Marit Brodshaug

Emnekart basert på standarder

Trondheim, Februar 2006

Norges teknisk-naturvitenskapelige universitet
Fakultet for Informasjonsteknologi, matematikk og elektroteknikk
Institutt for datateknikk og informasjonsvitenskap

Masteroppgave
Studieprogram: Master i informatikk

Hovedveileder: Trond Aalberg, IDI

NTNU
Det skapende universitet

Sammendrag

Emnekart er et hjelpemiddel for å navigere blant ressurser, men har i utgangspunktet ingen bestemt struktur for hvordan informasjon bør struktureres. Oppgaven tar derfor for seg ulike formater, som er benyttet for strukturering av informasjon, for å se om disse kan være til hjelp ved implementasjon av emnekart. Det er også slik at mange ressurser allerede har metadata knyttet til seg, eller er strukturert i standardiserte modeller, eller klassifisert etter gitte systemer, og det kan da være av interesse å fortsette og nyttiggjøre seg av denne informasjonen og struktureringen selv om man tar i bruk en standard som emnekart. For å undersøke dette er Dublin Core, Dewey og FRBR-modellen implementert i emnekart standarden.

Det er også i oppgaven sett på sammenfletting av forskjellige emnekart, og om en standardisering av emnekart kan gjøre sammenflettingen bedre. Det er testet både sammenfletting med to emnekart basert på samme struktur, i tillegg til sammenfletting på tvers av strukturer.

Forord

Jeg vil rette en stor takk til min veileder Trond Aalberg som alltid har vært positiv og imøtekommende. Han har vist interesse for arbeidet jeg har gjort, og hele tiden kommet med inspirerende idéer og gode råd for veien videre i prosessen.

Tusen takk også til mine gode venner og min familie som har hjulpet meg etter beste evne. De har oppmuntret, trøstet eller gitt meg dytt, ettersom hva som har vært mest nødvendig.

Tusen takk!

Trondheim 01.12.2005
Marit Brodshaug

Innhold

Figurer

Kapittel 1

Innledning

1.1 Bakgrunn

I dagens informasjonssamfunn er det lett å finne informasjon, og Internett er et hjelpemiddel for dette. Internett er en samlingsplass hvor alle kan legge ut og hente ned den informasjonen de ønsker. Et problem er at det i stor grad mangler strukturer på Internett. Det er ofte vanskelig å finne akkurat den informasjonen man er ute etter fordi den drukner i mengder av urelevant informasjon. Det er utviklet websteder og portaler som er med på å avgrense informasjon innenfor spesielle fagfelt osv. Men disse portalene trenger også en struktur for hvordan de skal være bygget opp for at ressursene skal være lette å finne.

Bibliotekene er en annen samlingsplass for informasjon. Men i motsetning til Internett har bibliotekene faste strukturer og klassifikasjonsmønstre for dataene de har lagret slik at det skal være lett å finne frem til den informasjonen man er ute etter.

Man ser nå et behov for å ta lærdom fra bibliotekfaget og lage strukturer på internett også, slik at gjenfinningen av relevant informasjon blir lettere. Ulike teknologier for Internett er derfor under utvikling, og en av teknikkene som har bredt vidt om seg er emnekart (Topic Maps). Denne teknologien fungerer som et kart for internettressurser og hjelper brukeren til å navigere seg frem til riktig ressurs ved hjelp av linker. Emnekartstandarden ser for seg at mennesket er ute etter å vite mer om et spesifikt emne, og all informasjon er derfor strukturert og satt sammen på bakgrunn av emner og hvordan emnene assosierer til hverandre.

Emnekartstandarden har som mål å samle all kunnskap på én plass, slik at man ikke skal behøve å lete mange plasser etter emnet man ønsker mer informasjon om. Men slik det fungerer i dag er det mange separate nettsteder som er basert på hvert sitt emnekart. Eksempler på nettsteder basert på emnekart er «forbrukerportalen.no» og «forskning.no». Begge disse emnekartene har et emne om helse, og man må dermed gå inn på begge emnekartene for å få vite det disse har av ressurser innen dette emnet. Etter emnekartets mål, som går ut på at man skal nå all informasjon fra en plass, burde disse emnekartene derfor ha vært slått sammen.

Det er flere grunner til at sammenslåinger av emnekart ikke blir utført. En av grunnene er at emnekartene blir laget av forskjellige instanser som ikke har noe med hverandre å gjøre, og som ikke ønsker å samarbeide og utveksle ressurser. En annen grunn er at det kan være vanskelig å bevare strukturen og brukervennligheten ved sammenslåing av emnekart som er helt forskjellig oppbygget. Slik det fungerer i dag, er det opp til utviklerne av emnekartene å lage ontologier for hvilke emner de skal ha med, og hvilke assosiasjoner de mener det er naturlig å ha mellom disse emnene. Sammenslåing skjer ved hjelp av like identifikatorer eller like navn på emner.

Det er derfor avhengig av at de forskjellige emnekartene benytter samme identifikatorer/navn på emner som omhandler det samme. Men selv om dette er tatt høyde for, kan likevel strukturen bli dårligere ved sammenslåing hvis den er veldig forskjellig i de ulike emnekartene.

1.2 Problemstilling

Emnekart er en populær plattform for organisering og deling av informasjon, og det er en aktuell problemstilling å se på alternative måter å implementere emnekart på.

Emnekartstandarden har i seg selv ikke noen fast struktur, eller noen modell for hvordan ressursene bør representeres. Det er aktuelt å se på hvordan ulike strukturer, formater og modeller kan benyttes i forbindelse med emnekart. På denne måten kan man tenke seg at bibliografiske ressurser som allerede er strukturert i et gitt format eller modell kan bestå i denne strukturen selv om de blir digitalisert og implementert i emnekart. De bibliografiske strukturene er nøye gjennomarbeidet og det er lagt mye arbeid i å få ressurser inn i disse strukturene og det kan derfor være hensiktsmessig å nyttiggjøre seg av dette i en standard som emnekart.

Metadataformater er formater som har vært flittig brukt i biblioteksammenheng, og er data som forteller noe om ressurser og gir viktig informasjon om dem. Dette er informasjon som man ikke ønsker skal bli borte ved konvertering over til emnekart, og det er aktuelt å se hvordan metadataformater kan bli implementert i emnekart.

En annen problemstilling er å se om det ved bruk av slike innarbeidede standarder i emnekart vil bli lettere å beholde strukturen ved sammenslåing av flere emnekart. Kan det være hensiktsmessig å standardisere emnekartet i større grad?

Problemstillingen som vil bli tatt opp i oppgaven er:

- Er det hensiktsmessig å bruke etablerte formater som metadatastandarder, klassifikasjonssystemer og konseptuelle referansemodeller i forbindelse med emnekart?
- Hva har standardisering av emnekart å si for sammenfletting av flere emnekart?

1.3 Fremgangsmåte

Oppgavens fokus har endret seg underveis ettersom det dukket opp stadig nye og interessante aspekter som var relevante. Jeg startet med å se på emnekart i forbindelse med FRBR-modellen. Grunnen til dette var at jeg skulle undersøke hvordan jeg kunne strukturere musikkinformasjon og gjøre det lett å navigere i informasjonen. Jeg skulle se hvordan det fungerte å bruke en modell som i utgangspunktet er laget for bokinformasjon til musikkdata, og implementerte musikkdata i FRBR-modellen. Dette er en modell som har en klar oppbygging, og strukturerer dataene på en bra måte, men kan være vanskelig å manøvrere i. Jeg valgte derfor å teste hvordan en slik modell kunne kombineres med emnekartstandarden som er godt egnet til navigering.

Det ble etterhvert interessant å se på hvordan andre formater kunne benyttes i samme hensikt, og oppgaven dreide seg mer over til metadataformater og sammenhengen til digitale bibliotek, og hvordan emnekartstandarden taklet ulike utfordringer i forhold til de ulike standardene.

