store datamængder;

- forskning inden for kontrolleret sprog - også set i et flersproget perspektiv - til brug for dokumentproduktion; herunder analyse af den sprogstil og tone som virksomhederne ønsker at anvende, samt opstilling af modeller for dette sprog;

- undersøgelse af hvilke sprogteknologiske metoder der kan anvendes til denne kvalitetssikring af dokumentproduktionen i form af f.eks. termstyring og grammatikkontrol.

Projektet er støttet af Center for IT-forskning og løber i perioden 2003-2004.

Indhold

1 Indledning

Med den eksplosive udbredelse af elektroniske dokumenter inden for offentlige institutioner, i private virksomheder og på internettet, er der opstået stigende behov for at kunne systematisere tekster og den viden de indeholder, således at søgning i dem kan effektiviseres.

I de senere år har søgning i tekster ændret sig fra at være rent strengbaseret til at anvende mere og mere videnbaserede teknikker. Fælles for disse tilgange er at der anvendes metadata i en eller anden form, således at man ved hjælp af nogle overordnede indholdskategorier kan klassificere og referere til viden. Disse metadata kan bestå af simple, standardiserede kategorier som det ses i fx Dublin Core Metadata, hvor man bl.a. kan definere en teksts *forfatter* og *emne* i dertil indrettede felter til brug for kategorisering og efterfølgende søgning. Eller de kan være strukturerede i form af ontologier bestående af flere hundrede eller flere tusinde begreber med en rig intern struktur i form af relationer på kryds og tværs. Sådanne ontologier kan udnyttes i søgning til at repræsentere elementer af tekstindhold i et formelt sprog. Et helt simpelt eksempel på dette er problemet med forskellige udtryk i tekster der har den samme betydning. Hvis fx *byrådsmedlem* og *medlem af byrådet* refererer til samme ontologiske begreb og teksten (automatisk eller semiautomatisk) er opmærket med denne information, kan dette udnyttes til kategorisering og søgning.

Hensigten med denne rapport er at undersøge state-of-the-art i relation til ontologier og andre former for metadata og at undersøge hvilke formelle sprog og værktøjer der dels er hensigtsmæssige, dels er tilgængelige på nuværende tidspunkt for et forskningsprojekt som VID (VIden og Dokumenthåndtering med sprogteknologi). Specielt interesserer vi os for ontologier, metadata og formelle sprog som har eller er ved at få status som standard inden for området.

Rapporten er inddelt i 3 kapitler. Kapitel 2 beskriver forskellige traditioner inden for ontologiarbejde og vi ser på hvorledes lingvistiske ontologier adskiller sig fra andre formelle ontologier til bl.a. ekspertsystemer. I kapitel 3 ser vi nærmere på standardiserede metadata til beskrivelse af dokumenter sådan som de typisk anvendes inden for biblioteks- og museumsverdenen til kategorisering af tekst- og genstandssamlinger. I kapitel 4 ser vi på formelle sprog og værktøjer til brug for ontologier og metadata. Stort set alle nyere formelle sprog til implementering af ontologier og metadata anvender en XML-baseret teknologi. Generelt gælder det at de mest komplekse og udtryksrige sprog er opbygget som en specialisering af de mere simple, og karakteristisk er det også at sprogene i øjeblikket er i konstant og rivende udvikling.

2 Ontologi – en strukturering af begreber

2.1 Indledning

I dette kapitel ser vi indledningsvis på forskellige tilgange til ontologier inden for filosofi, datalogi og lingvistik (2.2). Dernæst redegør vi i 2.3 for forskellige eksempler på formelle topontologier for derefter i 2.4 at give eksempler på ontologier som er blevet udviklet til mere specifikke informationssystemer. Endelig undersøger vi i 2.5 forskellige lingvistiske ontologier og leksikalske net.

2.2 Forskellige perspektiver på ontologier

Studiet af ontologier er historisk set en filosofisk disciplin stammende fra Aristoteles' undersøgelser af 'tingenes væsen'. Kort sagt har ontologi fra tidernes morgen beskæftiget sig med identifikation af centrale begreber og relationerne imellem disse.

I erkendelse af at dette er en foreteelse som er vanskelig – om ikke umulig - at udføre ud fra rent objektive, formålsuafhængige kriterier, definerer Sowa begrebet ontologi som 'a catalogue of the type of things that are assumed to exist in a domain of interest *D,* from the perspective of a person who uses a language *L* for the purpose of interest *D*' (Sowa 2000: p492).

Et sådant studie er interessant og essentielt inden for en lang række faglige discipliner spændende fra filosofi, logik og datalogi til terminologi og sprogvidenskab. Fælles for disse discipliner er at de har haft behov for at anskue begreber på en struktureret og gerne formelt baseret måde. Af samme grund er ontologier blevet opbygget og anvendt gennem de seneste tiår inden for flere forskellige faglige traditioner.

Inden for datalogi og mere specifikt kunstig intelligens (AI) har man typisk beskæftiget sig med den gren af emnet der omtales *formel ontologi*. Formel ontologi kan defineres som en logisk specifikation af essentielle begreber inden for et domæne struktureret ud fra de relationer der kan identificeres imellem dem (Nilsson 2001:p.11). Som udgangspunkt for en klassifikation af begreber i formel ontologi anvendes som regel inklusion, også benævnt is_a-relationen, fx kan man om forholdet mellem begreberne Hund og Dyr sige at Hund er underbegreb til (is_a) Dyr. Dette betyder normalt at Hund nedarver alle de egenskaber som gælder for Dyr. Inden for formel ontologi interesserer man sig for en yderligere aksiomatisk[1] karakteristik af begreberne. Begreber som er beriget med aksiomer eller definitioner i et formelt sprog gør det muligt at foretage logisk inferens på disse og dette er naturligvis centralt inden for videnrepræsentation. Fx kan man i en given ontologi for begrebet Kæledyr tilskrive følgende aksiom 'der eksisterer Person *x* og Dyr *y* sådan at *x* har et Kæledyr *y*'[2]. Et sådant aksiom gør det i

[1] Et aksiom er et logisk udsagn som hører til grundlaget i et system og som ikke kan bevises inden for dette system.

[2] Taget fra SUMO-ontologien: Sevcenko 2003:p.8: $\exists\, x, y : \mathit{instance}(x, \mathit{Person}) \wedge \mathit{instance}(y, \mathit{Animal}) \wedge \mathit{pet}(x, y)$.

princippet muligt at inferere i form af deduktion i et univers hvori begreber som dyr, kæledyr og mennesker forekommer.

Der findes en lang række formelle sprog til at udtrykke aksiomatiske egenskaber med hvoraf de fleste i en eller anden form bygger på første-ordenslogik[3]. Description Logic er fx et ofte anvendt formelt sprog i forbindelse med ontologiopbygning hvori man kan udtrykke mere komplekse relationer end tilfældet er i førsteordenslogik (se afsnit 4.1. om formelle sprog).

Et alternativ til ontologier med aksiomatiske beskrivelser er en formel ontologi baseret på prototyper, en såkaldt *prototypebaseret ontologi*. I en sådan ontologi defineres kategorier ud fra prototypiske instanser og for hver ny kategori måles den semantiske afstand til den prototypiske kategori. Der findes også hybride ontologier hvor fx metabegreberne beskrives ved hjælp af aksiomer, mens mere specifikke begreber beskrives ved hjælp af prototyper.

En mere anvendelsesorienteret afart af formel ontologi ses i *ontologier til informationssystemer*. Ontologier til informationssystemer er typisk skræddersyet til et ganske bestemt domæne, ofte dog også med reference til mere generelle og almengyldige ontologiske begreber. Sådanne ontologier har et ganske praktisk formål; de kan have til formål at udgøre kernen i et ekspertsystem eller til et system for indholdsbaseret søgning inden for det givne domæne. I og med at disse ontologier som regel er fagspecifikke, er de ofte nært beslægtede med de såkaldte *begrebssystemer* inden for terminologien. Graden af aksiomatisk karakteristik i disse ontologier afhænger af formålet; således indgår deduktion ofte som en central del af ekspertsystemers inferensmaskine hvorfor aksiomer er særligt vigtige i denne applikationstype. Mange systemer til indholdsbaseret søgning udnytter derimod primært inklusionsrelationen i forbindelse med ekspansion af søgestrengen for at forbedre søgemaskinens såkaldte *recall*[4]. Når man søger på *kræftformer* kan søgningen fx udvides til at omfatte begrebet *leukæmi* via inklusionsrelationen mellem de to begreber, og dermed fremfindes flere relevante dokumenter.

Lingvistiske ontologier adskiller sig fra andre ontologier ved at være forankret i det sproglige udtryk; i de leksikalske enheder – altså *ordene* i et givent sprog. En lingvistisk ontologi er med andre ord 'a system of symbols representing the concepts encoded by natural language expressions (lexical units, terms etc.)', Lenci (2003:5). Lingvistiske ontologier benyttes altså først og fremmest til at repræsentere leksikalsk viden på en

[3] Første-ordenslogik er en slags prædikatslogik hvor prædikaterne kun indeholder atomare elementer og hvor kvantorer kun binder atomare variable (Suber http://www.earlham.edu/~peters/courses/logsys/glossary.htm).

[4] Hvis en given database antages at indeholde i alt 50 dokumenter, der kan karakteriseres som værende relevante i forhold til en forespørgsel, og samme forespørgsel fremfinder alle 50 dokumenter, så har den en *recall* på 1. Hvis der kun fremfindes 10 af de 50 relevante dokumenter, så er *recall* på 10/50=0,2, hvilket omvendt betyder 0,8 (80%) af de relevante dokumenter ikke blev fundet, og derfor stadigvæk er ukendte for brugeren (http://www.pce-web.dk/search/size.htm).

struktureret måde og i sagens natur forholder de sig til konkrete sprog som fx *dansk* eller *spansk*.

Inden for sprogvidenskaben og særligt inden for datalingvistikken anvender man som oftest principperne fra formel ontologi. Inklusion er således også en central relation i lingvistiske ontologier; blot omtales relationen typisk inden for sprogvidenskaben som en hyponymi-relation[5]. Ofte vil man dog se at fokus i den lingvistiske ontologi er anderledes. Et af de problematiske aspekter når man behandler naturligt sprog (i modsætning til formelle sprog) er at det viser sig at være meget vanskeligt at lave korrekt inferens; førsteordenslogik er med andre ord ikke nuanceret nok til beskrivelse af især almensproget, bl.a. fordi almensproget ikke altid opfører sig kompositionelt. Frasers betydning kan fx i mange sammenhænge ikke udledes alene ud fra de enkelte ords betydning, men også ud fra deres sammenhæng; at *vaske op* betyder ikke at man vasker i retningen opad, *en grøn student* er ikke en student med farven grøn osv. Man vil derfor typisk se at den lingvistiske ontologi har nogle sondringer som er mere sprogligt/syntaktisk funderede end klassisk formel ontologi; en lingvistisk ontologi skal fx kunne bruges til at skelne imellem forskellige betydninger ved polyseme ord.

Selv om egentlig deduktion som nævnt ofte viser sig at være problematisk inden for lingvistiske ontologier, arbejdes der med inferens*lignende* mekanismer fx i forbindelse med entydiggørelse. Dette ses bl.a. ved Pustejovskys generative mekanismer (Pustejovsky 1995:105-141), så som *selektiv binding*. Selektiv binding foregår fx ved flertydige adjektiver hvor man beregner hvilken dimension i substantivets kernesemantik (*qualia structure*) det flertydige adjektiv modificerer. Ved et eksempel som *et let puslespil* kunne man fx antage at der var tale om et puslespil som ikke vejer ret meget. Den mest umiddelbare fortolkning er imidlertid at der er tale om et puslespil som er at let at lægge, dvs. at *let* her betyder 'som ikke volder nogen større problemer eller anstrengelser' (Nudansk Ordbog) og denne resolution kræver noget beregning. Ved at inferere ud fra puslespils kernesemantik - som bl.a. indeholder den centrale dimension *formål*, nemlig 'at samle brikker så de danner et hele' - kan der foretages en korrekt entydiggørelse af *let* [6]. Også i mange andre entydiggørelsessammenhænge er der en tendens til at formålet med kulturskabte ting spiller en meget vigtig rolle når vi fortolker de ord der omgiver substantivet, og formålet med kulturskabte entiteter må derfor tildeles en særlig vægt i de generative mekanismer[7].

Lingvistiske ontologier er nært beslægtede med såkaldte *leksikalske net* såsom WordNet (se nedenfor) og ofte bruges benævnelserne i flæng. Leksikalske net har som regel

[5] Når et begreb er hyponym til et andet begreb, betyder det at det er underbegreb. Fx er Hund hyponym til Dyr.

[6] I Nudansk ordbog har *let* følgende betydninger: *(i) som ikke vejer el. fylder ret meget* (fx taske: fysisk genstand) *(ii). som har en lav styrke el. grad* (fx trafik: fænomen) *(iii). som ikke volder nogen større problemer el. anstrengelser, el. som sker uden at man gør noget særligt* (fx om tilberedning: handling) *(iv) som er ubekymret el. overfladisk* (fx om person). Definitionen på *puslespil* er i Nudansk som følger: *et spil med træ- el. papbrikker i forskellige faconer som skal lægges sammen så de danner et hele.* Andre dimensioner i kernesemantikken for *puslespil* er således at det har *spil* som overbegreb og består af flere *pap-* eller *træbrikker*.

[7] Pustejovsky bruger selv eksemplerne *a fast car* og *a fast typist* til at illustrere selektiv binding. *Fast* går i begge tilfælde på den teliske rolle, altså hhv. *køre* og *taste*.

ingen aksiomatisk karakteristik, men består udelukkende af leksikalske enheder som er forbundet med hinanden ved hjælp af semantiske relationer, som fx hyponymirelationer, del-helhedsrelationer el. lign. Som regel vil man dog forudsætte at et leksikalsk net i det mindste er forbundet med en sproguafhængig topontologi[8] for at kunne gå under betegnelsen lingvistisk ontologi. I denne sammenhæng bør også nævnes *tesaurusser* som i den klassiske definition også er en ordsamling hvor ordene er ordnet efter betydning med angivelsen af relationerne til over- og underbegreber, samt synonymer, altså meget tæt på definitionen for leksikalske net. I praksis anvendes begrebet tesaurus nu om dage mest om oversigter over søgeord til brug for indeksering, jf. Hermes som bruges på bibliotekerne.

Den leksikalske viden som er udtrykt i lingvistiske ontologier og leksikalske net er svær at komme uden om hvis man ønsker at udvikle systemer der skal kunne håndtere tekst på en nuanceret måde. Også set i et multilingvalt perspektiv er det vigtigt at have lingvistiske ontologier som refererer til specifikke sprog. På et vist specificeringsniveau har forskellige sprog forskellig ontologisk struktur; på spansk har man fx et fælles overbegreb for fingre og tæer: *dedos*; det har vi ikke på dansk; der taler vi om bevægelige kropsdele eller lemmer, men disse begreber er overbegreb for mere end blot fingre og tæer.

I erkendelse af disse problemfelter er der stor interesse for at koble lingvistiske ontologier sammen med formelle ontologier, i det man på den måde kan kombinere den tekstlige forankring fra den lingvistiske tilgang med den større beregningskraft fra den formelle tilgang. Dette gælder bl.a. ved søgning i tekster.

I de senere år har der været en særlig fokus på *ontologistandarder* idet det i forbindelse med især Semantic Web-initiativet[9] er blevet særligt påkrævet at nå til en konsensus om hvordan en ontologi bør opbygges, hvori de mest centrale begreber består og på hvilken måde man kan sammensætte allerede eksisterende ontologier i et netværk af flere mere eller mindre domænespecifikke ontologier der kan interagere i samme system.

2.3 Eksempler på formelle top-ontologier

Se i øvrigt http://ksl-web.stanford.edu/kst/ontology-sources.html for en god oversigt over ontologier og ontologianvendelser.

2.3.1 Upper Cyc Ontology

http://www.cyc.com/cyc-2-1/cover.html.
Upper Cyc Ontology er et ældre og meget ambitiøst ontologiprojekt der er blevet afviklet i løbet af 80'erne og 90'erne i USA. Formålet var at udvikle en almen ontologi

[8] Ved en topontologi forstår man en ontologi der kun inderholder de mest generelle begreber.
[9] Semantic Web betegner en ny form for web-indhold som er forståelig for maskiner og som skal gøre det muligt at søge mere indholdsorienteret på internettet, jf. Berner-Lee et al. 2001.

til anvendelse i informationssystemer. Ontologien (eller dele af den) genanvendes i flere nyere projekter i forskellige versioner. I selve Upper Cyc Ontology er der tale om en formel top-ontologi bestående af ca. 3000 såkaldte konstanter (selve Cyc-ontologien er langt større, men til gengæld ikke offentlig tilgængelig) hvoraf nogle svarer til begreber, mens andre angiver relationer eller logiske operatorer. Konstanter relateres til hinanden ved hjælp af to strukturerende relationer #$isa[10] som forbinder en instans med en klasse, samt #&genls som angiver hvad konstanten er et underbegreb til.

Som det ses i figur 2.1 har hver konstant en definitionsdel i almindeligt sprog samt referencer til relaterede konstanter. *Skin* er altså en instans af *AnimalBodyPartType* og har derudover en række relaterede overbegreber så som *AnimalBodyPart* og *SheetOfSomeStuff*.

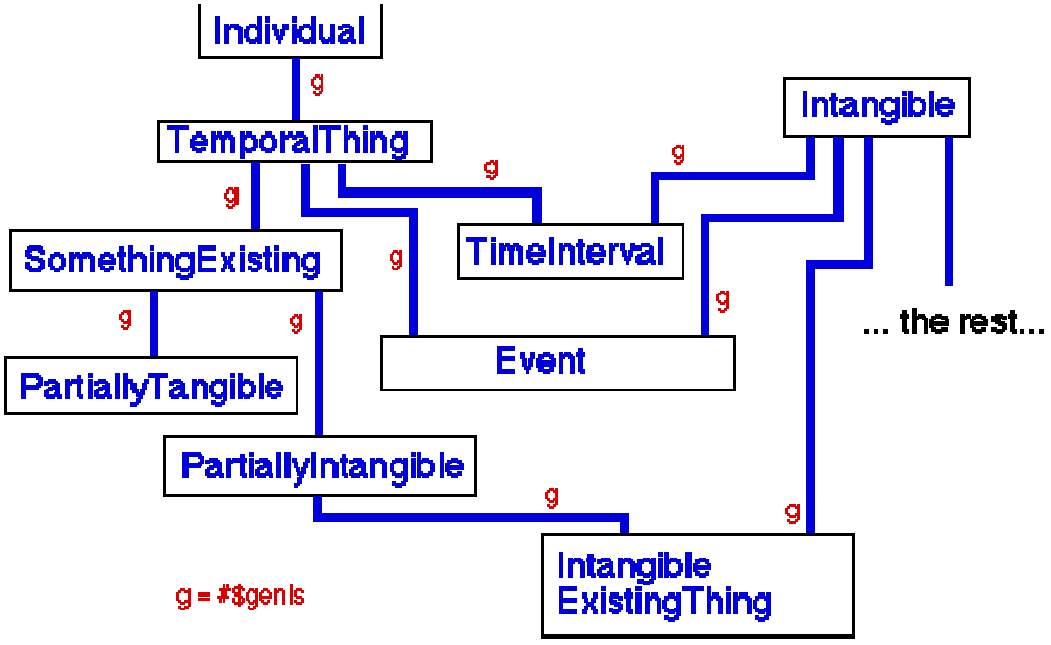
#$Skin

A (piece of) skin serves as outer protective and tactile sensory covering for (part of) an animal's body. This is the collection of all pieces of skin. Some examples include #$TheGoldenFleece (representing an entire skin of an animal) and (#$BodyPartFn #$YulBrynner #$Scalp) (representing a small portion of his skin).
isa: #$AnimalBodyPartType
genls: #$BiologicalLivingObject #$AnimalBodyPart #$SheetOfSomeStuff #$VibrationThroughAMediumSensor #$TactileSensor

Figur 2.1: Konstanten *skin* og dens relaterede konstanter i Upper Cyc Ontology

I Figur 2.2 gengives grafisk konstanten *event* og dets relaterede konstanter.

[10] Bemærk at der altså ikke tale om den almindelige fortolkning af isa, nemlig 'er undertype til' eller 'er underbegreb til', men snarere: 'er en instans af'.



Figur 2.2: Konstanten *event* og dens relaterede konstanter i Upper Cyc Ontology

Upper Cyc Ontology kan downloades i HTML fra hjemmesiden, men der arbejdes også på at konvertere den til RDF og Topic Maps. Endelig kan den downloades i ACCESS-databaseformat (se http://www.ontotext.com/downloads/CycMDB.html). Der arbejdes også videre med at offentliggøre en større del af Cyc-ontologien (6000 begreber) inden for det projekt der hedder OpenCyc (http://www.opencyc.org/releases/).

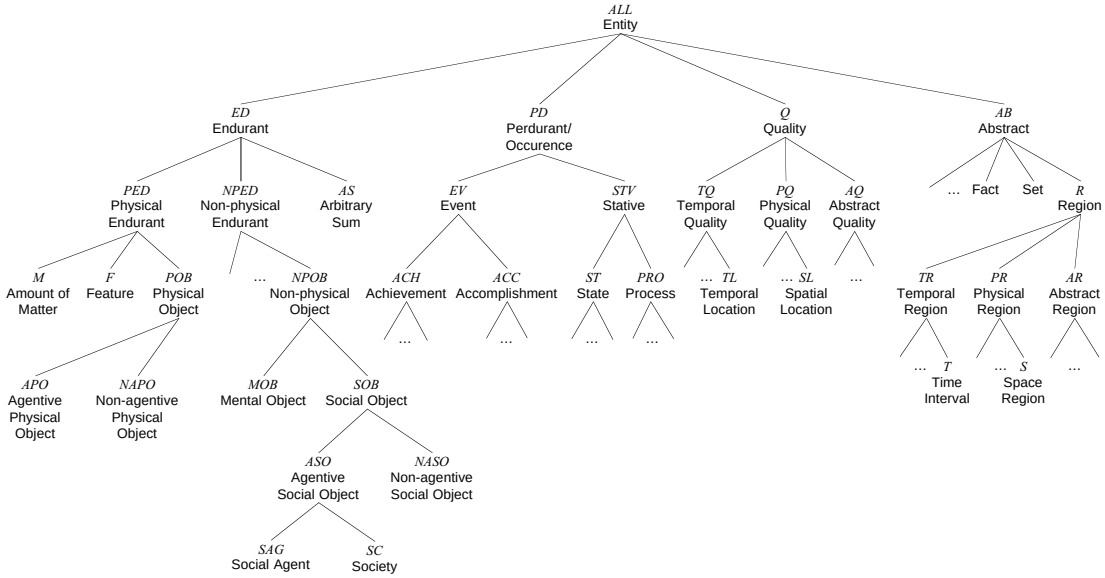
2.3.2 DOLCE (A Descriptive Ontology for Linguistic and Cognitive Engineering) http://www.isib.cnr.it/infor/ontology/DOLCE.html.

Denne ontologi er udviklet ved Laboratory for Applied Ontology, Italien, primært af Guarino og Welty. Der er tale om en formel ontologi bestående af omkring 200 metabegreber med en rig aksiomatisk karakteristik.

Der er taget grundigt stilling til de teoretiske aspekter omkring hvilke ontologiske begreber der er centrale og hvilke der ikke er. Ifølge Masolo et al. (2003:8) er der tale om en kognitivt funderet ontologi som baserer sig på hvorledes vi opfatter ting og ikke nødvendigvis på tings metafysiske eller biologiske egenskaber. Endvidere beskrives og defineres begreberne ud fra forskellige metaegenskaber hvoraf rigiditet (rigid properties) regnes for en særlig vigtig egenskab for ontologiens kernebegreber. Hvis et begreb besidder rigide egenskaber betyder det bl.a. at det bibeholder disse over tid; begrebet Person besidder fx rigide egenskaber i modsætning til begrebet Student. Den såkaldte OntoClean Methodology udviklet af Guarino & Welty (2002) er anvendt på DOLCE og skulle netop hjælpe brugeren med at træffe valg omkring begrebers status i ontologien.

Der skelnes på øverste niveau i ontologien mellem Endurant og Perdurant. Også for disse to begrebskategorier er deres tidsmæssige karakteristik central. Endurants *er i tid*, (fx Person), mens Perdurants *foregår i tid* (fx State). Basiskategorierne i DOLCE kan ses i figur 2.3.

Til begreberne i DOLCE-ontologien hører i øvrigt en rig aksiomatisk karakteristik udtrykt i førsteordenslogik.

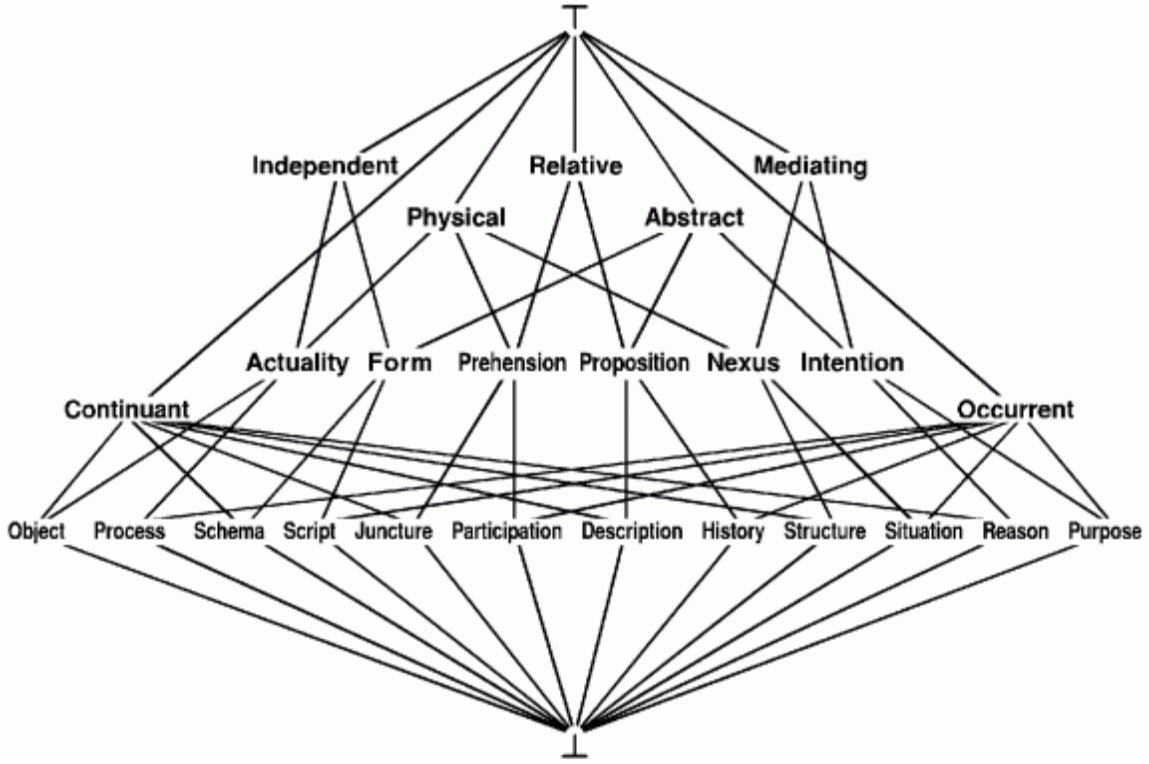


Figur 2.3: Basiskategorier i DOLCE (Masolo et al. 2003:9)

Der arbejdes i øjeblikket på at indarbejde WordNet i DOLCE-ontologien.
DOLCE kan downloades i formaterne DAML+OIL-format, RDF, XML og OWL. Ifølge hjemmesiden er ontologien under anvendelse i flere praktiske sammenhænge, bl.a. i sammenhænge hvori den er integreret med domænespecifikke ontologier.

2.3.3 KR Ontology (Sowas Knowledge Representation Ontology) **http://users.bestweb.net/~sowa/ontology/.**

Denne ontologi er en udløber af John F. Sowas bog Knowledge Representation (Sowa 2000). Ontologiens kategorier er baseret på traditioner fra logik, lingvistik, filosofi og kunstig intelligens. Relationerne i Sowas ontologi beskrives ved hjælp af såkaldte *konceptuelle grafer* som bl.a. udtrykkes i en gitterstruktur og altså ikke i rene taksonomiske strukturer som man ser i mange andre ontologier. Gitterstrukturen i figur 2.4 illustrerer ontologiens topkategorier.



Figur 2.4: Basiskategorier i KR-Ontologien

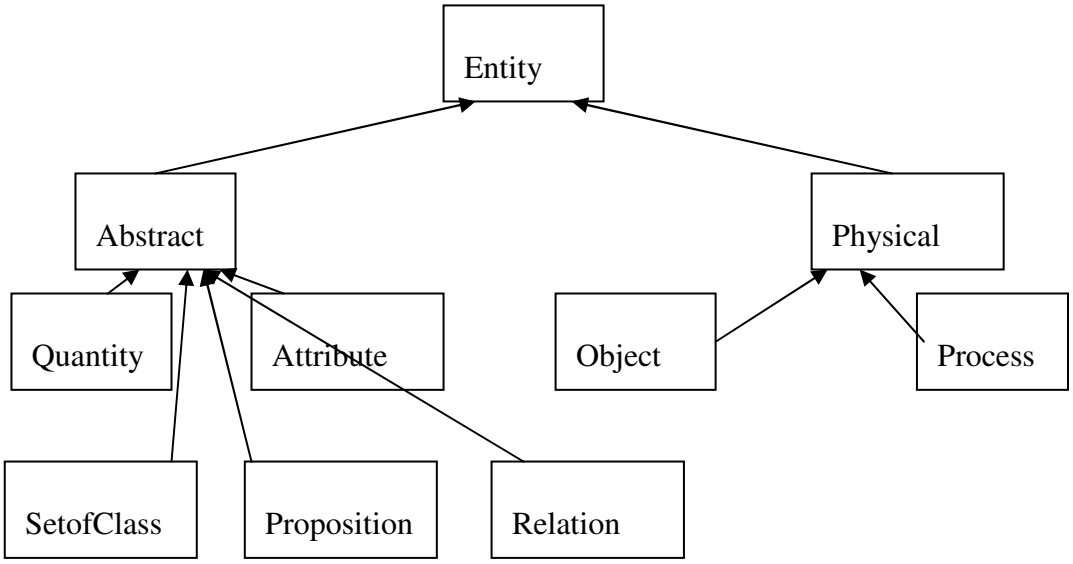
Ontologien er dynamisk sådan at forstå at der ikke er tale om et endeligt sæt af kategorier, men om et sæt af distinktive træk fra hvilket gitteret dynamisk kan genereres. Som ved DOLCE-ontologien fremstår de teoretiske aspekter omkring opbygningen af ontologien som særligt væsentlige og velbeskrevne, og flere af de angivne distinktioner ligner de distinktioner der findes i DOLCE-ontologien, fx skellet mellem continuant og occurrent som er mere eller mindre parallelle til hhv. Endurant og Perdurants i DOLCE.
KR-ontologiens dynamiske egenskaber er muligvis årsagen til at det ikke fremgår hvor omfattende ontologien er og i hvilken grad den kan downloades til brug i praktiske anvendelser.
Til repræsentation af ontologien anvendes det såkaldte Knowledge Interchange Format (KIF) (se afsnit 4.1.3).