Jeg implementerte derfor også Dublin Core og Dewey inn i emnekartstandarden for å se om dette kunne fungere hensiktsmessig. For Dewey benyttet jeg de samme ressursene som jeg

hadde benyttet i implementasjonen av FRBR-modellen, og dette var musikkdata spm har forbindelse med Billie Holiday. Disse emnekartene implementerte jeg manuelt inn i XTM-syntaksen. Ved implementasjonen av Dublin Core ville jeg i tillegg se hvordan det fungerte å benytte java for å lage XTM-filen automatisk. Jeg brukte her «Topic Maps For Java» (Tm4J), som er et javaverktøy for å lage, manipulere og publisere emnekart. Jeg brukte her ressurser hentet fra en database fra Open Archives Initiative (OAI).

Alle emnekartene jeg har laget har jeg publisert ved hjelp av et publiseringsverktøy for emnekart som er laget av Ontopia. Verktøyet heter Omnigator og er et gratis visualiseringsprogram som på en oversiktlig måte viser hvordan komponentene i emnekartet henger sammen, og er et hjelpemiddel i implementasjonsprosessen for å få oversikt.

Ved sammenfletting av emnekartene ble også Omnigator benyttet. Omnigator har en merge-funksjon (sammenflettings-funksjon) som automatisk fletter sammen de emnekartene man ønsker. Jeg laget først dobbelt opp av emnekartene, slik at jeg hadde to FRBR-emnekart, to Dublin Core-emnekart og to Dewey-emnekart. Jeg sammenflettet så emnekartene som var oppbygget på samme struktur. Deretter sammenflettet jeg emnekartene på kryss av standardene, og evaluerte hvordan sammenflettingen i praksis fungerte.

1.4 Resultater

Jeg har i oppgaven vist hvordan implementasjoner av emnekart basert på forskjellige standarder kan fungere.

Dublin Core er en settorientert metadatastandard som ikke er basert på nettverkstankegangen i motsetning til emnekartstandarden. Det var derfor nødvendig å legge inn egenkomponerte assosiasjoner i Dublin Core-standarden for at den skulle fungere i emnekart. DC kunne derfor ikke benyttes uredigert som standard.

Dewey er et klassifikasjonssystem som tar utgangspunkt i emner. Men her er emnene i størst grad satt sammen i hierarkier. Emnekart har også mulighet for implementasjon av hierarkier, og Dewey kunne bestå uendret ved implementasjonen. Men den utpregede nettverksstrukturen til emnekart ble litt begrenset med Dewey.

FRBR-modellen er en kompleks modell, men har komponenter som er nært knyttet opp mot komponentene i emnekartstandarden. Det var derfor relativt uproblematisk å benytte disse to standardene i kombinasjon.

Ved sammenfletting av like standarder kom det frem at strukturen i emnekartet ble beholdt på en god måte. Ved sammenslåing av emnekart på kryss av standardene derimot, ble strukturen i liten grad overlappende, og det ble i praksis to separate emnekart som hang sammen i noen få ledd, basert på like emner.

1.5 Oppgavens struktur

Denne oppgaven tar for seg hvordan man kan ta i bruk metadatastandarder, klassifikasjons-systemer og konseptuelle referansemodeller i forbindelse med emnekart. Først vil teorien for begrepene bli forklart, formatene vil deretter bli testet ut i praksis.

Kapittelinndelingen for oppgaven:

Kapittel 2 Dette er et teorikapittel som først og fremst tar for seg teorien rundt metadata

og beskrivelser av noen forskjellige metadatastandarder. Det sees også på begreper innenfor emnebasert klassifikasjon, og konseptuelle referansemodeller.

Kapittel 3 Her blir teorien rundt emnekart forklart. Kapittelet er et teorikapittel som forklarer emnekartstandarden, og går gjennom dens komponenter og syntaks.

Kapittel 4 Metadataformat representert ved Dublin Core blir her diskutert i forbindelse med emnekartstandarden, og en implementasjon av standardene blir utført og evaluert.

Kapittel 5 Deweys klassifikasjonssystem blir her diskutert i forbindelse med emnekart-standarden, og implementert i et emnekart.

Kapittel 6 Her blir det sett på den konseptuelle referansemodellen FRBR, og en implementasjon er blitt gjort slik at man skal se hvordan denne fungerer i forhold til emnekartstandarden.

Kapittel 7 I dette kapittelet er sammenfletting gjennomgått. Både sammenfletting av emnekart basert på like standarder og på tvers av standardene.

Kapittel 8 Til slutt følger en oppsummering av oppgaven.

Kapittel 2

Metadata

Metadata er fundamentet når det gjelder gjenfinning av data, og dette kapittelet starter med å forklare metadatabegrepet, og se på noen ulike metadatastandarder. Videre blir klassifikasjonsbegreper og konseptuelle referansemodeller beskrevet og forklart ved noen utvalgte standarder.

2.1 Hva er metadata?

Metadata er attributter som beskriver et objekt på en formell måte og defineres ofte som «data om data». I digitale bibliotek tenker man seg at metadata er informasjon/data om informasjonsobjekter. I denne oppgaven tenker vi i hovedsak på objekter i form av tradisjonelle dokumenter som finnes i biblioteket eller nettdokumenter. Men metadata-begrepet kan godt utvides til å gjelde andre objekter, som f.eks. personer eller gjenstander.

Metadata er mye brukt i forbindelse med katalogisering på bibliotekene, og der manifesterer metadataene seg som et katalogkort, en «bibliografisk post» eller lignende. Metadata representerer altså de dokumentene som skal inngå i et system ved hjelp av attributter, og fremstår derfor som et dokumentsurrogat på den måten at attributtene representerer et dokument i et system, uten at dokumentet selv er tilstede. Hensikten med bruk av metadata for dokumentsamlinger er å gjøre det lettere å gjenkjenne et dokument, og å finne det dokumentet man er ute etter. Brukeren må da få nok opplysninger fra metadataene til å kunne vite om dokumentet er av interesse. Det må også fremgå hvordan brukeren kan få tak i det aktuelle dokumentet.

Her følger noen eksempler på opplysninger som er knyttet til dokumenter og blir sett på som metadata:

- Hvem er forfatteren, evt. hvordan ble datasettet/informasjonen laget?
- Tittel på dokumentet
- Når ble informasjonen sist oppdatert?
- Hva er det datasettet/informasjonen handler om eller beskriver?
- Hvem har ansvaret for informasjonen, og hvordan kan man kontakte vedkommende?
- Har noen opphavsrett til datasettet/informasjonen?

- Hva er lagringsformat for informasjonen? (f.eks. XML, HTML ver 4.2, pdf, MPEG video, tab-separert fil, etc)

Alt omtalt i listen kan karakteriseres som metadata. Hvis det knyttes metadata til et dokument har vi bedre muligheter for å finne frem til og vurdere relevans for enkeltdokumenter innenfor mengden av alle dokumenter som er tilgjengelig.

Prosessen med å finne frem til og og arkivere metadata, har vært brukt i biblioteker i tusenvis av år og kalles her katalogisering.

For å utøve katalogisering på en standardisert måte, trenger vi regler for [4]:

Semantikk Katalogregler og andre regler som forteller hvilke elementer som skal være med i beskrivelsen.

Syntaks MARC-format, Dublin Core eller andre formater som sikrer en enhetlig presentasjon. Disse formatene blir forklart nærmere.

Notasjon Spesielt når katalogiseringen er «maskinleselig» må vi ha detaljerte formater som tillater utveksling og utnytting av katalogposter, f.eks. ISO 2709 (MARC-poster på magnetbånd), XML, HTML, SGML, o.l.

2.2 Metadatastandarder

Det finnes en rekke forskjellige metadatasystemer, og de varierer sterkt når det gjelder kompleksitet, generalitet, og hvilken notasjon som brukes.

Formatet som i stor grad har vært benyttet i biblioteksystemer er MARC. En hovedmotivasjon for å utvikle nye formater er behovet for å katalogisere nettdokumenter. Dette stiller nye krav til formatet på grunn av dokumentenes egenart sammenlignet med tradisjonelle papirdokumenter. Dessuten er det et ønske om at katalogiseringsarbeid skal kunne utføres av andre enn bibliotekarer. Dette har påvirket utviklingen av enkle formater som f.eks. Dublin Core. Det er en rekke forskjellige metadataformater, og noen er spesifikt utviklet for spesielle samfunn, eksempler på dette er metadata innenfor læringsmiljøet (LOM) og metadata for multimediainnhold (MPEG-7).