2.3.4 SUMO (Suggested Upper Merged Ontology)

http://suo.ieee.org http://ontology.teknowledge.com/.

SUMO er en nyligt udviklet ontologi udarbejdet under IEEE[11] Standard Upper Ontology Working Group. Målet for denne arbejdsgruppe er at udvikle en standardontologi som kan understøtte udveksling af data, informationssøgning, automatisk inferens samt natursprogsbehandling.
SUMO udgør en top-ontologi bestående af 1000 metabegreber suppleret med et antal aksiomer. Aksiomerne er udtrykt i førsteordenslogik.

Begreberne spænder fra generelle begreber som fx Quantity til relativt specifikke som fx Bird. Begreberne er sammensat i et en-dimensionelt hierarki med top-knuden Entity, som repræsenterer det mest generelle begreb og som derfra deler sig i de klassiske grupperinger i form af Abstract og Physical som det ses i figur 2.5.



Figur 2.5 Basiskategorier i SUMO (Sevcenko 2000:2)

For at knytte an til den sproglige dimension er SUMO integreret med WordNet (se afsnit 2.5.2). Denne integration realiseres ved at der for hver synonymimængde (synset) i WordNet tilskrives et SUMO-overbegreb. For synset {*animal, beast, fauna*} tilskrives fx SUMO-begrebet Animal.

SUMO-ontologien kan downloades i følgende formater/værktøjer: DAML, LOOM, XML, og Protege (pprj, pont, pins filer) (se kapitel 4).

[11] IEEE står for of Electrical and Electronics Engineers, Inc og er en ikke-kommerciel forening for omkring 380.000 medlemmer i 150 lande.

2.4 Domæneontologier til informationssystemer

Domæneontologier udvikles ofte som komponenter til konkrete projekter og er i mange tilfælde ikke offentligt tilgængelige. Nedenfor gives dog et eksempel på en offentlig tilgængelig museumsontologi samt link til et websted hvorfra man kan finde eksempler på andre domæneontologier som kan downloades.

2.4.1 CIDOC Conceptual Reference Model (CRM)

http://cidoc.ics.forth.gr/ samt Crofts et al 2001.
Ontologien der er ganske omfattende, er udviklet over en årrække af CIDOC Documentation Standards Working Group og er beregnet til beskrivelse af museumsdokumentation og dækker således et bredt område inden for kulturarv. Det er en ontologi der er udviklet ud fra praktiske behov for strukturering af begreber inden for museumsverdenen og der er tale om en formelt baseret ontologi der dels anvender en top-ontologi, dels indeholder definitioner i et formelt sprog. Der arbejdes videre med ontologien i et forsøg på at få gjort den til en ISO-standard.

2.4.2 DAML (DARPA Agent Markup Language)

http://www.daml.org/ samt **http://www.cs.umd.edu/projects/plus/DAML/.**

Formålet med det amerikanske initiativ DAML (DARPA Agent Markup Language) er at udvikle et sprog og nogle værktøjer som kan fungere som standard inden for Semantic Web-teknologien (se Afsnit 4.1.1 om formelle sprog). En anden vigtig pointe ved DAML-initiativet er i ontologisammenhæng at det udgør et samlingssted med links til domænespecifikke ontologier til informationssystemer, alle udtrykt i DAML. Man kan fx downloade et rejseontologi, en vejrontologi eller en organisationsontologi. Alt efter formål kan man så yderligere specificere en sådan ontologi så den passer til de teksttyper man ønsker at behandle. Se i øvrigt også http://www.cs.umd.edu/projects/plus/SHOE/onts/index.html (SHOE for Simple HTML Ontology Extensions) hvorfra flere ontologier kan downloades i det såkaldte SHOE-format, som er et udvidet HTML-format hvor forfatteren kan annottere sine websider.

2.5 Eksempler på lingvistiske ontologier og leksikalske net

2.5.1 Princeton WordNet

http://www.cogsci.princeton.edu/~wn/obtain.shtml og Fellbaum 2000.
WordNet er en elektronisk leksikalsk database bestående af 90.000 engelske begreber struktureret efter psykolingvistiske principper i såkaldte synonymimængder (synsets), hvori indgår ord som er synonymer eller delvise synonymer til hinanden. Den leksikalske database omtales ofte som et leksikalsk net eller en lingvistisk ontologi idet begreberne er forbundet med hinanden ved hjælp af semantiske relationer. Der er tale om et meget finmasket net hvor hver leksikalsk indgang beskrives med dets (ofte mange) underbetydninger. I figur 2.6 ses en af de seks betydninger som *dog* har i

WordNet samt to af de semantiske relationer som man har etableret til andre begreber (der findes en hel række relationstyper).

Sense 1 of *dog*
SYNSET: dog, domestic dog, Canis familiaris
MERONYM[12] flag
HYPERONYM[13] canine, canid
HYPERONYM carnivore
HYPERONYM placental, placental mammal, eutherian mammal
HYPERONYM mammal
HYPERONYM vertebrate, craniate
HYPERONYM chordate
HYPERONYM animal, animate being, beast, brute, creature, fauna
HYPERONYM organism, being
HYPERONYM living thing, animate thing
HYPERONYM object, physical object
HYPERONYM entity

Figur 2.6: betydning 1 af *dog* samt dens hyperonymer (overbegreber) i WordNet

Hvert begreb har udover det i figur 2.6 angivne en definition i almindeligt sprog svarende til definitionen i en ordbog (og er altså *ikke* beskrevet ved hjælp af logiske udsagn).
På grund af sin størrelse af WordNet meget flittigt benyttet i datamatiske sammenhænge; bl.a. linker flere top-ontologier som nævnt op imod WordNet. WordNet kan downloades i en Unix-database eller en windowsdatabase.

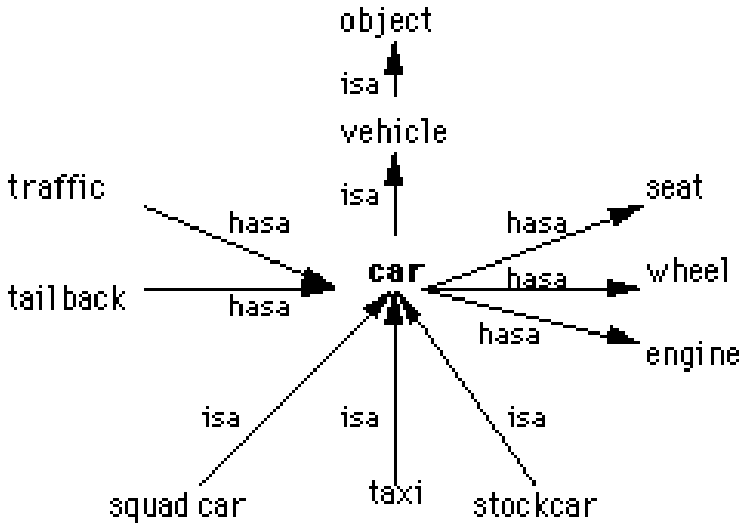
2.5.2 EuroWordNet

http://www.illc.uva.nl/EuroWordNet/ samt Vossen et al. (1999).

EuroWordNet er en multilingval database med leksikalske net på adskillige europæiske sprog (hollandsk, italiensk, spansk, tysk, fransk, tjekkisk og estisk). Der er ca. 20.000 begreber defineret på hvert sprog.

Strukturen i EuroWordNet er den samme som i WordNet idet de forskellige begreber er samlet i såkaldte synonymigruppe (synsets) og forbundet med hinanden via semantiske relationer som det ses for *car* i figur 2.7.

[12] Meronym-relationen svarer til er-del-af eller HAS PART.
[13] Hyperonym svarer til er-overbegreb-til eller IS A,



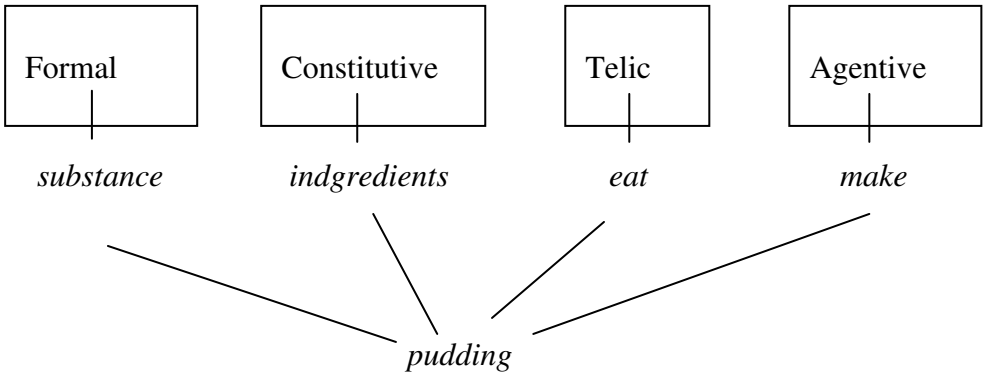
Figur 2.7: semantiske relationer tilknyttet *car* i EuroWordNet

Databasen består af en række leksikalske net for forskellige sprog som er linket til hinanden via det såkaldte Inter-Lingual Index som er baseret på WordNet. EuroWordNet knytter an til en top-ontologi som baserer sig på Lyons hypoteser om 1st, 2nd og 3rd order entities (Lyons 1977) hvor 1st order entities repræsenterer konkrete fysiske ting, 2nd order entities ting som egenskaber, handlinger, processer, tilstand og hændelser og 3rd order entities abstrakte begreber. Endvidere er tilknyttet adskillige domænespecifikke ontologier.

EuroWordNet forhandles via ELRA/ELDA i en Polaris Database-version sammen browserværktøjet Periscope.

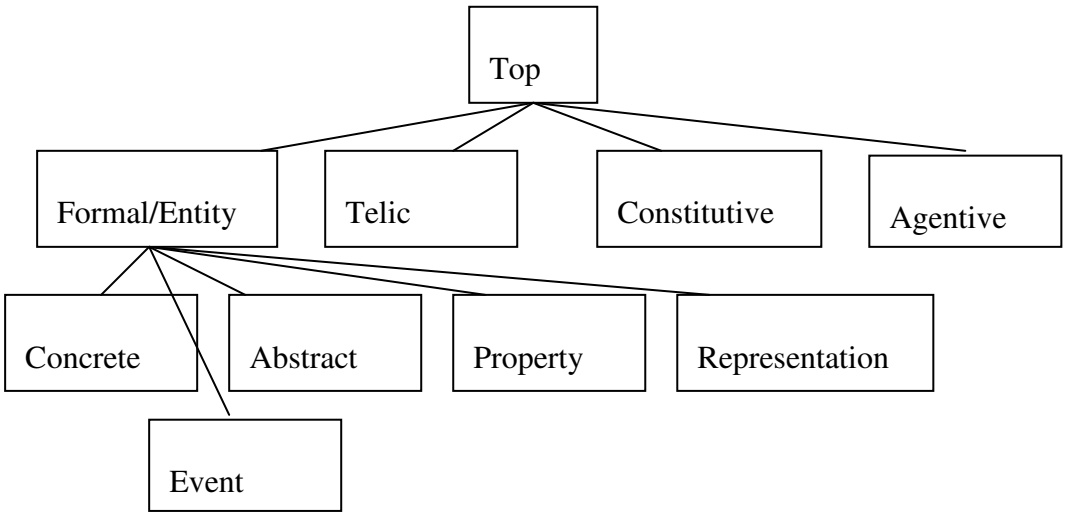
2.5.3 SIMPLE-ontologien

http://www.hltcentral.org/projects/detail.php?acronym=simple og Lenci et al. 2000. Som det antydes af titlen er SIMPLE-ontologien (Semantic Information for Plurilingual Multifunctional Lexica) udviklet som et struktureringsværktøj for udviklingen af multilingvale semantiske ordbøger til forskellige sprogteknologiske formål. Der er altså i udgangspunktet tale om en lingvistisk ontologi som har sondringer som er af lingvistisk relevans. Dette betyder fx at man under begrebet Animal har valgt ikke har følge de biologiske sondringer (som i WordNet) mellem fx pattedyr og krybdyr, men blot underkategoriserer i WaterAnimal, EarthAnimal og AirAnimal, da man har skønnet at denne sondring har mere lingvistisk relevant end den biologiske, som de færreste lægmænd alligevel kender i detaljer.
Ontologien består af 138 metabegreber og har fire dimensioner idet den følger Pustejovskys teori om at begrebers kernesemantik kan beskrives ud fra en firedelt qualiastruktur. I figur 2.8 antydes ontologiens 4-dimensionale struktur idet et begreb som *pudding* nedarver fra alle 4 top-knuder (*pudding* er en slags substans (*substance*) som tilbedes (*make*) ved at blande nogle ingredienser (*ingredients*). Formålet med buddingen er at den skal indtages (*eat*).



Figur 2.8: De fire betydningsdimensioner i begrebet *pudding* (budding) (Lenci et al. 2000:pp 17):

Imidlertid er det under top-knuden Formal at den egentlige basis-ontologi udfolder sig med begrebet Entity som øverste type under Formal. Således er 133 af de 138 begreber i topontologien placeret under Formal. Der anvendes stort set samme distinktioner i basisontologien som i EuroWordNet, nemlig i udgangspunktet Lyons tredelte struktur i 1st, 2nd og 3rd order entiteter. I figur 2.9 ses de øverste kategorier i SIMPLE-ontologien.



Figur 2.9: SIMPLE-Ontologiens øverste kategorier

Hvert af de 12 lande der deltog i SIMPLE-projektet har udviklet en ordbase med omkring 10.000 begreber for deres eget sprog; heriblandt dansk. Den danske ontologi er tilgængelig som sgml-fil (konvertering til databaseformat i ORACLE er i gang) til forskningsformål.

2.5.4 MikroKosmos

Mikrokosmos' egen hjemmeside er nu lukket, men info fås på www.csee.umbchttp://.edu/~dingli1/student/cmsc691k/mikrokosmos.htm samt i Onyshkevych & Nirenburg 1995.

Mikrokosmos-ontologien er en større lingvistisk ontologi (antallet af begrebet oplyses dog ikke) som blev bygget i forbindelse med et praktisk maskinoversættelsessystem ved det nu lukkede Computing Research Laboratory (CRL) i New Mexico State University. Ontologien er bl.a. interessant i en historisk sammenhæng fordi den var en af de første større ontologier der blev bygget med et klart datalingvistisk formål, nemlig maskinoversættelse.