Metadataformatene MARC, LOM, Mpeg-7, og Dublin Core er forklart videre i kapittelet, men det er i hovedsak lagt vekt på Dublin Core siden dette formatet skal benyttes ytterligere i oppgaven i forbindelse med implementasjon av emnekart. MARC, LOM og Mpeg-7 er beskrevet for å få et innblikk i at det finnes mange ulike metadatastandarder, og noen er spesielt tilpasset spesielle miljø.

2.2.1 IEEE Learning object metadata (LOM)

LOM er et utkast til en standard som skal hjelpe til i beskrivelsen av læringsobjekter. Læringsobjekter blir da sett på som entiteter som blir benyttet i teknologistøttet læring, men objektene trenger ikke være digitale.

LOM sine mål er å [2]:

- hjelpe studenter og lærere å søke, hente og bruke læringsobjekter

- legge grunnlaget for deling og utveksling av læringsobjekter på tvers av teknologistøttede læringssystemer
- legge til rette for å lage læringsobjekter i enheter som kan bli kombinert eller oppsplittet på meningsfulle måter
- legge til rette for automatisk eller dynamisk komposisjon av individuelle leksjoner for enkeltpersoner

Strukturen til LOM tar utgangspunkt i ni metadatakategorier [2]:

Generelt Ressursens kontekstuavhengige og semantiske egenskaper

Livssyklus Egenskapene som har med ressursens livssyklus å gjøre

Meta-metadata Gjelder beskrivelsen selv og ikke ressursen som beskrives

Teknisk Tekniske egenskaper

Utdanning Læringsmessige og pedagogiske egenskaper ved ressursen

Rettighet Beskriver betingelser for bruk av ressursen

Relasjon Opplysninger som knytter ressursen til andre ressurser

Annotasjon Kommentarer angående den pedagogiske bruken av ressursen

Klassifikasjon Beskriver emnet som ressursen tilhører

Disse hovedkategoriene er igjen inndelt videre, og formatet har totalt ca 60 felt.

En overgang mellom LOM og Dublin Core er under utvikling, og formatet er interessant å se nærmere på når det gjelder digitale læringsmiljøer.

2.2.2 MPEG-7

MPEG-7 er en ISO-standard som er utviklet av MPEG (Moving Picture Expert Group). Standarden har formelt fått navnet: «Multimedia Content Description Interface» og er utviklet spesielt for å beskrive, indeksere og behandle multimediainnhold på digital form. Målet er å tilby et rikt og standardisert verktøysett for å klassifisere, indeksere, søke og innhente multimediainnhold.

Standarden består i hovedsak av tre deler [5]

- En lyd- og bildedel som definerer teknologi for lydmessige og visuelle karakteristikker som farge, form, lydeffekt, melodi og liknende.
- Beskrivelsesskjema for multimedia. Strukturerte skjema for en hierarkisk beskrivelse av innholdet ved hjelp av metadata og de audiovisuelle karakteristikkene fra andre deler.
- Description Definition Language (DDL), et språk for å definere nye beskrivelsesskjema eller utvide eksisterende skjema.

MPEG-7, som andre MPEG standarder, beskriver kun hvordan informasjonen skal struktureres og kodes, samt et nødvendig minimalt systemverktøysett. Den spesifiserer ikke implementering av koding, eller på hvilken måte metadata skal produseres eller brukes når den først er kodet i MPEG-7 formatet.

2.2.3 MARC

MAchine Readable Cataloging (MARC) er en metadatastandard (ISO 2709) som brukes til å merke bibliografisk informasjon, og til å utveksle slik informasjon elektronisk. Systemet går ut på at man merker de enkelte bibliografiske elementene med koder (tag-er). Standarden sier ikke noe om hvilke felter som skal brukes når, og hvordan de skal fylles ut (ut over det som er minimum), kun hvordan feltene skal struktureres som en lang streng.

MARC-formatet er basert på kortkatalogens måte å organisere bibliografiske data på, og er bare en automatisering av denne. Ulempen med dette er at da MARC-standarden ble utviklet tok de ikke hensyn til hvilke muligheter ny teknologi ga, og fulgte kun gammelt oppsett.

En MARC post inneholder et «hode» som forteller hvor dataene ligger i posten (start og slutt på posten), og selve dataene som kan være av variabel lengde. For å identifisere feltene benytter man feltkoder og delfeltkoder. Hovedfeltene identifiseres med et tresifret nummer 001-999, mens delfeltene vises ved: $+bokstav, eks.: $a eller $b osv.

Hovedfeltene til MARC [6]:

0XX Koder, numre, klassifikasjon m.m.

1XX Hovedordningsord

2XX Tittel m.m.

3XX Fysisk beskrivelse

4XX Serie

5XX Noter

6XX Emneord

7XX Biinførsler

8XX Serie i annen form

9XX Se-henvisninger

Alle feltene over kan igjen deles opp. F.eks. har «[2XX] Tittel m.m.» disse feltene under seg [6]:

240 Standardtittel

241 Originaltittel

245 Tittel

246 Parallelltittel

250 Utgave

254 Musikktrykkets fysiske presentasjon

255 Kartografisk materiale

256 Filkarakteristika

260 Utgivelse, distribusjon

Delfelter for 245 tittel ser slik ut [6]:

$ a - Hovedtittel

$ b - Annen tittelinformasjon

$ c - Ansvarsangivelse

$ h - Generell materialbetegnelse

$ n - Nummer for del av verk

$ p - Tittel for del av verk

$ w - Sorteringsfelt for delfelt $ a

Det finnes forskjellige MARC dialekter og felter/delfelter kan variere fra land til land og fra system til system selv om hovedtrekkene er like. Forskjellige systemer er: BIBSYS-MARC, NORMARC, DANMARC, LCMARC (MARC21), UNIMARC.

MARC-poster er komplekse og er ikke åpenbart systematiske. Postenen inneholder en rekke felter man kan fylle inn i, og som ikke gir noen entydig forklaring på hvordan ting bør skrives. Siden MARC ikke er laget med tanke på ny teknologi, er det svært vanskelig å konvertere formatet fullstendig til datamodeller. For nettdokumenter har man derfor utviklet Dublin Core.

2.2.4 Dublin Core

Dublin Core (DC) er en metadatastandard med et sett enkle men effektive elementer [7]. Disse elementene danner en standard for hvilke metadata som kan registreres om et dokument. Det er mulig å innlemme metadata som en del av selve primærdokumentet (f.eks. ved hjelp av HTML eller SGML), eller det kan legges ved for eksempel i egne databaser. DC skal i hovedsak gjøre det så enkelt å legge til metadata på nettdokumenter at det ikke kreves ekspertise på linje med hva som trengs for vanlig katalogisering i biblioteker. Ved at det er enkelt, blir det i stor utstrekning tatt i bruk, og det kan fylle det store behovet om å få metadata inn i nettdokumenter for å gjøre presisjonen ved søking bedre og informasjonsgjenfinningen lettere [4].

Det første møtet om Dublin Core ble holdt i 1995 i Dublin Ohio, noe som preger navnet. «Core», betyr «kjerne» og henspeiler på at Dublin Core består av et sett av kjernefelt[8]. Man kan si at Dublin Core har 15 kjernefelt/elementer som gjør det mulig å beskrive viktige opplysninger om et dokument.