I ontologien skelnes der skarpt mellem sprogspecifik viden – udtrykt i ordbogsdelen – og sprogneutral viden som udtrykkes i ontologidelen. Ords semantik gengives altså dels i ordbogsdelen dels i ontologidelen. Denne skelnen giver også mulighed for at have en 'en til mange'-relation eller en 'mange til en'-relation mellem begreb og ord. Ontologien er blevet anvendt som en slags interlingva mellem flere forskellige sprog i forbindelse med oversættelsen.

2.6 Konkluderende bemærkninger

Der er adskillige initiativer igang der går ud på at udforske metoder til at integrere ontologier; dels metoder til integration af flere af de ovenfor nævnte topontologier, dels integration af flere delvist overlappende domæneontologier (se også afsnit 4.2.2 om værktøjer til dette). Som nævnt er der også foretaget en række forsøg med integration af lingvistiske ontologier og leksikalske net med formelt baserede ontologier.
Der er ingen tvivl om nytten af sådanne sammenligningsinitiativer hvis man ønsker at bevæge sig hen i retningen af standarder på ontologiområdet. Og standarder synes at være en nødvendighed hvis ontologier skal blive reelt rentable at anvende også i praktiske sammenhænge.

Et andet aspekt som vi ikke vil komme yderligere ind på i denne rapport, men som er vigtig at udforske hvis ontologier skal blive praktisk anvendelige i en større målestok, er semiautomatisk opbygning af ontologier. Initiativer på dette område sker dels ved hjælp af statistisk baserede modeller hvor man ser på hvordan ord statistisk optræder sammen i tekst, for på basis heraf at udlede noget om hvorvidt de befinder sig tæt på hinanden ontologisk set. En andel tilgang baserer sig på lingvistiske ressourcer i form af ordbøger; semiautomatisk ontologiopbygning ud fra ordbogsdefinitioner og andre oplysninger i ordbøger, er således også et forskningsområde som synes perspektivrigt.

I relation til forskningsprojektet VID er der umiddelbart flere af de beskrevne ontologier der kan have interesse. Især synes DOLCE-ontologien interessant fordi det er en formel ontologi der forholder sig meget klart til lingvistisk ontologi og som har et klart teoretisk udgangspunkt; den er desuden let tilgængelig også i formalismer som er relevante for os (se Kapitel 4). Men også SIMPLE-ontologien er særlig interessant af

flere årsager. Dels er det en lingvistisk ontologi der bygger Pustejovskys flerdimensionelle semantiske struktur som synes perspektivrig bl.a. i forbindelse med entydiggørelse. Dels har SIMPLE-ontologien den fordel at der knytter sig en ordbase til den med 10.000 danske begreber som kan finde umiddelbar anvendelse i et projekt hvor der arbejdes med danske tekster. WordNet findes derimod ikke på dansk; alligevel er basen muligvis interessant simpelthen på grund af sin høje dækningsgrad; om ikke andet så som sammenligning eller støtte for dansk. Endelig bør der for de domænespecifikke ontologier der udvikles i VID, i så vid udstrækning som muligt tages udgangspunkt i eksisterende domæneontologier som kan downloades fra nettet. Man kan så forestille sig at disse kan blive yderligere specificeret i forhold til det konkrete tekstmateriale.

3 Metadata til beskrivelse af dokumenter

Inden for arkiv-, biblioteks- og museumsverdenen har man lange traditioner for at gemme ting på en systematisk måde. Her har man systematiseret og kategoriseret samlinger af dokumenter, bøger, musik, kunst mm. ved hjælp af kartotekskort så man effektivt kunne søge og finde information. Til dette formål har man derfor en lang tradition for strukturering af begreber i form af ontologier, tesaursusser og andre klassifikationssystemer, og man har haft brug for oplysningstyper der beskriver fx et dokuments form og indhold ud fra nogle fastlagte kriterier.

Udviklingen af standardiserede metadata omhandler bl.a. spørgsmålet om hvordan man kan overføre metoder fra de traditionelle områder til alle tænkelige former for elektronisk information der befinder sig alle tænkelige steder.

3.1 Hvad er metadata

Kort sagt betyder metadata *data om data*, og er oplysninger der beskriver fx et dokuments form og indhold. I dette kapitel definerer vi metadata som *data om elektroniske dokumenter*[14] fordi det oftest er sådan metadata bliver anvendt. Vi går ikke under dokumentniveau og taler derfor ikke om metadata der fx beskriver afsnit. I praksis er metadata supplerende oplysninger om forfatter, titel, dato, format, emne osv., som man tilfører dokumentet.

Metadataenes funktion er at give så udtømmende beskrivelse af dokumentet som nødvendigt for at kunne finde præcis dét dokument igen. Der findes derfor forskellige typer af metadata alt efter hvilket domæne og hvilken disciplin man befinder sig inden for, og efter hvor man gemmer eller søger efter dokumentet, om det er i en intern database, på et intranet eller på internettet. Fx skal de metadata man bruger i museumsverdenen til at karakterisere objekter med, være af en anden art og muligvis mere detaljerede end dem man bruger i biblioteksverdenen. De eksisterende metadatasystemer er altså af varierende størrelse, struktur, indhold og finkornethed.

3.1.1 Metadatasystemer

Enhver kan definere det metadatasystem de har brug for i en given situation. Hvis man fx har en database med dokumenter eller "databaseposter" om plantefrø, ville relevante metadata kunne inkludere elementer som illustreret i figur 3.1.

[14] *Elektroniske dokumenter* skal her forstås bredt som alle elektroniske resurser og kan være tekstdokumenter såvel som billeder, musik eller film.

Plantenavn	Såning	Høst	Trivsel	Avl	mm.
Latinsk	Tid	Tid	ift. lys	Sted	…
Dansk	Sted		ift. vand	Forhold	…

Figur 3.1 Fiktivt skema over metadata til plantefrø

Man ville så kunne søge og finde bestemte slags frø i forhold til fx skyggetålende planter. Det er klart at sådan et metadatasystem kun vil kunne bruges i forhold til plantefrø og komme til kort over for en database der indeholder jobannoncer eller juridiske tekster. Hvis man vil udveksle information, genbruge information eller søge på tværs af databaser eller hjemmesider, er det derfor nødvendigt at blive enige om standarder der gør metadataene forenelige.

Generelt set kan et metadatasystem indeholde oplysninger der beskriver et dokuments indhold, dets form, tilvejebringelse mm. og oplysninger der beskriver strukturen i dokumentet samt dets relationer til andre dokumenter. Altså oplysninger om indhold, kontekst og struktur. Det er dog forskelligt fra applikationsområde til applikationsområde hvilke af de tre typer der er centrale, og hvilke der overhovedet er med. I figur 3.2 kan man se eksempler på de tre typer metadata fra forskellige metadatasystemer (bemærk at det ikke er udtømmende lister men blot eksempler):

Metadata-systemer	Metadatatyper		
	Indhold	**Kontekst**	**Struktur**
BibTeX	-	Author	Crossref
		Editor	
		Publisher	
		Howpublished	
CDWA	Subject Matter - Description	Creation Creator Identity	Related Visual Dokumentation –Relation Type
	Subject Matter - Description Indexing Terms	Creation - Creator Role	Related Visual Dokumentation –Image Type
	Subject Matter -Identification Indexing Terms	Creation Date	Related Visual Dokumentation –Image Meassurements
	Subject Matter -Interpretation Indexing Terms	Creation Place	Related Visual Dokumentation –Image Format
Dublin Core	Subject	Creator	Identifier
	Description	Publisher	Relation

Figur 3.2 Eksempler på metadatatyper fra 3 forskellige metadatasystemer

BibTeX er skabt til at generere bibliografier i LaTeX og indeholder derfor ikke indholdsbeskrivende metadata. BibTeX består af 24 metadata.
CDWA (*Categories for the Description of Works of Art*) er på den anden side et meget omfattende system med en rig og detaljeret udtrykskraft i forhold til beskrivelse af kunstgenstande. CDWA består af 26 hovedkategorier og 94 underkategorier.
Det sidste metadatasystem, som vi vil undersøge nærmere i det følgende afsnit, er Dublin Core, der er skabt som et minimalt metadatasystem der kan bruges til søgning og på tværs af fagdomæner.

3.2 Dublin Core Metadata

Dublin Core (DC) er en standard der dels kan bruges på tværs af fagområder og dels kan bruges til søgning i såvel databaser som på internettet. Intensionerne bag DC har været at få et metadatasystem til internettet der både er tilstrækkelig detaljeret til at kunne rumme de nødvendige oplysninger fra forskellige områder, og samtidig er så begrænset og simpelt at det er let og overskueligt at bruge for alle. Enkeltheden gør det både billigere at tilskrive metadata til et dokument og lettere at udveksle metadata mellem forskellige systemer.

3.2.1 Dublin Core element set

DC består af 15 forskellige metadataelementer der er blevet fastlagt gennem diskussioner i en international gruppe bestående at repræsentanter fra biblioteker, museer, tekstkodningsfaget, internetstandarder og beslægtede miljøer. Det første *DC matadata element set* blev fastlagt i 1995, og i 2003 blev det nuværende DC godkendt som ISO-standard - ISO 15836.
Hvert element er valgfrit, kan gentages og kan desuden præciseres ved hjælp af en række fastlagte *kvalifikatorer*. Da DC er skabt til søgning og specielt til søgning på internettet, består det hovedsageligt af metadata som beskriver dokumenters indhold.

Skemaet i figur 3.3 er en beskrivelse af de 15 elementer i Dublin Core:

Navn	**1 Titel** (DC.Title)
Definition	Navnet på dokumentet, givet af forfatteren
Kvalifikator	*Alternative*, et navn der bruges som alternativ til den formelle titel
Navn	**2 Ophav** (DC.Creator)
Definition	Personer eller organisationer der er ansvarlig for dokumentets intellektuelle indhold
Kvalifikator	÷
Navn	**3 Emne** (DC.Subject)
Definition	Det dokumentet handler om, udtrykt vha. koder fra anerkendte klassifikationsystemer fx DDC, UDC. Og/eller stikord eller sætninger der beskriver indholdet.
Kvalifikator	÷
Navn	**4 Beskrivelse** (DC.Description)
Definition	En redegørelse af indholdet
Kvalifikator	*Table of Content*, en indholdsfortegnelse over dokumentets underafsnit *Abstract*, et resumé af dokumentets indhold
Navn	**5 Udgiver** (DC.Publisher)
Definition	Den enhed der er ansvarlig for at stille dokumentet til rådighed i den nuværende form fx person, forlag eller firma
Kvalifikator	÷
Navn	**6 Anden bidragyder** (DC.Contributors)
Definition	Person eller organisation der har bidraget sekundært til dokumentets indhold fx redaktør, oversætter eller illustrator
Kvalifikator	÷
Navn	**7 Dato** (DC.Date)
Definition	Den dato dokumentet er oprettet eller gjort tilgængeligt
Kvalifikator	*Created*, den dato hvor dokumentet er oprettet/skabt *Valid*, dato, evt. periode, hvor dokumentet er

	gyldigt *Available*, dato, evt. periode, hvor dokumentet er tilgængelig *Issued*, dato for den formelle udstedelse/publikation af dokumentet *Modified*, dato er hvornår dokumentet er ændret
Navn	**8 Resursetype** (DC.Type)
Definition	Dokumentets udtryksform eller genre, fx hjemmeside, manual, digt, ordbog, sagsmappe mm.
Kvalifikator	÷
Navn	**9 Format** (DC.Format)
Definition	Dokumentets dataformat, fx text/html, Postscript, jpg mm.
Kvalifikator	*Extent*, dokumentets størrelse og/el. varighed *Medium*, dokumentets materielle og/el. fysiske beliggenhed *IMT*, internetformat (MIME types) fx html, xml
Navn	**10 Identifikator** (DC.Identifier)
Definition	En unik identifikation af dokumentet fx ID-nr el. URL
Kvalifikator	÷
Navn	**11 Kilde** (DC.Source)
Definition	Det originale værk som dokumentet bygger på
Kvalifikator	÷
Navn	**12 Sprog** (DC.Language)
Definition	Sproget som dokumentet er skrevet på. Ved flere sprog gentages feltet
Kvalifikator	÷
Navn	**13 Relationer** (DC.Relation)
Definition	Dokumentets relationer til andre dokumenter
Kvalifikator	*Is Version Of*, dokumentets version af referencedokumentet *Has Version*, dokumentets andre versioner *Is Replaced By*, dokumentet er erstattet af et andet *Replaces*, dokumentet er erstattet af flg. andre dokumenter *Is Required By*, dokumentet er fysisk el. logisk påkrævet af andre dokumenter *Requires*, dokumentet kræver flg. dokumenter *Is Part Of*, dokumentet er en fysisk el. logisk del af flg. dokument *Has Part*, dokumentet inkluderer flg. andre dokumenter *Is Referenced By*, flg. dokumenter refererer til dokumentet *References*, dokumentet refererer til flg. dokumenter *Is Format Of*, dokumentet har samme indhold som, men et andet format end flg. dokument *Has Format*, dokumentet eksisterede før flg. dokument med samme indhold, men forskelligt

	format
Navn	**14 Dækning** (DC.Coverage)
Definition	Den geografiske og/el. tidsmæssige afgrænsning af dokumentets intellektuelle indhold
Kvalifikator	*Spatial*, de rummæssige karakteristika ved dokumentets intellektuelle indhold
Navn	**15 Rettigheder** (DC.Rights)
Definition	Et link til en copyright-erklæring
Kvalifikator	÷

Figur 3.3 De 15 Dublin Core Metadata m. kvalifikatorer

I afsnit 3.3.1 ses et eksempel på hvordan metadataene for dette dokument ser ud.

3.2.2 Brug af Dublin Core

Dublin Core Metadata Initiative anbefaler at bruge DC som udgangspunkt når man skal opbygge et fagspecifikt metadatasystem. En af de store fordele ved at bygge på en standard er at man kan udveksle oplysninger med andre anerkendte, fagspecifikke systemer.
Man arbejder fx i det offentlige Danmark, OIO, med at gøre DC til den standard man bruger inden for al offentlig forvaltning. Dette er parallelt med initiativer i forskellige lande så som Australien, UK, Irland, USA, Canada samt i EU. Inden for OIO har man udarbejdet en dansk DC metadata-kerne der skal bruges til udveksling af dokumenter, blanketter og sager mellem EDH/ESDH-systemer (Elektronisk Sags og DokumentHåndtering).

Selvom det ikke har været formålet, er det pt. mest offentlige myndigheder der bruger DC. En af grundene er den ”hønen-og-ægget”-problematik der er omkring understøttelsen af DC på internettet. Private firmaer vil ikke bruge DC hvis de tilgængelige søgemaskiner ikke understøtter og bruger matadataene, og søgemaskinerne vil ikke anvende metadata i søgningen hvis brugerne ikke annoterer deres dokumenter med metadata. Desuden er der tendenser til at fylde *spam* i nogle af metadataene for at få sine dokumenter højt placeret blandt søgehits.
Disse problemer arbejdes der på at løse på internationalt plan gennem World Wide Web Corsortium (W3C) og Dublin Core Metadata Initiative.

3.3 Automatisk behandling af Dublin Core

Dette afsnit handler om generering af Dublin Core metadata, mens anvendelsen af metadata til indeksering, kategorisering og søgning ikke vil blive behandlet. Først i dette kapitel definerede vi metadata som *data om elektroniske dokumenter*. I dette afsnit indskrænker vi betydningen til *data om tekstdokumenter* da det er tekstanalyse vi arbejder med og ikke fx billedanalyse.