De 15 elementene (kjernefeltene) er [9]:

- Tittel (title)
- Forfatter/opphavsmann (creator)
- Beskrivelse (description)
- Utgiver (publisher)
- Annen bidragsyter (contributors)
- Dato (date)
- Emne (subject)

- Type (type)
- Format (format)
- Identifikator (identifier)
- Kilde (source)
- Språk (language)
- Relasjon (relation)
- Dekning (coverage)
- Rettigheter (rights)

Dublin Core har i utgangspunktet ikke noen bestemt syntaks, og det er stor valgfrihet i hvordan man kan bruke feltene. Det er f.eks. ikke nødvendig å fylle ut informasjon om alle de 15 elementene for alle dokumenter, og det er mulig å gjenta et element flere ganger ved behov. Dette gjør det mulig å tilpasse oppsettet etter ønsket bruk. DC gir derfor mulighet for stor fleksibilitet, i tillegg til å være enkelt å ta i bruk.

Ulempen med denne enkelheten som DC representerer, er at man registrerer mindre informasjon om dokumentene enn man gjør i andre metadatasystemer som f.eks. MARC. Dette kan by på problemer hvis man f.eks. skal konvertere dokumenter som benytter MARC-formatet til å bli beskrevet i DC, siden mye informasjon kan gå tapt.

For å ha mulighet til å legge til litt ekstra informasjon og kunne strukturere informasjonen litt mer, har flere av elementene kvalifikatorer (qualifiers). Slike kvalifikatorer brukes til å spesifisere innholdet til elementene ytteligere, og er opplysninger som sier noe mer om hvordan elementene skal tolkes. Eksempler på dette kan være å spesifisere «dato» ved å si at det er en utgivelsesdato, dato for oppdatering o.l. Eller det kan være snakk om at en klassifikasjonskode blir kvalifisert til å være hentet fra Dewey Decimal Classification Ed.21. osv. (Dewey kapittel 2.3.1 på neste side).

Når det gjelder implementasjonen av DC, blir det som regel benyttet XML eller HTML når det skal leses maskinelt. Men DC skal kunne brukes av alle typer miljøer, og på alle typer dokumenter som tekst, lyd, bilder og objekter, og kan også brukes i metadataposter i databaser osv. DC er en standard med 15 gitte elementer som kan implementeres etter som det passer formålet.

Under følger et eksempel som viser hvordan DC kan implementeres ved hjelp av HTML/XHTML-tagger. Eksempelet er hentet fra Dublin Core Metadata Initiative [10], og viser tittelen på dokumentet som omhandler nettopp det å forklare hvordan DC kan implementeres. Videre i koden er det elementer med innhold for forfatter, identifikasjon for siden, format, type osv.

2.3 Emnebasert klassifikasjon

Metadataformater hjelper til med å beskrive dokumenter slik at det blir lettere å finne dem igjen. For å strukturere dokumentene bedre kan man ta i bruk emnebasert klassifisering.

Emnebasert klassifisering organiserer og grupperer objekter etter hvilket emne de omhandler. Slik klassifisering kan gjøres uavhengig av om dokumentet som klassifiseres er en bok, et lydopptak eller en elektronisk fil [11]. Klassifisering går generelt ut på at man deler inn objekter

```
<head profile="http://dublincore.org/documents/dcq-html/">
<title>Expressing Dublin Core in HTML/XHTML meta and link elements</title>
<link rel="schema.DC" href="http://purl.org/dc/elements/1.1/" />
<link rel="schema.DCTERMS" href="http://purl.org/dc/terms/" />

<meta name="DC.title" lang="en" content="Expressing Dublin Core
in HTML/XHTML meta and link elements" />
<meta name="DC.creator" content="Andy Powell, UKOLN, University of Bath" />
<meta name="DCTERMS.issued" scheme="DCTERMS.W3CDTF" content="2003-11-01" />
<meta name="DC.identifier" scheme="DCTERMS.URI"
content="http://dublincore.org/documents/dcq-html/" />
<link rel="DCTERMS.replaces" hreflang="en"
href="http://dublincore.org/documents/2000/08/15/dcq-html/" />
<meta name="DCTERMS.abstract" content="This document describes how
qualified Dublin Core metadata can be encoded
in HTML/XHTML &lt;meta&gt; elements" />
<meta name="DC.format" scheme="DCTERMS.IMT" content="text/html" />
<meta name="DC.type" scheme="DCTERMS.DCMIType" content="Text" />
</head>
```

Figur 2.1: Kodeeksempel for Dublin Core

i klasser. Objekter med samme emne og med flere like kjennetegn, hører inn under samme klasse. Dette er veldig nyttig for å kunne se hva som hører sammen, og det blir lettere å finne frem i mengder av data. Man slipper å se gjennom alle dokumenter for å finne det man skal ha. I stedet kan kategorien det er interesse for finnes. Søkemassen blir dermed veldig redusert.

Når bøker blir klassifisert er det viktig at hver bok kun blir satt under én kategori, og dette gjelder også hvis det er flere kopier av en bok. Uten denne regelen vil det bli veldig forvirrende for brukeren, som da må sjekke flere steder om boka er der.

Et problem er at klassifiseringen skjer manuelt og at ulike personer kan ha forskjellig mening om hvilken klasse boken hører inn under. For å hjelpe til med dette er det viktig å ha klare systemer for hvordan det skal gjøres.

Metadata er sentralt når det gjelder klassifisering. Ved hjelp av den formelle og strukturerte informasjonen metadata gir om dokumenter, er det lettere å kunne sortere og klassifisere dem. Sammenhengen i dette er at emnene som benyttes til klassifikasjon blir fylt inn i metadatapostene som strukturerer dataene. Vi kan si at metadata beskriver objektene som skal klassifiseres, mens emnene blir brukt til å klassifisere dem.

2.3.1 Dewey

Dewey er et desimalklassifikasjonssystem utviklet av amerikaneren Melvil Dewey. Systemet ble utgitt i 1876, men har blitt utvidet og revidert en rekke ganger. Dewey-systemet brukes nå i mer enn 135 land og er oversatt til 35 forskjellige språk. Det er i norske biblioteker det mest brukte klassifikasjonssystemet, og har vært i bruk siden slutten av 1890-tallet [12]. Jeg har i denne oppgaven tatt utgangspunkt i den femte norske utgaven [11], som baserer seg på Dewey Decimal Classification, Ed.21.

Deweys desimalklassifikasjon (DDK) er et klassifikasjonssystem som ved hjelp av notasjon ordner bøker og annet informasjonsmateriale etter innhold. Notasjonen som blir brukt er

klassenummer i form av tall. Tallet gis til dokumentene ettersom hvilket fag dokumentet handler om. Deweys desimalklassifikasjon er basert på tankegangen om at man kan dele all menneskelig viten inn i ti hovedklasser. Disse hovedklassene er navngitt med tall fra 0 til 9, og vanligvis skrevet med tre sifre [11]:

000 Generelle emner

100 Filosofi, overnaturlige fenomener, psykologi

200 Religion

300 Samfunnsvitenskap

400 Språk og språkvitenskap

500 Naturvitenskap og matematikk

600 Teknologi (anvendt vitenskap)

700 Kunst og underholdning

800 Litteratur og litteraturvitenskap

900 Geografi, historie og deres hjelpefag

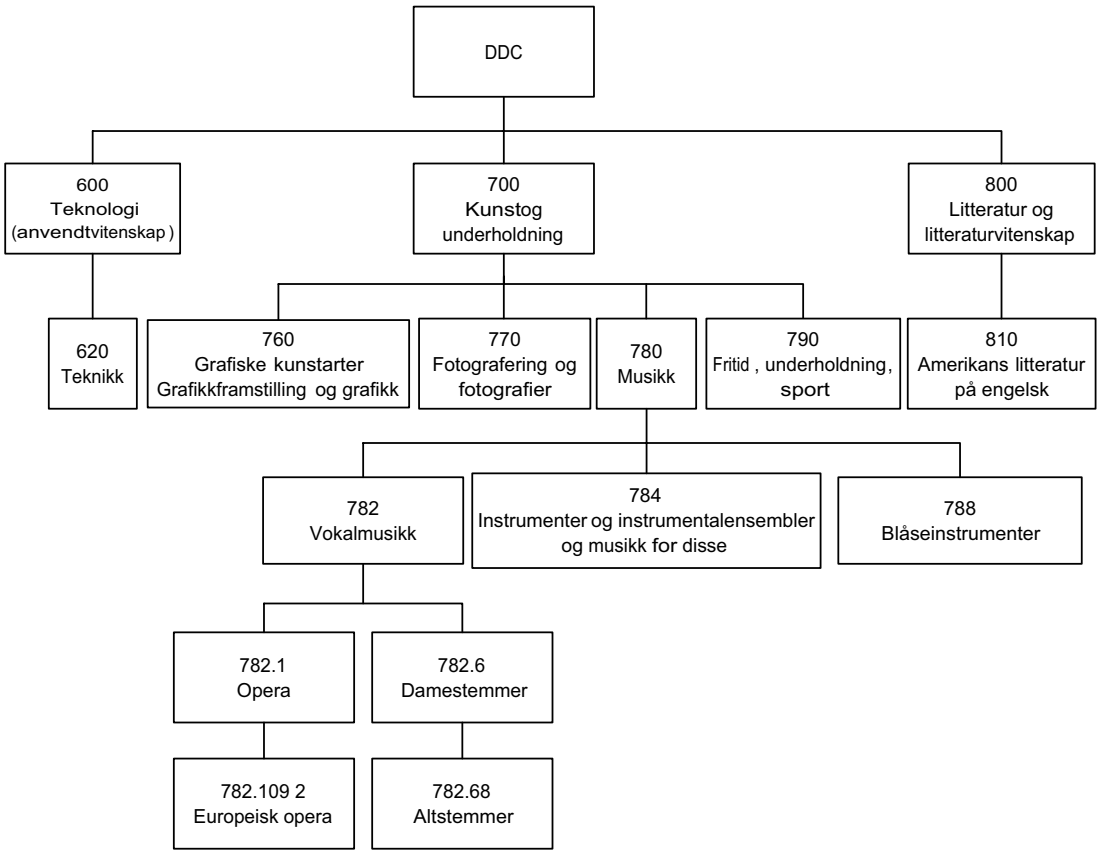
Deweys hovedklasser er vide, og ikke begrenset til enkeltfag, men indikerer i stedet et fagområde. For å ha mulighet til å spesifisere grundigere hvilket fag et dokument omhandler, tar DDK i bruk en hierarkisk oppbygging. Alle hovedklassene kan deles i ti underavdelinger, og hver av disse igjen i ti seksjoner. Det blir dermed 1000 grupper som betegnes ved tallrekken fra 000 til 999. Hver av disse kan deles videre opp, og da settes det alltid punktum etter det tredje tallet. Liste over hovedklassene og nivået under dette er vist i vedlegg A på side 91.

Deweytallet sammen med en bokstavkode fungerer som en hyllesignatur, «adresse» for hvilke hylle dokumentet er plassert i. Hyllesignaturen blir festet direkte på bokryggen, slik at det skal være lett å se den når bøkene står i bibliotekhyllene. Det er dermed lett for de ansatte å kunne sortere bøkene slik at de er plassert sammen med andre bøker som omhandler samme fag. Og det blir dermed lettere for brukerne å finne bøker innen ett gitt fag/emne.

Hvis brukeren skal finne en bok som omhandler for eksempel underholdning, så kan han gå til en hylle hvor Deweynumrene er på 700-tallet, og vet da at der vil han finne bøker om underholdning generelt. Men han er kanskje ikke interessert i all type underholdning, men kun «musikk», noe som vil finnes under 780-tallet. 780 er igjen oppdelt mer spesifikt i for eksempel «opera» og igjen i «Europeisk opera» som var det han spesifikt var ute etter. Og finner her for eksempel boken som omhandler operaen «Tosca», eller eventuelt CD-utgivelsen for denne operaen.

Deweytallene i seg selv skiller ikke ut hva som er tekster, lyd eller videoopptak, men Dewey-tabellen kommer med forslag til hvordan det kan gjøres. Eksempler er enten å legge til flere tall bak det opprinnelige Deweynummeret, eller legge til bokstaver foran tallet. Dette kan være M788.92 for å vise at det er noter for trompet, eller L788.92 for å vise at det er en lydfil med trompet.

Figur 2.2 på neste side viser et lite subsett av hvordan Dewey er bygget opp, og viser hvordan den hierarkiske strukturen fungerer. Jo lenger ned i hierarkiet man kommer, jo mer spesifikt blir faget.



Figur 2.2: Modell over ett subsett av Deweynummer

2.3.2 Kontrollert vokabular

Kontrollert vokabular er et vidt begrep, men i denne sammenhengen mener vi en liste med navngitte emner som kan bli brukt ved klassifisering. Vokabularet består av termer som er bestemte navn for et bestemt konsept. Det er vanlig å skille mellom termer og konsepter ved å si at en term er navnet på et konsept, og at det samme konseptet kan ha flere forskjellige navn, og også at den samme termen kan navngi flere emner [1].

Forskjellen mellom vokabularet når det gjelder metadata og kontrollert vokabular, er at metadatavokabularet sier noe om egenskapene til objektene, mens det kontrollerte vokabularet inneholder emner som blir brukt til klassifisering.

Hensikten med kontrollert vokabular er å forhindre at forfattere definerer uhensiktsmessige termer. Dette kan være termer som er for vidtomfangende, eller for spesifikke, skiller seg ut fra de termene som er blitt brukt om samme emne tidligere osv. Forskjellige forfattere kan f.eks. benytte termer som «topic navigation maps» eller «topic map» om samme emne, mens det kontrollerte vokabularet kan tvinge alle til å benytte «topic maps». Når alle benytter de samme termene om de samme emnene blir det lettere å klassifisere og organisere informasjonen.

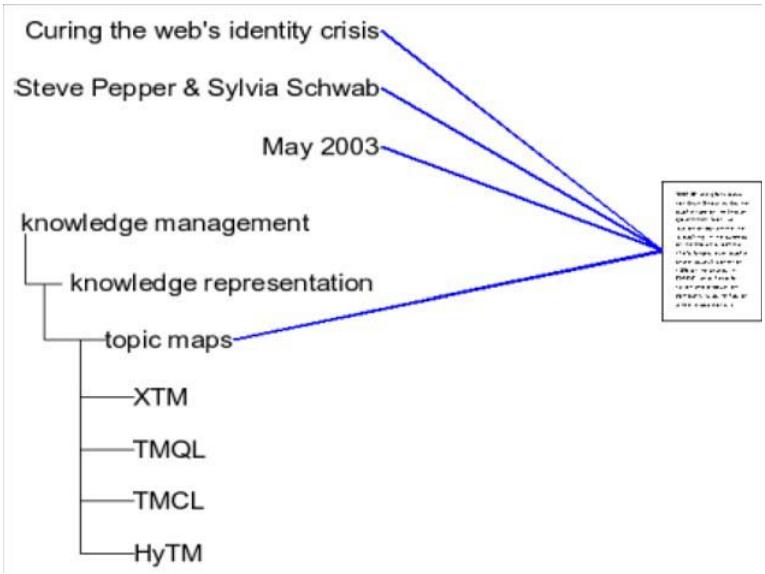
Den helt enkleste formen for kontrollert vokabular er kun å skrive en liste med ord og si at disse skal benyttes. Men det finnes også mer avanserte skjemaer og strukturer som taksonomier, tesauruser og ontologier, og disse skal bli videre forklart.

2.3.3 Taksonomi

Begrepet taksonomi har blitt benyttet i ulike sammenhenger, men er i utgangspunktet en form for abstrakt struktur. Taksonomi blir i denne settingen sett på som emnebasert klassifikasjon som strukturerer termene i det kontrollerte vokabularet inn i hierarkier [1].

Fordelen med taksonomi er at den grupperer relaterte termer, og strukturerer disse slik at det blir lettere å finne den termen man skal benytte. Dette gjør det lettere for brukeren å se at f.eks. termene «topic maps» og «XTM» er relatert med hverandre, og det blir da lettere å se hva som skal benyttes i hvilken sammenheng.

Taksonomien hjelper brukerne ved å beskrive emnene. Når det gjelder metadata er det ikke forskjell mellom et enkelt kontrollert vokabular og en taksonomi, men metadataene relaterer kun objekter til emner, mens taksonomien strukturerer emnene i hierarkier. Taksonomien beskriver altså emnene som blir brukt til klassifisering, og som kan bli brukt i metadata, men er ikke i seg selv metadata.