3.3.1 Repræsentation af Dublin Core

I 1996, året efter at man havde vedtaget den første version af DC, besluttede man at lave en arkitektur for metadata der kunne sikre udvekslingen af forskellige typer metadata mellem forskellige systemer og applikationer. Kritikken af DC var at den til tider var for snæver, at man kunne have brug for andre typer af metadata end dem DC kan tilbyde. Det kunne fx være metadata om betingelser eller evaluering. Den vedtagne arkitektur, The Warwick Framework, er designet til at kunne indeholde DC såvel som andre typer metadata og tager ikke stilling til metadataenes indhold, kun til deres udformning.
I 1997 introducerede World Wide Web Consortium (W3C) så RDF som er en instantiering af The Warwick Framework til internettet.

I figur 3.4 ses DC-metadata for dette dokument implementeret i RDF. En nærmere beskrivelse af RDF ses i kapitel 4 afsnit 4.1.2.

```
<?xml version="1.0"?>
<!DOCTYPE rdf:RDF >
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
         xmlns:dc ="http://purl.org/dc/elements/1.1/">
 <rdf:Description rdf:about="http://cst.dk/vid/onto.1.doc[15]">
   <dc:title>Ontologier og metadata til brug for søgning i tekst</dc:title>
   <dc:creator>Bolette Sandford Pedersen, Costanza Navarretta,
                Dorte Haltrup Hansen</dc:creator>
   <dc:subject>ontologi, dokument, metadata, format, Wordnet, søgning,
                internet, metadatasystem, førsteordenslogik, database,
                standard </dc:subject>
   <dc:description> State of the art for ontologier, formelle sprog, værktøj til
                at opbygge og redigere i ontologier samt metadata til
                beskrivelse at dokumenter</dc:description>
   <dc:publisher>VID, Center for Sprogteknologi</dc:publisher>
   <dc:contributor>Projektet VID</dc:contributor>
   <dc:date>24-09 2003</dc:date>
   <dc:type> Rapport</dc:type>
   <dc:format> Word</dc:format>
   <dc:format> 360471 bytes</dc:format>
   <dc:identifier>"http://cst.dk/vid/onto.1.doc"</dc:identifier>
   <dc:source> </dc:source>
   <dc:language>da</dc:language>

   <dc:relation> Gómez-Pérez, A. et al. eds, 2002. OntoWeb: A survey on

       ontology tools, Deliverable 1.3. IST-2000-29243.

       http://babage.dia.fi.upm.es/ontoweb/wp1/OntoRoadMap/index.htmlHendler, J. Agents
       and the Semantic Web. University of Maryland.
       http://www.cs.umd.edu/~hendler/AgentWeb.html,</dc:relation>
   <dc:coverage>2003</dc:coverage>
```

[15] Vi antager at " http://cst.dk/vid/onto.1.doc" er adressen på dette dokument

```
   <dc:rights> </dc:rights>
    </rdf:Description>
</rdf:RDF>
```

Figur 3.4 Metadata for dette dokument udtrykt i RDF

De uudfyldte metadata, nemlig: *dc:source* og *dc:rights* har ikke været relevante/mulige at udfylde for dette dokument. Desuden er feltet *dc:relation* kun delvis udfyldt da der kun er medtaget to referencer.

3.3.2 Værktøjer til generering af Dublin Core

Der findes en række værktøjer på internettet man kan bruge til at generere metadata for et givent dokument. Generelt for disse værktøjer er at de er udformet som blanketter som brugeren selv skal udfylde. Kun et produkt, nemlig det engelske *DC-dot*, skaber automatisk metadata som brugeren kan acceptere eller redigere i. Resultatet, de genererede metadata, kan så ved hjælp af *cut & paste* sættes ind i ens dokument. Det man får ved at bruge disse værktøjer er altså metadata der er opmærket med XML-syntaks så de fortolkes som metadata og ikke som almindelig tekst.

Den ønskelige situation ville være at metadata blev skabt fuldautomatisk, så den enkelte forfatter af et dokument ikke behøvede at spekulere på dem. For engelsk er man er nået et stykke ad vejen med *DC-dot*; men også her er det kun visse metadata der kan skabes automatisk, nemlig: titel, emne, dato, resursetype og format.

For danske dokumenter har man mulighed for at få skabt syntaktisk velformede metadata, men ikke at få dem skabt automatisk. I næste afsnit vil vi beskrive hvordan man kunne forestille sig at sprogteknologiske værktøjer ville kunne bruges til det.

3.3.3 Automatisk generering af danske Dublin Core metadata

Måske er det ikke indlysende at man ikke blot kan bruge det engelske system *DC-dot* til at generere danske metadata. For de tekniske metadata så som *Date, Type* og *Format*, vil det da også være muligt; men for metadata der kræver at man går ind i dokumentets tekst og ser på det intellektuelle indhold, fx *Subject* og *Description*, er det nødvendigt at kende det pågældende sprog.

Subject kan udtrykkes ved hjælp af emnekoder der bindes op til et klassifikationssystem og/eller ved hjælp af nøgleord der karakteriserer dokumentets indhold.

I Figur 3.5 ses nøgleord der er genereret fuldautomatisk til dette dokument. Ordene i første kolonne er skabt af sprogteknologiske værktøjer; mens ordene i anden kolonne kommer fra det engelske system *DC-dot*. Skemaet skulle med tydelighed vise hvorfor man ikke kan bruge et program der er lavet til engelske dokumenter, til at generere danske nøgleord.

Nøgleord for dette dokument	
Genereret af sprogteknologisk værktøj	**Genereret af metadata-editoren DC-dot**
ontologi	WordNet
dokument	selektiv
sprog	Online Presentation
værktøj	sprog
metadata	www.metamaten.dk
begreb	rdf:RDF
system	Merged
egenskab	Znalosti
relation	Udgiver
OWL	car
XML	Extent
resurse	Registry
format	dog
element	Ontology
Dublin_Core	vejer
data	DC-dot
RDF	anstrengelser
figur	qualia
eksempel	CDWA
Wordnet	at sammenflette
indhold	Schemas
definition	OntoGenerator
søgning	for Plurilingual
internet	ontologi
beskrivelse	Engineering
metadatasystem	ontologistandarder
klasse	sker
domæne	rdf
DL	BibTeX
DC	Dublin

Fig. 3.5 Nøgleord for dette dokument fundet fuldautomatisk

Vi har ikke kunnet finde dokumentation for hvordan *DC-dot* generere nøgleord; men formodentligt selekteres blandt andet ord der er skrevet med stor og ord der genkendes som engelske. Hvorfor ord som *selektiv, vejer* og *anstrengelser* er med på listen er derimod en gåde.

Den sprogteknologiske proces der er brugt, består af:

En tokeniser, der opdeler teksten i relevante tokens som er tal, ord, flerordsforbindelser, forkortelser, tegn mv.

fx …. systemer **i_forbindelse_med** digital forvaltning …

En navnegenkender, der markerer og evt. kategorisere tekstens navne

fx Hvordan bruges **Dublin_Core/**NAVN

En POS-tagger, der tildeler ordklasser til alle ord,

fx …. systemer/**N** i_forbindelse_med/**PRÆP** digital/**ADJ** forvaltning/**N**.

En lemmatiser, der finder grundformen af alle ord,

fx 287 **en** (139 en, 148 et)
330 **være** (305 er, 4 var, 14 være, 1 værende, 6 været)

Efter disse processer er de 30 hyppigste substantiver og egennavne udtrukket som nøgleord. Det kan diskuteres om man skal sætte grænsen ved 30, om den skulle være lavere eller om man skulle frasortere hyppige almenord som fx *eksempel*.

I en interaktiv proces, som i *DC-dot*, vil det være muligt at korrigere de fundne nøgleord – tilføje nye og slette andre. Man kunne fx slette: *eksempel, relation, figur, beskrivelse, klasse* og tilføje: *DAML+OIL, Topic_Maps, standard* og *web*.

Hvordan man kommer fra nøgleord til en emnekode er en mere kompleks sag – det er et område man arbejder med inden for automatisk klassifikation; men som vi ikke vil komme nærmere ind på her.

3.4 Konkluderende bemærkninger

Hvis man skal opbygge et metadatasystem til at beskrive danske dokumenter, er det oplagt at man tager udgangspunkt i ISO-standarden *Dublin Core metadata element set*. Fordelen ved at bruge denne standard er for det første at den ikke er fagspecifik, for det andet at den er let at anvende for ikke-specialister, og sidst at man med den kan udveksle, genbruge og søge information på tværs af databaser og hjemmesider. Er Dublin Core for begrænset til at udtrykke de oplysninger man har brug for, kan man gøre brug af metadata fra andre metadatasystemer gennem RDF-fomalismen. Endelig er det vigtigt at understrege at der i metadataene kan refereres til begreber som indgår i ontologier og tesaurusser (som det blev illustreret fx under DC.Subject hvor der ofte refereres til anerkendte klassifikationssystemer). Dette kan udnyttes målrettet inden for bestemte domæner hvor domæneontologien udvikles til helt specifikke formål, fx til at afbilde organiseringen eller arbejdsgangen i en virksomhed.

Som det blev illustreret i kapitlet er det desuden anbefalelsesværdigt at man bruger sprogteknologiske værktøjer som er tilpasset dansk til at støtte generering af de metadata der beskriver dokumentets semantiske indhold, fordi metadataene i så tilfælde generelt bliver af bedre kvalitet.

Først i dette kapitel definerede vi metadata på dokumentniveau; men man kunne også forestille sig at man havde brug for metadata til at beskrive fx det semantiske indhold (altså primært DC.Subject) på afsnitniveau. Ønsker en virksomhed fx at optimere vedligeholdelsen og tilpasningen af standarddokumenter således at ændringer kun skal foretages et sted, vil en indholdsmæssig opmærkning af tekst*stumper* være relevant.

Problemet er at det er vanskeligt at lave standarder der går under dokumentniveau og gælder for alle typer dokumenter og alle slags afsnit. Dette betyder ofte at man frit definerer nye metadata eller *tags* som så udelukkende finder anvendelse inden for den specifikke applikation. Et mere attraktivt alternativ – som i højere grad sikrer genanvendelighed - kunne dog være at bruge sammme metadatasystem som gælder for hele dokumenter og altså laver beskrivelse af dokumentet på mikro- såvel som på makroniveau.

4 Formelle sprog og værktøjer

4.1 Formelle sprog

Som nævnt i kapitel 2 har ontologier været implementeret i forskellige typer formelle sprog. Først for nyligt er der blevet defineret standardsprog til modellering af ontologier og protokoller til at udveksle eksisterende ontologier er blevet udarbejdet. Initiativet Semantic Web har desuden sat gang i udviklingen af standarder inden for Web-baserede og ontologirelaterede teknologier. I afsnit 4.1.1 gennemgår vi kort de forskellige typer formelle sprog som bliver anvendt til at repræsentere viden. I afsnit 4.1.2 beskrives Web-baserede formelle sprog til at beskrive metadata og ontologier som er blevet udviklet til at være en standard eller som er blevet til de-facto standarder fordi de er blevet anvendt af flere organisationer i adskillige projekter. I afsnit 4.1.3 beskrives kort nogle af de tiltag der er gjort for at definere protokoller til udveksling af eksisterende ontologier, som er implementeret i forskellige formelle sprog. I afsnit 4.2 gennemgås nogle værktøjer til at opbygge, editere og udveksle ontologier og endelig, i afsnit 4.3, gives en konklusion.

4.1.1 Traditionelle formelle sprog til videnrepræsentation

De fleste formelle sprog til at repræsentere viden er blevet udviklet til såkaldte videnbaserede systemer. Disse systemer anvender domænespecifik viden implementeret i ontologier, som også bliver kaldet videnbaser.

Formelle sprog til videnrepræsentation følger en eller flere af følgende repræsentationsmodeller: regelbaserede, logikbaserede eller objektorienterede modeller og er derfor kendt som regelbaserede, logikbaserede og objektorienterede sprog.

Regelbaserede sprog repræsenterer viden med få *facts* og en mængde regler til at beregne hvad man kan bruge disse *facts* til og hvad man kan konkludere fra dem. Regelbaserede sprog er især blevet anvendt i ekspertsystemer til at beskrive små domænespecifikke ontologier. Da viden repræsenteres i både facts og regler er det svært at overskue videnmodellen bagved regelbaserede sprog. Disse er også ufleksible fordi man bliver nødt til at revidere reglerne hver gang man skal editere en ontologi. Regelbaserede sprog egnes derfor ikke til at beskrive større ontologier og er ikke særligt udbredte i dag.

Logikbaserede sprog bygger på første-ordenslogik. På grund af deres natur har disse sprog en klart defineret syntaks og semantik. Da første-ordenslogik ikke kan beskrive fænomener som tid og modalitet, har man defineret forskellige typer logiksprog som udvider første-ordenslogik på en passende måde. Der findes mange typer logiksprog, fx modallogik, temporallogik og defaultlogik.

Objektorienterede sprog adskiller sig fra regelbaserede sprog idet viden repræsenteres deklarativt i stedet for proceduralt. I objektorienterede sprog organiseres begreber i klasser (*classes*) og i underklasser (*subclasses*). Begreber i en underklasse arver egenskaber fra overklasserne. De første objektorienterede videnrepræsentationssprog er blevet udviklet til at modellere viden i de såkaldte semantiske net. Disse net er simple

grafer hvis knuder er begreber, mens buer mellem knuderne repræsenterer relationerne mellem begreberne. Det vigtigste relation i semantiske net er *subclass* som i store træk svarer til is_a-relationen beskrevet i afsnit 2.2. I nyere objektorienterede videnrepræsentationssprog kaldes klasser for f*rames* og disse sprog er derfor også kendt som frame-baserede sprog. I frame-baserede sprog beskrives et begreb i en *frame* som også indeholder begrebets eventuelle attributter. Attributter kaldes *slots* mens attributværdier kaldes *facets*.

De mest anvendte formelle sprog til at beskrive ontologier i dag er hybride sprog kendt under betegnelse beskrivelseslogik (*Description Logics* eller *DL*). Beskrivelseslogiksprog er hybride fordi de bygger på en frame-baseret model og anvender logik. DL-sprog er beslægtet med sprog som CycL og Loom (MacGregor, 1999) som var nogle af de mest kendte videnrepræsentationssprog i halvfemserne. Det bør bemærkes at alle implementerede formelle sprog til videnrepræsentation ikke kun beskriver ontologierne, men også indeholder inferensmekanismer til at foretage logiske slutninger på de beskrevne data.

4.1.2 Web-baserede formelle sprog til videnrepræsentation

Alle formelle sprog som er blevet udviklet til at implementere ontologier til internettet, anvender XML-baseret teknologi og bliver løbende videreudviklet. Formelle sprog til at beskrive viden på internettet har en varierende udtryksrigdom. De mest simple og generelle sprog er blevet udviklet til at beskrive resurser på internettet med en simpel mængde af metadata, som beskrevet i kapitel 3. De mest komplekse og udtryksrige sprog der er udviklet til at modellere komplekse ontologier, er opbygget som en specialisering af disse generelle sprog og har samme udtrykskraft som de traditionelle videnrepræsentationssprog som blev gennemgået i afsnit 4.1.1. I det følgende beskrives de mest anvendte Web-baserede formelle sprog til videnrepræsentation.

XML

eXtensible Markup Language (XML) er et formelt opmærkningssprog, dvs. en notation til at opmærke data ved hjælp af bl.a. tags ”< >”. Sproget er blevet udviklet af Word Wide Web Consortium (W3C)[16] for at kunne udveksle data på internettet. XML er en undermængde af et andet notationssprog, Standard Generalized Markup Language (SGML) som blev udviklet i halvfjerdserne. XML består af en mængde syntaktiske regler for at strukturere dokumenter sådan at computere kan læse, udveksle og generere data, samt sikre at disse data er utvetydige. XML understøtter UNICODE, et system hvor et entydigt tal angiver hvert tegn, uafhængigt af programtype, sprog og system.

XML indeholder ingen restriktioner om betydningen af de XML-opmærkede dokumenter. Betydningen bestemmes af dem der skaber dokumenterne. Derfor er XML et metasprog hvormed man definerer andre sprog, bl.a. sprog til at beskrive ontologier. Hovedkomponenter i XML-dokumenter er såkaldte elementer (*elements*) som er hierarkisk organiserede. XML-dokumenter har et unikt rodelement. Elementer kan

[16] W3C er et konsortium af offentlige og private organisationer fra hele verden der har til formål at definere standarder og protokoller til at kunne genanvende data på internettet.

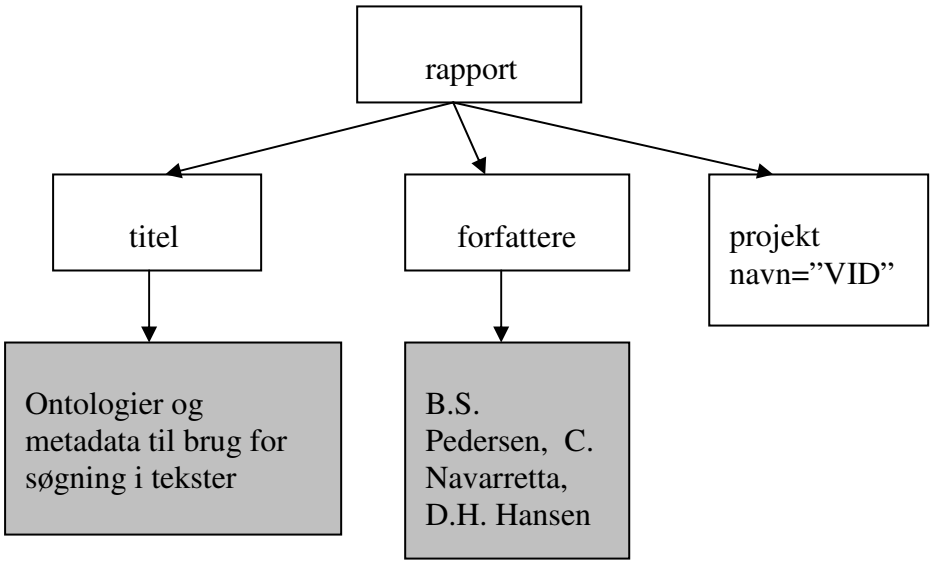
indeholde tekst, andre elementer eller være tomme og de kan være tilknyttet et udefineret antal attributter (*attributes*). XML-specifikationerne findes på adressen http://www.w3.org/TR/REC-xml.

Strukturen, indholdet og semantikken af XML-dokumenter kan specificeres i såkaldte skemaer. De mest anvendte skemaer er udtrykt i XML Schema som selv er et XML-sprog. I et skema kan man bl.a. definere hvilke elementer, attributter og attributværdier der er lovlige i et bestemt domæne og man kan fastsætte deres type. Derfor definerer skemaer et givet sprogs vokabular. XML-dokumenter der følger et eller flere skemaer siges at være gyldige (*valid*) ifølge disse skemaer. Specifikationerne for sproget XML Schema findes på adressen http://www.w3.org/XML/Schema.
Et eksempel på et simpelt XML-dokument kan ses i figur 4.1.

```
<?xml version="1.0" encoding=”ISO-8859-1”?>
<rapport >
     < titel >Ontologier og metadata til brug for søgning i tekster</
titel>
     < forfattere> B.S. Pedersen, C. Navarretta, D.H. Hansen</
forfattere>
     < projekt navn=”VID”/>
</rapport>
```

Figur 4.1: XML-dokument

Den første linie i XML-dokumentet angiver den versionen af XML og den type kodning som dokumentet anvender. Tekst indeholdt i tagsene ”< >”er metadata, mens tekst udenfor tagsene er selve data. Strukturen af dokumentet i 4.1 repræsenteres grafisk i figur 4.2, hvor de hvide bokser er metadata og de grå bokser er data.



Figur 4.2: Strukturen af XML-dokument i 4.1

I 4.2 ses den hierarkiske struktur af XML-dokumentet med rodelementet ”rapport” og de tre underelementer, ”titel”, ”forfattere” og ”projekt”. Elementet ”projekt” har et attribut ”navn” med værdi ”VID”. Data i dokumentet er titlen og forfatterne af denne rapport.

Nye standarder og protokoller der supplerer XML-specifikationer, bliver hele tiden udviklet under WC3. Nogle eksempler er *XML namespaces* som giver mulighed for at anvende opmærkninger fra forskellige XML-applikationer i samme XML-dokument, *XLink* som er en standard til at tilføje hyperlinks til et XML-dokument og *XSL* som er et sprog til at beskrive stylesheets for XML-dokumenter. *XML namespaces* er vigtige fordi ontologisprog, som bemærket tidligere, bruger mere generelle metadatasprog ved at genanvende elementer og attributter fra disse sprog via *XML namespaces*. XML namespaces er simple attributter der binder bestemte XML-elementer og –attributter til et domæne udenfor det aktuelle XML-dokument. Fx identificerer *rdf:class* og *owl:class* to forskellige elementer kaldet *class* som tilhører henholdsvis RDF og OWL, to sprog som beskrives senere. De to prefikser *rdf:* og *owl:* bindes domænerne for de to sprog med XML-namespace-attributter hvis værdier er henvisninger til internet-resurser, kendt som *Universal Ressourse Identifiers* (URI)[17]. Eksempler på disse attributter er i figur 4.3.

```
xmlns:rdf = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:owl = "http://www.w3.org/2002/7/owl#"
```

Figur 4.3: XML Namespaces

RDF

Resource Description Framework (RDF) er et standardsprog til at beskrive og udveksle metadata der henviser til forskellige typer resurser på internettet. RDF er blevet udviklet af W3C samtidig med XML og kan anvende XML-syntaks. RDF er udviklet under initiativet Semantic WEB og anvender følgende grundlæggende begreber:

- En *resurse* (*resource*): enhver ting som kan identificeres på internettet, derfor kan en resurse være en hjemmeside, et billede, et program, et element i et XML-dokument mm.
- En *egenskab* (*property*): en resurse som har et navn og kan anvendes som egenskab. En rapport, fx, kan have egenskaber som forfatter, editor, udgiver, antal sider, udgivelsesdato.
- En *sætning* (*statement*): en sammensætning af en resurse, en egenskab og en værdi, som også kan være en resurse. Et eksempel på en RDF-sætning er ”emnet for denne rapport er ontologi-baseret søgning”.

Når RDF udtrykkes i XML-syntaksen, er rodelementet et RDF-element ved navn *rdf:RDF*. Resurser beskrives med *rdf:Description*-elementet som består af en tredobbelt enhed (*triple*) der forbinder en resurse (subjektet) til en værdi (objektet) gennem en egenskab (prædikatet). *rdf:Description* er tilknyttet et attribut *about* som identificerer resursen. Værdien for *about*-attributtet er et URI eller en anden type henvisning til den

[17] URI inkluderer internet-adresser (URL) og standardangivelser af resurser (URN).

beskrevne resurse. Lister af resurser kan samles i grupper kaldet beholdere (*containers*). Underelementer af *rdf:Description* angiver egenskaberne.

Figur 4.4 er et eksempel på et simpelt RDF/XML-dokument taget fra E.R. Harold & W.S. Means (2001).

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">

  <rdf:Description about="urn:isbn:0596000588">
    <author>E. Rusty Harold</author>
    <author>W. Scott Means</author>
  </rdf:Description>

</rdf:RDF>
```

Figur 4.4: RDF/XML-dokument

I RDF-dokumentet i 4.4 er RDF-sætningen følgende: "forfatterne af bogen med ISBN 0596000588[18] er E. Rusty Harold og W. Scott Means". Resursen man siger noget om, er bogen, egenskaberne for bogen er de to forfattere. Værdien af de to forfattere er strengene med deres navne. Disse værdier kunne også være internet-adresser.

RDF indeholder ikke mekanismer til at foretage logisk inferens på de RDF-opmærkede data. Dele af RDF er dog blevet genanvendt i Web-baserede ontologisprog fordi RDF beskriver og identificerer data på internettet (resurser) med sætninger.

Som beskrevet i kapitel 3 kan Dublin Core Metadata også udtrykkes i RDF.
RDF-specifikationen findes på adressen http://www.w3.org/TR/1999/REC-rdf-syntax-19990222.

RDFS
RDF Schemas (RSFS) er RDF-typesystemer som beskriver RDFs vokabular. RDFS-typesystemer følger det objektorienterede videnmodelleringsparadigme som blev omhandlet i afsnit 4.1.1. Resurser som er beskrevet i RDF, bliver i RDFS organiseret i frames eller klasser (*classes*) som er hierarkisk struktureret og som er tilknyttet egenskaber (*properties*). Medlemmer af en klasse arver fælles karakteristika fra deres overklasser, mens egenskaberne er tilknyttet den enkelte klasse. En samling af RDF-klasser inden for et bestemt domæne definerer et RDF-skema som anvendes til at validere dokumenter der følger RDF-datamodellen. Derfor har RDFS samme funktion for RDF-dokumenter som XML skemaer har for XML-dokumenter. Fordi RDFS anvender det objektorienterede videnmodelleringsparadigme til at modellere internetdata beskrevet i RDF, kan sproget også bruges til at beskrive taksonomier og simple ontologier på internettet. RDFS, ligesom RDF, er blevet inkorporeret i Web-baserede ontologisprog.

RDFS-specifikationen findes på adressen http://www.w3.org/TR/2000/CR-rdf-schema-20000327.

[18] ISBN identificerer bogen *XML in a Nutshell 2. edition*.

DAML-OIL

DARPA Agent Markup Language (DAML) blev udviklet med støtte fra den amerikanske regering. DAML-sproget er blevet flettet sammen med et andet sprog udviklet i projektet Ontology Inference Layer (OIL). Det resulterende DAML+OIL-sprog er en videreudvikling af RDFS således at DAML+OIL kan udtrykke mere komplekse relationer mellem klasser og egenskaber, samt restriktioner om disse som er nødvendige for at kunne beskrive komplekse ontologier. For eksempel er det muligt i DAML+OIL at udtrykke at to klasser eller egenskaber er identiske, samt at nogle egenskaber kan have flere værdimængder, mens dette er ikke muligt i RDFS. Klassebegrebet fra RDFS er også videreudviklet og man kan udtrykke restriktioner om kombinationer af klasser. Formelt er elementet *daml:Class* en underklasse, dvs. en specialisering, af RDFS-elementet *rdfs:Class*. DAML+OIL bygger også på XML Schema. Mere specifikt er egenskabsværdier i DAML+OIL begrænset til de datatyper som er defineret i XML Schema eller til brugerdefinerede typer.

DAML+OILs semantik er veldefineret både modelteoretisk og aksiomatisk og det er muligt at anvende inferensnmekanismer på de data som er beskrevet i DAML+OIL. DAML+OIL-specifikationerne kan ses på http://www.w3.org/TR/daml+oil-reference.

Et uddrag fra en DAML+OIL-ontologi om sportsprodukter taget fra ontologien http://www.xml.com/2002/03/13/examples/SuperSports findes i figur 4.5.

```
<?xml version="1.0" encoding="UTF-8" ?>
- <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:daml="http://www.w3.org/2001/10/daml+oil#"
xmlns:dt="http://rdfinference.org/eg/supersports/dt"
xmlns:ss="http://rdfinference.org/eg/supersports/metadata/"
xmlns:xsd="http://www.w3.org/2000/10/XMLSchema#"
xml:base="http://rdfinference.org/eg/supersports/metadata/">
 <daml:Ontology rdf:about="">
  <daml:versionInfo>1.0</daml:versionInfo>
  <rdfs:comment>An ontology of Super Sports Inc. store
products</rdfs:comment>
  <daml:imports rdf:resource="http://www.daml.org/2001/03/daml+oil"
/>
  </daml:Ontology>