Figur 2.3: Bruk av taksonomi i metadata [1]

I figur 2.3, blir forskjellen mellom metadataene og taksonomien forklart. Her viser de tykke linjene på skrå metadata, mens de tynne linjene mellom ordene danner taksonomien og er en del av den emnebaserte klassifikasjonen. Vi ser at de tykke linjene er informasjon om dokumentet, mens de tynne linjene er strukturerte termer for hva dokumentet handler om. Fordelen med dette er at hvis vi har et annet dokument som omhandler noe av det samme, så vet vi hvordan emnene er relatert til hverandre.

Taksonomien hjelper brukeren ved å gi informasjon om konseptene. Men det er som regel ikke all informasjon som blir fanget opp og vist i taksonomien. I figuren er det f.eks. ingen ting som forteller at:

- «XML Topic Maps» er synonymt med «XTM»
- «topic navigation maps» er synonymt med «topic maps», men at man bør benytte termen «topic maps»
- det finnes en relasjon mellom topic maps og emnebasert klassifikasjon, og også mellom topic maps og semantisk web

- det finnes en relasjon mellom XTM, XML, HyTM og SGML

Dette får konsekvenser for sluttbrukeren fordi han da må søke på nøyaktig riktig term og lete på de riktige plassene i hierarkiet for å finne den termen han er ute etter.

2.3.4 Tesaurus

Tesaurus er i hovedsak en videreføring av taksonomi, og er bedre egnet til å beskrive verden ved å i tillegg til å strukturere emner i hierarki, slik det blir gjort med taksonomi, også tillater å legge utsagn til emnene. I følge ISO standarden for tesaurus (ISO2788) kan følgende utsagn legges til [1]:

BT - broader term refererer til termen som er plassert over i hierarkiet. Denne termen som er høyere i hierarkiet har en bredere mening, og er mindre spesifikk i forhold til termen som kommer under. Det finnes også en motsatt mulighet kalt, NT, for «narrower term».

Man kan si at taksonomier er tesauruser som kun benytter BT og NT for å bygge et hierarki. Alle tesauruser inneholder derfor en taksonomi.

SN - scope note er en setning som er lagt til termen for å forklare meningen til termen bedre. Dette kan være nyttig i sammenhenger hvor meningen av termen ikke kommer frem av konteksten.

Et eksempel er at brukere ofte benytter termen «XTM» når de egentlig mener «topic maps». Ved å legge til en liten forklarende setning i form av en SN, kan man forklare at XTM er XML formatet for Topic maps, og det blir dermed lettere for brukeren å skjønne sammenhengen mellom termene.

USE sier noe om at man heller burde bruke en annen term i istedet. Termene det gjelder er da synonymer og USE indikerer hvem av ordene som skal brukes. Eksempel: under «topic navigation maps» kan det stå: USE «topic maps». Dette indikerer da at man oppfordrer til å benytte «topic maps» i stedet for. Det finnes også et inverst utsagn her som heter UF, som settes under «topic maps» og refererer til «topic navigation maps».

TT - top term indikerer den øverste termen i hierarkiet.

RT - related term refererer til en term som er relatert med termen man ser på. Vi kan for eksempel si at til termen «topic maps» har man relaterte termer som: «emnebasert klassifikasjon» og «ontologier».

Vi kan si at tesaurus står for et mye rikere vokabular for å beskrive termene enn det taksonomier gjør. Og ved å bruke tesaurus i stedet for taksonomier kan man løse mange praktiske problemer når det gjelder klassifisering av objekter og søking på dem.

Tesaurus i praksis

Tesaurus sikrer konsistent termbruk, og gir en standard for praksis mellom personer som samarbeider. Man slipper å kun ha ett emneord å velge i, slik som under klassifisering. Med hierarkiske tesaurus har man et verktøy som kontrollerer vokabularet, spesielt for indeksering og gjenfinning i spesifikke domener.

Når man skal bygge en tesaurus starter man med å skrive en liste over substantiv som er viktige for et dokument. Dette kan være enkeltord eller sammensatte uttrykk. Substantiv er det som betyr mest for en tekst. Adverb må ikke brukes i en tesaurus fordi dette forteller lite om hva teksten handler om. Man må bestemme om ordene skal stå i entall eller flertallsform, om det er lov med forkortinger av ord osv. Som regel vil det være mest hensiktsmessig å ikke bruke flertallsform av ord, og heller ikke forkortelser.

For å finne ut hvilke ord som skal være med i en tesaurus har man ulike metoder. Det er ennå ingen optimal metode for å gjøre dette helt automatisk, men man kan komme et stykke på vei. Man kan f.eks. lage en stoppordliste hvor ord som: er, har, å, og, ikke osv., blir fjernet. Dette er ord som er i alle dokumenter og som ikke forteller noe om dokumentet handler om for eksempel hunder eller snøsmelting. Det er også mulig å finne frekvensen av alle ordene i en dokumentsamling som omhandler et tema, og sammenlikne denne med frekvensen av en generell referansesamling. Ord som forekommer ofte i alle dokumenter betyr ikke noe spesielt for et dokument. Mens ord som forekommer ofte i orginalsamlingen, men sjelden i referansesamlingen tyder på at disse er viktige for det spesielle temaet som samlingen omhandler.

2.3.5 Ontologi

En ontologi er en modell for hvordan man kan forklare verden innenfor et bestemt fagområde (digitale bibliotek, fiskeri, medisin etc.), og inneholder termer, spesifikasjon av disse termene og hvordan de er beslektet.

Ontologier går utover definisjonen av en tesaurus ved at den tillater mulighet til å skape uendelige mengder av forskjellige semantiske relasjoner, og den er også en del av utviklingen av semantisk web dvs. det som ligger i ønsket om å utvikle weben til å bli et verktøy som gir muligheter for mer direkte kommunikasjon [13].

Tidligere har vi sett på ulike vokabular i forbindelse med å forklare emner benyttet i emnebasert klassifikasjon. Disse har alle hatt kontrollert vokabular for emnebeskrivelse. Ontologier har i motsetning fritt vokabular.

Taksonomien har kun relasjoner bygget opp av broader/narrower-termer i form av et hierarki. Termene som blir beskrevet er frie, men språket som blir brukt for å beskrive disse termene er bestemt.

Tesaurus utvider dette ved å ta i bruk RT, USE og SN for å forklare termene bedre. Men i tesauruser er språket gitt ved at man kun kan benytte disse bestemte utvidelsene. Hadde det ikke vært for dette kunne Tesaurus blitt kalt en ontologi, men i praksis blir ikke tesaurus sett på som ontologi på grunn av innskrenkningen av muligheter som det gitte vokabularet gir.

Når det gjelder ontologier har forfatteren av emnebeskrivelsene lov til å definere språket helt fritt etter hva som er mest hensiktsmessig i forhold til settingen. Ontologi innen datateknologi har forankring i kunstig intelligens, men blir stadig mer brukt i informasjonsgjenfinning. Innen informajsonsgjenfinning har man laget standarden Topic Maps (emnekart) som er bygget opp som et rammeverk basert på ontologier. Emnekart skal bli diskutert nærmere i kapittel 3 på side 35.

Det finnes forskjellige ontologier, og forskjellige bruksområder for disse. Konseptuelle referansemodeller er et av områdene som tar i bruk ontologier i sin oppbygging.

2.4 Konseptuelle referansemodeller

Et viktig poeng når det gjelder informasjonsgjenfinning, er at det ikke bare dreier seg om å finne et bestemt dokument, men også det å kunne orientere seg og finne frem i en bibliografisk struktur. Ontologier, som beskrevet over, gjør det lettere å navigere i termer og se sammenhengene mellom disse. Men det er også viktig å ha strukturer på selve dokumentene. Gjenfinningsdata dreier seg ikke bare om beskrivelse av hvert enkelt dokument i en samling, men om å sette disse dokumentene i sammenheng [2].