- <daml:Class rdf:ID="Product">
  <rdfs:label>Product</rdfs:label>
  <rdfs:comment>An item sold by Super Sports Inc.</rdfs:comment>
- <daml:disjointUnionOf parseType="daml:collection">
- <daml:Class rdf:ID="CurrentProduct">
  <rdfs:label>Current Product</rdfs:label>
  <rdfs:comment>An item currently sold by Super Sports Inc. at the
time of query</rdfs:comment>
  </daml:Class>…
</rdf:RDF>
```

Figur: 4.5 Uddrag fra et DAML+OIL-ontologi om sportprodukter

Som man kan se i uddraget, beskrives DAML+OIL ontologier ved at anvende XML-syntaks og elementer fra XML Schema, RDF, RDFS og DAML+OIL. I eksemplet defineres de relevante XML-namespaces-attributter i RDF-rodelementet (*rdf:RDF*). Selve ontologien over produkter fra firmaet ”Super Sports” defineres i elementet

daml:Ontology, som indeholder *rdfs:comment*-elementet. Klassen ”product” beskrives med DAML+OIL-elementet *daml:Class*. som også indeholder RDF-, RDFS-elementer og attributter.

OWL

Ontology Web Language (OWL) er blevet udviklet af W3C med DAML+OIL som model og er standardsproget til at beskrive Web-baserede ontologier. Som i DAML+OIL anvendes XML-syntaksen i OWL. OWL bruger elementer og attributter fra RDF, RDFS og XML Schema og giver mulighed for at udtrykke komplekse beskrivelser af klasser, egenskaber og relationerne mellem dem. Dette svarer til hvad man kan repræsentere i de mest avancerede videnrepræsentationssprog.

OWL findes i tre versioner af stigende kompleksitetsniveau: *OWL Lite*, *OWL DL* og OWL *Full*. OWL Lite er kernesproget hvorpå de andre to versioner er etableret.

- OWL Lite kan beskrive klassifikationshierarkier og kan udtrykke simple restriktioner på de beskrevne data. Det er egnet til at beskrive tesaurusser og taksonomier.
- OWL DL indeholder alle OWLs sprogskonstruktioner og har maksimal udtryksevne. Det er altid muligt automatisk at lave logiske slutninger på de data OWL DL beskriver, dvs. data i OWL DL er afgørlige. OWL DL er nært beslægtet med familien af formelle videnrepræsentationssprog kendt som Description Logics (se afsnit 4.1.1).
- OWL Full har samme udtryksevne som OWL DL, men tillader adskillige syntaktiske friheder i beskrivelsen af data. Inferensmekanismer kan ikke anvendes på alle dets dele. Data i OWL Full er derfor ikke altid afgørlige.

OWL DL ligner DAML-OIL, men inkorporerer de nyeste specifikationer af RDF, RDFS og XML Schema. Der er enkelte semantiske forskelle mellem OWL DL og DAML+OIL og OWL DL er mere stringent. Der findes et antal værktøjer til at konvertere DAML+OIL-ontologier til OWL (en liste over disse værktøjer findes på adressen http://www.daml.org/tools/).

OWL-specifikationen er beskrevet på http://www.w3.org/TR/owl-ref/.

Figur 4.6 indeholder et lille uddrag af en ontologi om vin taget fra http://www.w3.org/TR/2002/WD-owl-guide-20021104/wine.owl.

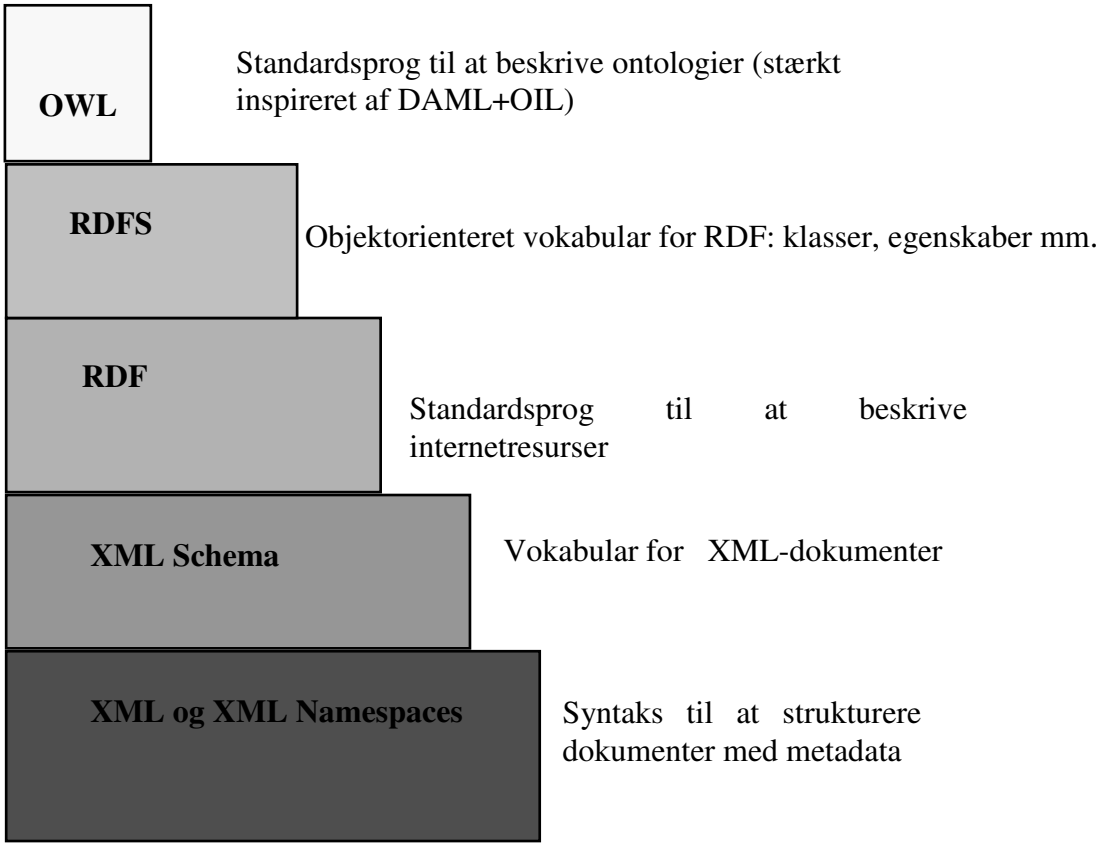
```
<?xml version="1.0"?>

<rdf:RDF
  xmlns     = "http://www.example.org/wine#"
  xmlns:vin = "http://www.example.org/wine#"
  xmlns:food= "http://www.example.org/food#"
  xmlns:dte = "http://www.example.org/wine-dt"
  xmlns:owl = "http://www.w3.org/2002/7/owl#"
  xmlns:rdf = "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs= "http://www.w3.org/2000/01/rdf-schema#"
  xmlns:xsd = "http://www.w3.org/2000/10/XMLSchema#">

  <owl:Ontology rdf:about="http://www.example.org/wine.owl">
    <rdfs:comment>
```

```
      Derived from the DAML Wine ontology at

    http://ontolingua.stanford.edu/doc/chimaera/ontologies/wines.dam
    l
    Substantially changed, in particular the Region based relations.
     </rdfs:comment>
  </owl:Ontology>

  <owl:Class rdf:ID="Wine">
    <rdfs:subClassOf rdf:resource="&food;PotableLiquid" />
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#hasMaker" />
       <owl:cardinality>1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>……
</owl:Class>……
<DessertWine rdf:ID="WhitehallLanePrimavera">
    <locatedIn rdf:resource="#NapaRegion" />
    <hasSugar  rdf:resource="#Sweet" />
    <hasFlavor rdf:resource="#Delicate" />
    <hasBody   rdf:resource="#Light" />
  </DessertWine> …
</rdf:RDF>
```

Figur 4.6: Uddrag af OWL-ontologi om vin

I ontologiuddraget i 4.6 angives de relevante XML-Namespaces i rodelementet, *rdf:RDF*. Dernæst introduceres selve ontologien i elementet *owl:Ontologi*. Kommentarer skrives i underelementer kaldet *rdfs:comment*. Vinbegrebet beskrives i et *owl:Class*-element og defineres som underklasse af ”drikbar væske” (*PotableLiquid*) som igen er en underklasse af begrebet ”mad” (*food*). Vin er bundet til egenskaben ”menneskeskabt” (*hasMaker*). Endelig introduceres en bestemt type dessertvin som beskrives med dens produktionsområde (*NapaRegion*) og fysiske egenskaber.

Forholdet mellem XML, XML Schema, RDF, RDFS og OWL illustreres i figur 4.7.



Figur 4.7: OWL og de underliggende XML-baserede standardsprog

Topic Maps

Topic Maps er blevet udviklet uafhængigt af W3C under initiativet Semantic Web. Formålet med Topic Maps er at modellere viden på internettet. Selv om Topic Maps og RDF er udviklet med samme formål, er de to formelle sprog ret forskellige og anvendes af forskellige brugergrupper. For nyligt er der blevet dannet en arbejdsgruppe som skal undersøge hvorvidt det er muligt at konvertere XTM til RDF (bl.a. Garshol, Living with Topic maps and RDF). I Topic Maps har man samlet ideer fra indekser, glossarier, tesaurusser samt semantiske net, herunder især Conceptual Graphs (Sowa 1984). Topic Maps er mere udtryksrig end RDF, men sproget er ikke så stringent som OWL DL.

Topic Maps er en ISO-standard (ISO/IEC 13250). I de nyere versioner er sproget udtrykt med XML-syntaksen og kaldes XTM (XML Topic Maps). Specifikationer for XTM findes på adressen http://www.topicmaps.org/xtm/.

Viden om et område/emne fra forskellige Web-sider og resurser kan beskrives i et *topic map*.

Hovedkomponenterne i et topic map er følgende:

- *Topics*: inkluderer emner, objekter, begreber mm.
- *Associations*: er indbyrdes relationer mellem topics. Associations er også topics.
- *Occurrences*: er forekomster af topics i relevante resurser.
- *Subject Identity, Facets, Scope:* anvendes til at forbinde samme topic med forskellige navne, samt at angive restriktioner på forekomsterne af et topic i bestemte domæner.

Rodelementet i et XTM-dokument er et topic-map. I figur 4.8 gives et uddrag af et topic map om opera taget fra http://www.ontopia.net.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<topicMap  id="operatm-tm"
                    xmlns="http://www.topicmaps.org/xtm/1.0/"
                    xmlns:xlink="http://www.w3.org/1999/xlink" >
<topic id="verdi">
    <instanceOf><topicRef xlink:href="opera-
template.xtmp#composer"/></instanceOf>
   <baseName>
     <baseNameString>Verdi, Giuseppe</baseNameString>
   </baseName>
   <baseName>
     <scope><topicRef xlink:href="opera-
template.xtmp#normal"/></scope>
     <baseNameString>Giuseppe Verdi</baseNameString>
   </baseName>
   <baseName>
     <scope><topicRef xlink:href="opera-
template.xtmp#shortname"/></scope>
     <baseNameString>Verdi</baseNameString>
   </baseName>
 .....
 </topic>
…..
<topic id="nabucco">
<instanceOf><topicRef
xlink:href="operatemplate.xtmp#opera"/></instanceOf>
<subjectIdentity>
<subjectIndicatorRef
xlink:href="http://opera.stanford.edu/opera/Verdi/Nabucco"/>
</subjectIdentity>
<!-- premiere: la-scala (1842 (9 Mar)) -->
<!-- written-by: solera -->
<!-- based-on: nabuchodonosor nabucodonosor -->
<!-- published-by: ricordi-p -->
<!-- character-in-opera: nabucco-c ismaele zaccaria abigaille fenena
high-priest-of-baal abdallo anna3 -->
<!-- aria-in-opera: -->
<!-- setting: jerusalem babylon -->
<baseName>
<baseNameString>Nabucco</baseNameString>
</baseName>.......
</topic>
<association>
<instanceOf><topicRef xlink:href="opera-template.xtmp#composed-
by"/></instanceOf>
<scope><topicRef xlink:href="opera-template.xtmp#music"/></scope>
<member>
<roleSpec><topicRef xlink:href="opera-
template.xtmp#composer"/></roleSpec>
<topicRef xlink:href="#verdi"/>
</member>
<member>
<roleSpec><topicRef xlink:href="opera-
template.xtmp#opera"/></roleSpec>
<topicRef xlink:href="#nabucco"/>
</member>
</association>…
</topicMap>
```

Figur 4.8: Uddrag fra Topic Maps om Opera

I uddraget defineres topic ”Verdi, Giuseppe”, som er en komponist. I det aktuelle domæne (topic map) refererer ”Verdi, Giuseppe”, ”Giuseppe Verdi” og “Verdi” til samme topic. Topic ”Nabucco” af type “opera” relateres til topic ”Verdi, Giuseppe” med relationen (Association) komponeret-af *(composed-by*).

4.1.3 Fælles protokoller og formater til udveksling af eksisterende videnbaser

Samtidig med udviklingen af formelle standardsprog til modellering af viden på internettet, har eksperter inden for videnrepræsentationsområdet arbejdet med at definere fælles formater og protokoller til udveskling af eksisterende ontologier implementeret i videnbaserede systemer.

KIF (Knowledge Interchange Format)
http://logic.stanford.edu/kif/kif.html
KIF er blevet udviklet under Knowledge Sharing Effort, et konsortium der har til formål at finde metoder til at genbruge eksisterende videnbaser implementeret i forskellige sprog. KIF er et formelt sprog til at beskrive og dele viden mellem programmer. KIF har en deklarativ semantik og anvender første-ordenslogik. KIF er ikke blevet udviklet som et nyt implementeringsprog, men som et slags mellemsprog som eksisterende formelle sprog skal kunne konverteres til.

OKBC (Open Knowledge Base Connectivity)
http://www.ai.sri.com/~okbc/
OKBC er blevet udviklet af Stanford Research Institute (SRI), California, og er en protokol til at kunne få fat i oplysninger gemt i videnbaser implementeret i forskellige formelle sprog. OKBC-protokollen dækker frame-baserede sprog og dens underlæggende videnrepræsentationsmodel tillader at beskrive klasser, individer, *slots* and *facets*. OKBC er blevet en de-facto standard.

ONTOLINGUA
Stanford Universitet, California, har udviklet ONTOLINGUA som er logisk kompatibelt med KIF og som anvender en frame-baseret videnrepræsentation. ONTOLINGUA anvendes i ONTOLINGUA-serveren som er blevet opbygget til at skabe, browse og editere ontologier, samt uddrage de data som er beskrevet i disse ontologier (se afsnit 4.2).

OCML (Operational Conceptual Modeling Language)
http://kmi.open.ac.uk/projects/ocml/
OCML er et objektorienteret sprog til at udtrykke relationer, funktioner, regler, klasser og instanser. Det er blevet udviklet af Knowledge Media Institute (KMI) Open University, Storbritannien, og er kompatibelt med Ontolingua. OCML indeholder mekanismer til at foretage logiske slutninger på de modellerede data.

XOL

XOL: An XML-Based Ontology Exchange Language

XML-based Ontology exchange Language (XOL) er udviklet af Stanford Research Institute (SRI), California, til at udveksle ontologier om bioinformatik, men kan anvendes til andre domæner. Sproget anvender XML-syntaks og implementerer en undermængde af et fælles videnrepræsentationsprotokol kaldet OKCB.

4.2 Værktøjer

I dette afsnit beskriver vi nogle værktøjer til at opbygge, editere og navigere ontologier i de sprog eller med de protokoller som blev omtalt i afsnit 4.1.1 og 4.1.2. Vores udgangspunkt har været Gómez-Pérez et al. (ed.) (2002)'s rapport, som indeholder en oversigt over ontologirelaterede værktøjer, samt en evaluering af disse. Gómez-Pérez et al. deler værktøjerne i følgende grupper: værktøjer til at opbygge ontologier, værktøjer til at sammenflette ontologier, værktøjer til at validere ontologier, ontologi-baserede annotationsværktøjer og værktøjer til at uddrage viden fra ontologier gemt i databaser (query systems). I denne rapport beskrives kun værktøjer under de første tre grupper med fokus på de værktøjer som er frit tilgængelige, dvs. som enten er OpenSource eller er frit anvendelige på internetservere.

4.2.1 Værktøjer til at opbygge og editere ontologier

Der findes mange værktøjer til at opbygge og editere ontologier. En meget overordnet oversigt over et stort antal værktøjer findes på
http://xml.com/2002/11/06/Ontology_Editor_Survey.html.
I det følgende beskriver vi kort nogle af de mest anvendte af disse editorer.

Apollo
http://apollo.open.acuk/

Apollo er et system til at opbygge ontologier og er blevet udviklet af Knowledge Media Institute (KMI) på Open University, Storbritannien. Apollo-videnbaser kan oversættes til andre formater via små hjælpeprogrammer kaldet *plugins*. Apollo-modellen er frame-baseret og kompatibel med OKCB-protokollen. Apollo kan frit downloades på internettet. Det er udviklet i Java og kan køre på både UNIX og Windows. Systemet har ingen inferenssupport.

Protégé-2000
http://protege.stanford.edu/
Protégé-2000 er et værktøj til at modellere og editere videnbaser. Det er blevet udviklet af Stanford University og bliver anvendt af brugere over hele verden. Protégé-2000 kan frit hentes på internettet under Mozillas Open-Source-licens. Systemet er programmeret i Java og kører under adskillige operativsystemer (bl.a. Windows, Sun, Linux). Værktøjet bliver hele tiden videreudviklet. Videnmodellen bag ved Protégé-2000-videnbaser er kompatibel med OKBC-protokollen, den bygger på frames og første-ordenslogik.

Bibliotek i Protégé-2000 indeholder adskillige plugins som er blevet udviklet af systemets brugere og som tilføjer forskelligartede funktionaliteter til det oprindelige system. Fx er der plugins til at konvertere Protégé-2000-ontologier fra/til andre

formalismer såsom XML + XML Schemas, RDF, Topic Maps og DAML+OIL (OWL-plugins er under udvikling). Dog virker ikke alle plugins med samme version af værktøjet. Der findes diverse plugins som tilføjer inferensmekanismer til Protégé-2000 eller importerer i Protégé-2000 dele af eksisterende on-line baser såsom WordNet.