I forhold til f.eks. museer, hvor dokumentene består av avbildninger av gjenstander eller fotografier av steder, vil det være nødvendig med en beskrivelse utover det rent innholdsmessige. Gjenstandene og fotografiene må plasseres i en sammenheng som kan gi dem større informasjonsverdi, som opplysninger om datoer, hvilke personer som er avbildet, hvilket sted de er avbildet osv. Konseptuelle referansemodeller er laget for å hjelpe til med dette.

Videre i kapittelet vil det bli sett på noen ulike konseptuelle referansemodeller; CRM, ABC-ontology og FRBR-modellen. FRBR-modellen vil bli forklart grundigere enn de andre modellene siden denne senere i oppgaven vil bli benyttet i forbindelse med implementering av emnekart.

2.4.1 CIDOC CRM

CIDOCs[1] konseptuelle referansemodell, CRM, er en modell som er utviklet for å vise strukturer og sammenhenger i museumsdata. CRM er et felles semantisk rammeverk for informasjon om kulturelle gjenstander, og skal skape en basis for en felles forståelse.

Modellen er spesifikt relatert til museers og arkivers behov. Målet er at den skal fungere som et begrepsmessig lim som kan knytte sammen ulike kilder for kulturell dokumentasjon. CRM tar utgangspunkt i hendelser og knytter gjenstander sammen på bakgrunn av disse hendelsene. På denne måten blir det relasjoner mellom informasjon uten at informasjonen bindes til spesifikke klassifikasjonssystemer og kunnskapskategorier. Man unngår på denne måten at informasjon blir utilgjengelig ved at den struktureres i avgrensede hierarkier. CRM sier ikke noe om hva kulturelle institusjoner bør dokumentere, men forklarer logisk hva som faktisk kan dokumenteres. På denne måten bidrar databasestrukturen til å tilrettelegge for utveksling med andre databaser [14].

2.4.2 ABC-ontologi og modell

ABC-ontologi er en konseptuell modell som er utviklet for å hjelpe til med interoperabiliteten mellom metadataontologier fra forskjellige domener.

Målene for modellen er [15]:

- Lage en konseptuell basis for å forstå og analysere eksisterende metadataontologier og instanser.
- Gi veiledning og retningslinjer til de som skal utvikle ontologier.
- Utvikle en konseptuell basis for automatisk mapping mellom metadataontologier.

ABC-modellen inneholder en rekke basisentiteter og relasjoner som er vanlige på tvers av ulike metadataontologier som f.eks. tids og stedsangivelser. Ved utarbeidelsen av modellen ble det

[1] International Committee for Documentation of the International Council of Museums

samarbeidet med flere ulike standarder og modeller, bl.a. Dublin Core Metadata Initiative, og IFLA Functional Requirements for Bibliographic Records (FRBR). I den seneste versjonen av ABC-modellen har også museumene kommet på banen representert ved CIDOC/CRM.

IFLAs FRBR-modell er stadig mer tatt i bruk innen digitale bibliotek, og er den av de konseptuelle modellene som vil bli sett nøyere på i oppgaven.

2.4.3 FRBR-modellen

I 1998 utga IFLA den endelige rapporten om Functional Requirements for Bibliographic Records [16]. Dette var et resultat av mange års arbeid fra en studiegruppe av IFLA[2]. Studiegruppen hadde i oppdrag å gå igjennom de eksisterende katalogiseringsprinsippene for å se om det var rom for forbedringer. Katalogteknologien hadde vært utsatt for store endringer ved bl.a. økende bruk av edb-baserte systemer, men dette hadde ikke ført til endringer av katalogiseringsreglene. Det var derfor behov for å undersøke hvordan det kunne lages en struktur som var lettere å forstå for datasystemene, og som kunne hjelpe til med en strukturell utveksling av data mellom ulike systemer.

FRBR-modellen er et rammeverk for de bibliografiske elementene, og er et grunnlag for datasystemer som kan la brukerne navigere i de bibliografiske postene på en enkel og strukturert måte. På norsk blir oversettelsen av FRBR: «funksjonskrav til bibliografiske poster» [17], noe som viser til at FRBR-modellen er utviklet for å ta utgangspunkt i brukernes krav til bibliografiske posters funksjonalitet.

Studiegruppen som laget FRBR-modellen definerte den som: «en konseptuell modell som representerer et generalisert syn på det bibliografiske universet» [18].

Det er lettere å forstå denne definisjonen ved å se på noen av de essensielle ordene: «Konsept» er et ord som er synonymt med «begrep», og dette kan igjen defineres som en «forestilling eller idé». Termen «modell» kan sees på som en teoretisk fremstilling av et forhold, som kan være til hjelp ved analyser og beregninger. FRBR-modellen er ikke en gjengivelse av en gjenstand i forminsket målestokk slik som andre modeller ofte er, men er i stedet en begrepsmessig fremstilling som danner utgangspunktet for en analyse av «det bibliografiske universet».[18]

FRBR-modellens mål er i hovedsak å gjøre det lett for brukeren å *finne* entitetene han er ute etter, å kunne *identifisere* dem, å kunne *velge* entitetene han vil ha, og til slutt å *få tak i* den spesielle entiteten.

For å oppnå dette ble FRBR-modellen utviklet ved hjelp av ER-analyse (Entitet-Relasjon-analyse). ER-analysen gjorde det mulig å bestemme hvilke komponenter/entiteter som skulle inngå i modellen, hvilke egenskaper disse komponentene skulle ha, samt relasjonene mellom komponentene. Modellens hovedgrupper er derfor entiteter, attributter og relasjoner, og disse vil det bli gjort rede for videre i oppgaven.

Entiteter

FRBR-modellens entiteter, er de komponentene som er av kjerneinteresse for brukerne av bibliografiske data. Disse kan deles i tre grupper[2]:

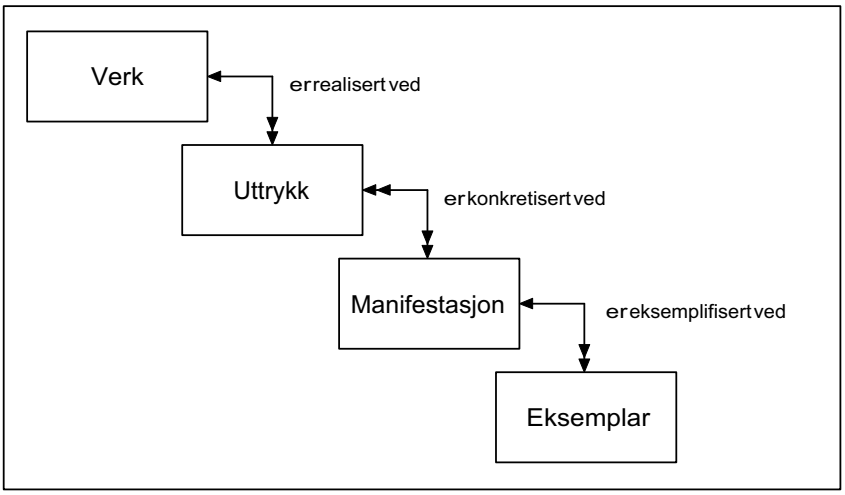
- Gruppe 1: Produktene
- Gruppe 2: Ansvaret for produktene

[2] International Federation of Library Associations and Institutions

- Gruppe 3: Emnet for produktene

Gruppe 1 - Produktene

Gruppe 1 inneholder entiteter som er av intellektuell eller kunstnerisk art, og som beskriver et produkt. Figur 2.4, viser de fire entitetstypene som hører til under denne gruppen: verk, uttrykk, manifestasjon og eksemplar.



Figur 2.4: Entiteter i gruppe 1, og deres relasjoner [2]

- **Verk**
  Entitetstypen «verk» inneholder abstrakte entiteter. Den inneholder ingen informasjon om fysiske objekter som kan pekes på. For å kunne si hva som er et verk må man se om det ligger et selvstendig intellektuelt eller kunstnerisk arbeid bak. Siden verket er abstrakt kan man kun kjenne igjen verket når det er realisert ved hjelp av ett eller flere uttrykk.