WebODE
http://webode.dia.fi.upm.es/
WebODE er et system der hjælper brugeren til at opbygge og sammenflette ontologier. WebODE er udviklet af Technical School of Computer Science i Madrid (UPM). WebODE findes i en serverversion på internettet eller som system som kan fås via en licensaftale. I WebODE gemmes ontologier i en relationel database og man kan importere og gemme ontologier fra/i forskellige formater såsom Flogic, RDFS og DAML+OIL. WebODEs inferensmekanisme er implementeret i Prolog. Videmodellen bagved de opbyggede databaser er baseret på frames og første-ordenslogik.

OntoEdit
http://www.ontoprise.de
OntoEdit er et system til at udvikle ontologier gennem en grafisk grænseflade. OntoEdit er udviklet af Ontoprise® og er frit tilgængelig i en minimal version. OntoEdit findes også i en professionel version som kan købes. Denne version indeholder ekstra faciliteter (plugins) såsom inferensmekanismer, konsistenskontrol mm. Man kan eksportere og gemme ontologier fra/i XML, RDFS og DAML+OIL. Videnmodellen som anvendes i OntoEdit bygger på frames og første-ordenslogik.

Evaluering af værktøjer til at opbygge og editere ontologier
Gómez-Pérez et al. (2002) evaluerer værktøjer til at opbygge ontologier ud fra følgende egenskaber:

- tilgængelighed: hvordan kan man få adgang til værktøjet?
- software-arkitekturen: kan værktøjet udvides og i hvilke formater kan man gemme ontologierne?
- interoperationalitet: kan værktøjet spille sammen med andre ontologi-relaterede værktøjer eller systemer?
- videnrepræsentationsparadigme samt metoder: hvilke videnrepræsentationsformalismer og opbygningsmetoder understøtter værktøjet?
- inferenssupport: indeholder værktøjet inferensmekanismer?
- anvendelighed: er værktøjet nemt at anvende og indeholder det faciliteter som grafiske grænseflader og ontologibiblioteker der kan understøtte opbygning og anvendelse af ontologier?

Ud fra disse kriterier konkluderer Gómez-Pérez et al. at der findes et stort antal værktøjer til at opbygge og editere ontologier og at disse værktøjer har mange fælles træk. Alligevel er der ingen af værktøjerne der fuldstændig kan integreres i andre ontologi-relaterede systemer. Gómez-Pérez et al. bemærker også at der ikke findes værktøjer der støtter brugeren i alle faserne af ontologiarbejdet.

Vi har installeret og afprøvet Apollo og Protégé-2000, samt afprøvet WebODE-serveren. Systemerne er nemme at bruge, men konvertering til og fra andre formater

virkede ikke helt efter hensigten. Fordelen med Protégé-2000 er at man kan få hjælp til at anvende systemet via en mailing-liste, samt at systemet er veldokumenteret og løbende bliver videreudviklet. Endeligt findes der mange ontologier i Protégé-2000-formatet.

4.2.2 Værktøjer til at sammenflette ontologier

ONTOLINGUA-serveren
http//ontolingua.stanford.edu/
Ontolingua-serveren er en Web-server som støtter opbygningen af genanvendelige ontologier. Den er udviklet af Knowledge Systems Laboratory (KSL) på Stanford University, California. Serveren giver adgang til et bibliotek af ontologier, til oversættere til andre implementationssprog såsom Prolog, CLIPS og Loom samt en editor til at skabe, editere og browse ontologier. Fjerntliggende editorer kan også arbejde med ontologierne i ONTOLINGUA-serveren. Både lokale og fjerntliggende applikationer har adgang til ontologierne i ONTOLINGUA-biblioteket gennem OKCB-protokollen.

Chimaera
http://www.ksl.stanford.edu/software/chimaera/
Chimaera er et system som også er udviklet af KSL, Stanford University. Chimera støtter brugere der vil sammenflette ontologier på internettet og indeholder faciliteter til at teste det opnåede resultat. Chimera understøtter ontologier i adskillige formater som er kompatible med OKBC-protokollen.

PROMPT
http://protege.stanford.edu/plugins/prompt/prompt.html
PROMPT er et værktøj til semi-automatisk fletning af ontologier (Noy & Musen, 2000). PROMPT er et plugin til Protégé-2000 og støtter brugeren i fletningsprocessen.
PROMPT identificerer steder hvor forskellige ontologier kan integreres, foreslår hvad brugeren kan gøre i hver fase af processen, identificerer problemer med integrationen af komponenter mm. PROMPT undersøger bl.a. klassenavne, klassehierarkier, attributter og attributnavne.

ODEMerge
http://webode.dia.fi.upm.es/
ODEMerge er et værktøj til at flette ontologier som er integreret i WebODE. Det er derfor et klient-serverværktøj.
ODEMerge støtter sammenfletning af ontologier efter følgende metode: 1) konverterer de to ontologier til samme format, 2) validerer dem 3) fletter dem 4) validerer det opnået resultat 5) konverterer den resulterende ontologi til det ønskede format.

Evaluering af værktøjer til at sammenflette ontologier
Gómez-Pérez et al. (2002) kigger på hvorvidt processen til at samle og flette ontologier kan automatiseres og hvor mange ontologikomponenter der kan samles. De konkluderer at værktøjerne anvender enten en ”bottom-up” eller en ”top-down” strategi til at samle

ontologier. I bottom-up-strategien starter sammenfletningen fra instanserne i ontologierne, mens der i top-down-strategien sammenflettes med udgangspunkt i de mest generelle begreber i ontologierne. I ingen af de evaluerede værktøjer sker sammenfletningen fuldt automatisk og sammenfletningen dækker ikke aksiomer og regler.

4.2.3 Evalueringsværktøjer

Gómez-Pérez et al. (2002) foreslår at systemer til evaluering af ontologi-relaterede værktøjer bør gennemgå følgende aspekter:

1. syntaksen og semantikken af de skabte ontologier i forhold til de standarder som værktøjerne understøtter.
2. værktøjernes tekniske egenskaber, såsom deres evne til at spille sammen med andre teknologier, deres hastighed, deres grænseflade mm.

Alle de eksisterende evalueringsværktøjer er delkomponenter af andre værktøjer. Gómez-Pérez et al. (2002) analyserer følgende værktøjer: **OntoAnalyser** og **OntoGenerator** som er plugins for OntoEdit og er implementeret af Ontoprise og Universitetet af Karlsruhe (AIFB);**One-T** som er udviklet af ontologigruppen i det Tekniske Universitet i Madrid (UPM) som en komponent af Ontolingua-serveren; **OntoClean** som er udviklet af ontologigruppen i CNR-Padova samt UPM-Madrid og som en plugin for WebODE.

OntoGenerator checker hvor hurtigt ontologi-baserede værktøjer fungerer, hvor store ontologier de kan håndtere og hvor stor hukommelse værktøjerne anvender til at håndtere ontologierne.

OntoAnalysis, One-t og OntoClean checker om de ontologier som opbygges i værktøjerne følger syntaksen for det formelle sprog de anvender, samt om disse ontologierne er konsistente semantisk.

Ud fra denne analyse konkluderer Gómez-Pérez et al. (2002) at ingen af disse evalueringsværktøjer er komplette.

4.3 Konkluderende bemærkninger

Der findes forskellige typer formelle sprog til at beskrive ontologier. De sprog som er mest anvendte i dag er sprog der tilhører sprogfamilien beskrivelseslogik (DL). Disse sprog følger den objektorienterede videnrepræsentationsmodel, også kendt som den frame-baserede model, og anvender første-ordenslogik. For nyligt har man defineret standardprotokoller og formater som eksisterende ontologier der er implementeret i forskellige formalismer, bør kunne konverteres til. De meste anvendte af disse formalismer og protokoller er den logiske formalisme KIF og protokollen OKBC, som understøtter objektorienterede videnrepræsentationsmodeller. Hvis man ønsker at ontologier skal kunne anvendes i forskellige systemer og applikationer, er det vigtigt at man implementerer dem i formelle sprog som kan konverteres til KIF og som er kompatible med OKBC-protokollen,.

OWL er standarden for at beskrive ontologier der beskriver internetdata. OWL er beslægtet med DL-sprogene, kan konverteres til KIF og følger en frame-baseret videnrepræsentationsmodel som er kompatibel med OKBC-protokollen. OWL er en videreudvikling af sproget DAML+OIL, anvender XML-syntaksen og er en specialisering af RDF og RDFS. OWL findes i tre versioner, med forskellig udtryksrigdom. Et andet sprog til at beskrive ontologier over internet-data som er ved at få fodfæste i visse applikationsområder, er Topic Maps. Topic Maps adskiller sig i mange henseender fra OWL, men anvender XML-syntaksen og er tæt på traditionen fra de semantiske net.

Der findes mange værktøjer til at opbygge, editere og navigere ontologier. Disse værktøjer er blevet implementeret af både universiteter og private virksomheder og opbygger ontologier i værktøjspecifikke videnrepræsentationssprog. Ontologierne kan dog ofte konverteres til KIF og er kompatible med OKBC-modellen. Nogle af værktøjerne indeholder desuden konverteringsmekanismer til og fra andre standardsprog. Vi har kun beskrevet nogle af de mest kendte og frit tilgængelige ontologirelaterede værktøjer og har afprøvet dem. Endeligt har vi gennemgået de vigtigste evalueringskriterier som er blevet anvendt til at evaluere ontologirelaterede værktøjer i EU-projektet OntoWEB (Gómez-Pérez et al., 2000).

I VID-projektet vil vi arbejde med Protégé-2000 fordi det er det mest anvendte værktøj til at opbygge og editere ontologier som løbende bliver vedligeholdt og ajourført. Desuden indeholder Protégé-2000 inferensmekanismer og det er muligt at konvertere Protégé-2000-ontologier til/fra andre formelle standardsprog.

Referencer

Berners-Lee, T., J. Hendler & O. Lassila. 2001. The Semantic Web. http://scientificamerican.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21&catID=2.

Corcho, O. & A. Gómez-Pérez, 2000. A Roadmap to Ontology Specification Languages. ELAW00, Proceedings of the 12th International Conference on Knowledge Engineering and Knowledge Management.

Crofts, N., I. Dionissiadou, M. Doerr, M. Stiff (2001). Definition of the CIDOC object-oriented Conceptual Reference Model. Technical Report, ICS-FORTH, Greece.

Gómez-Pérez, A. et al. eds, 2002. *OntoWeb: A survey on ontology tools*, Deliverable 1.3. IST-2000-29243. http://babage.dia.fi.upm.es/ontoweb/wp1/OntoRoadMap/index.html

Fellbaum, C. 2000. WordNet - An Electronic Lexical Database, MIT Press.

Guarino, N. & C. Welty 2002. Evaluating Ontological Decisions with OntoClean. *Communications of the ACM,* 45(2):61-65.

Harold, E.R.& W.S. Means, 2002, *XML in a Nutshell*, second edition, O'Reilly.

Hendler, J. Agents and the Semantic Web. University of Maryland. http://www.cs.umd.edu/~hendler/AgentWeb.html.

MacGregor, R., 1999. Retrospective on Loom, http://www.isi.edu/isd/LOOM/papers/macgregor/Loom_R

Nilsson, Jørgen Fischer. 2001. A Logico-Algebraic Framework for Ontologies. OntoLog, in: Per Anker Jensen & Peter Skadhauge *Ontology-based Interpretations of Noun Phrases, Proceedings from the First International OntoQuery Workshop,*11-43. University of Southern Denmark, Kolding.

Noy, N.F & M.A. Musen, 2000. PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment. In: *Seventh National Conference on Artificial Intelligence (AAAI-2000)*. Austin, TX, 2000.

Onyshkevych, B. & S. Nirenburg. 1995. A Lexicon for Knowledge-based Machine Translation. In: *Machine Translation 10:1-2.* Kluwer Academic Publishers.

Pepper, S. The TAO of Topic Maps. STEP infotek, Oslo. http://www.ontopia.net/topicmaps/learn_more.htm.

Guarino, N. & C. Welty (2002) Evaluating Ontological Decisions with OntoClean Communications of the ACM, 45(2): 61-65.

Lenci, Alessandro, Federica Busa, Nilda Ruimy, Elisabetta Gola, Monica Monachini, Nicoletta Calzolari, Antonio Zampolli, James Pustejovsky, Emilie Guimier, Lee Humphreys, Ursula Von Rekovsky, Antoine Ogonowsky, Clair

McCauley, Wim Peters, Ivonne Peters, Rob Gaizauskas, Marta Villegas & Ole Norling-Christensen. 2000. *SIMPLE Linguistic Specifications*. Unpublished SIMPLE report, University of Pisa.

Lyons, J. 1977. Semantics. Cambridge University Press, London.

Masolo, C. S. Borgo, A. Gangemi, N. Guarino, A. Oltramari, L. Schneider. 2003. *WonderWeb Deliverable D17, Preliminary Report,* Padova, Italy.

Sevcenko, M. 2003. Online Presentation of an Upper Ontology. In *Proceedings of Znalosti 2003*, Ostrava, Czech Republic, February 19-21, 2003.

Sowa, J., 1984. Conceptual Structures - Information Processing in Mind and Machine, MA: Addison-Wesley Pub.

Sowa, John F. 2000. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*, Brooks Cole Publishing Co., Pacific Grove, CA.

Vossen, Piek (ed.). 1999. *EuroWordNet, A Multilingual Database with Lexical Semantic Networks*, Kluwer Academic Publishers, The Netherlands.

Relevante links om metadata

Om metadata generelt:

God introduktion til metadata:
http://www.getty.edu/research/institute/standards/intrometadata/

Meget omfattende liste over metadataresurser:
http://www.ifla.org/II/metadata.htm

Overblik over metadatastandarder (1997):
http://www.ukoln.ac.uk/metadata/desire/overview/

Overblik over metadata-initiativer (*Schemas*):
http://www.schemas-forum.org/

Om at komme fra et metadatasystem til et andet:
Mapping between metadata formats
http://www.ukoln.ac.uk/metadata/interoperability

Gode råd om netsteder (vejledning om metadata fra IT- og Telestyrelsen):
http://netsteder.dk/raad/metadata/index.html

Dublin Core:

Dublin Core Metadata Initiative:
http://dublincore.org

The Dublin Core matadata element set, ISO 15836:

http://www.niso.org/international/SC4/n515.pdf

World Wide Web Consortiet:
http://www.w3.org

The Warwick Framework:
http://www.dlib.org/dlib/july96/07weibel.html

Offentlig Information Online, OIO:
http://www.oio.dk

Værktøjer til generering af metadata:

Oversigt over værktøjer til generering af Dublin Core Metadata:
http://dublincore.org/tools/

DC-dot, et engelsk værktøj til fuldautomatisk generering af DC metadata:
http://www.ukoln.ac.uk/metadata/dcdot/

Dansk BiblioteksCenter /Damnarks National Bibliografs metadatablanket:
http://purl.dk/metadata/meta_lang.htm

Dansk værktøj til generering af metadata:
www.metamaten.dk

Metadata og søgning:

"*Metadata and the World Wide Web*"
www.getty.edu/research/institute/standards/intrometadata/2_articles/gill/content.html

"*A metadata Registry for the Semantic Web*"
www.dlib.org/dlib/may02/wagner/05/wagner.html