  Hvis et verk blir gjort om fra én kunstnerisk form til en annen, vil det bli sett på som et nytt verk. Dette gjelder også hvis verket blir utsatt for stor grad av uavhengig, intellektuelt eller kunstnerisk arbeid. Det er likevel viktig å se at selv om det er forskjellige verk, så er det en relasjon mellom dem, og dette er det mulig å uttrykke ved hjelp av FRBR-modellen. Vi kan vise at et verk kan være emnet for et annet verk. Et eksempel på dette er forfatteren Viktor Hugo som skapte verket «Lés Miserables». Dette verket ble senere emne for en musikal med samme navn og med samme handling, men som likevel var et nytt verk fordi det var et nytt intellektuelt og kunstnerisk arbeid. FRBR-modellen gjør det da mulig å vise sammenhengen mellom disse to verkene som har mye til felles med en «verk til verk»-relasjon.

- **Uttrykk**
  Entitetstypen «uttrykk» danner realiseringen og virkeliggjøringen av et verk. Dette kan skje ved at verket blir skrevet ned med tegn slik at det kan leses som tekst, eventuelt som musikalsk eller koreografisk notasjon. Verket kan også virkeliggjøres ved lyder, bilder, gjenstander, bevegelser osv.

  Et verk kan bli realisert ved hjelp av mange uttrykk. Et partitur som viser notene til et verk, er et uttrykk. Når notene blir spilt og omgjort til lyd, blir dette et nytt uttrykk av det samme verket. Ulike oversettelser blir også sett på som uttrykk av et verk.

- **Manifestasjon**
  En manifestasjon av et verk viser den konkrete utformingen av et uttrykk. En slik konkret utforming vil si at uttrykket av verket har blitt gjort fysisk tilgjengelig på et medium som for eksempel papir, lydbånd eller videoopptak.

  Et eksempel på en manifestasjon, er boken «Lés Miserables» av Viktor Hugo i et bestemt opplag, med et spesielt ISBN-nummer. Det kan være flere bøker med samme ISBN-nummer, men det er akkurat det samme innholdet.

- **Eksemplar**
  Et enkelt eksemplar av en manifestasjon blir sett på som et eksemplar.

  Eksemplaret, kan være den éne spesielle boken av «Lés Miserables», som for eksempel er signert av forfatteren, eller som noen har stående i bokhyllen sin.

Gruppe 2 - Ansvar

Gruppe 2 tar for seg de entitetstypene som kan være ansvarlige for de forskjellige produktene som er omtalt i gruppe 1.

- **Person**
  En person i FRBR-modellen blir sett på som et individ som er involvert i å skape eller fremføre et verk. Entitetstypen «person» kan også inneholde personer som er emner for et verk, for eksempel individer som blir omtalt i biografier.

- **Korporasjon**
  Entitetstypen «korporasjon» tar for seg en organisasjon eller en gruppe av individer. Det kan også være grupper av organisasjoner som opptrer som en enhet og kan identifiseres ved et gitt navn.

Gruppe 3 - Emne

Gruppe 3 tar for seg de entitetstypene som kan være emner for verkene. Innholdet i gruppe 3 er et begrep, en gjenstand, en hendelse eller et sted.

- **Begrep**
  Entitetstypen «begrep» tar for seg alt som kan være emne for et verk. Dette kan være en abstrakt tanke eller en idé om verket. Ved å benytte denne entitetstypen er det mulig å lage relasjoner mellom verket og begrepet som er emne for verket.

- **Gjenstand**
  En gjenstand er et fysisk objekt. Ved å lage en entitetstype for gjenstander kan man lage relasjoner for å si hvilke gjenstander som er emne for verket.

- **Hendelse**
  Entitetstypen «hendelse» tar for seg hendelser, epoker og tidsperioder som kan være emnet for et verk.

- **Sted**
  Sted henviser til stedsnavn som kan være emne for et verk. Dette kan være steder både på og utenfor jorda, nåværende eller historisk osv.

Relasjoner

Relasjonene i FRBR-modellen viser først og fremst hvordan de ulike entitetene henger sammen. Eksempler på dette er:

- «verk» er realisert ved «uttrykk»
- «uttrykk» er konkretisert ved «manifestasjon»
- «manifestasjon» er eksemplifisert ved «eksemplar»

Det er også slik at entitetene kan ha relasjoner til andre entiteter av samme type. Dette kan være en relasjon som for eksempel viser forholdet mellom to forskjellige verk. Et verk kan ha et etterfølgende verk, og et verk kan være en etterfølger av et verk, se figur 2.5 på side 34.

Attributter

Attributtene inneholder beskrivelsen om entiteten.

Vanlige attributter kan være tittel, navn, id-nummer osv., og disse har som oftest kun én verdi, men det kan også være flere verdier for hvert attributt. Det er også mulig at attributtene endrer verdi over tid.

Attributtene kan deles inn i to kategorier. Den første kategorien tar for seg attributter som er en integrert del av entiteten. Dette kan være karakteristikker for entiteten av fysisk art, og egenskaper som kan sees på som «etikettopplysninger». Disse attributtene bestemmes ved å undersøke informasjon om selve entiteten.

Den andre kategorien inneholder attributter som legges til entiteten. Dette kan være tilordning av identifikatorer, eller informasjon om konteksten rundt entiteten. Dette er informasjon som kun kan finnes ved bruk av eksterne kilder.

Entitetene i FRBR-modellen har mange attributter av begge de to kategoriene. Det er ikke nødvendig å fylle inn informasjon om hvert enkelt attributt, men man har muligheten hvis informasjonen er tilgjengelig.

Hele FRBR-modellen med tilhørende entiteter, relasjoner og attributter er vist figur 2.5 på side 34.

2.5 Identifikatorer

Metadata kan benyttes til å beskrive og identifisere forskjellige elementer om en ressurs. Men vi trenger også noe for å identifisere selve ressursen. Identifikatorer brukes for å kunne idenitfisere informasjonsobjekter, og kan defineres slik: «En identifikator er et entydig navn, kjennetegn eller merke som brukes for å identifisere noe» [2].

Navn benytter vi hele tiden i dagligtalen vår om blant annet personer og plasser. Dette er nødvendig for at vi skal vite akkurat hvilken person vi omtaler, eller hvilken spesielle plass vi skal møte noen på osv. På samme måte som vi trenger navn for å kommunisere med verden rundt oss og kunne organisere denne, så trenger vi identifikatorer for å kunne kommunisere og organisere informasjonsobjekter i digitale bibliotek og den digitale verden generelt [2].

Forskjellen når det gjelder navn og identifikatorer er at navn er forståelige, lesbare og «lett å huske» for mennesker. Identifikatorer er en form for navn, men skal være en unik verdi som identifiserer noe på en presis og utvetydig måte. Denne skal stort sett kun leses og forståes av maskiner, og er ofte nummer eller kombinasjoner av nummer og bokstaver, og trenger ikke å være «lesbar» for mennesker.

I bibliotekverdenen er det viktig å kunne identifisere litteratur og andre åndsverk, og identifikatorsystemene som er mest brukt her er ISBN (International Standard Book Numbering) og ISSN (International Standard Serial Number). Disse identifikatorene er bygget opp av nummersekvenser og er kun laget for å gi publikasjoner unike nummer, og er ikke forståelige for mennesker.

På World Wide Web har man en litt annen form for identifisering. Her bruker man adresser og lokatorer i form av URI, URN og URL. URI (Uniform Resource Identifier) er en overordnet term for alle de tre begrepene og sees på som en identifikator som refererer til noe på internett. URN (Uniform Resource Name), er en underkategori av URI og er mer spesifikk for å gi et varig og globalt unikt navn til en ressurs. URL (Uniform Resource Identifier) tar form som adresser og er identifikatorer som forklarer lokaliseringen til en ressurs [2].

Vi kan oppsummere med at lokatorer og adresser forteller noe om hvor ressurser befinner seg, men ikke hva som befinner seg der, mens navn og identifikatorer er kjennemerker for en ressurs uavhengig av lokalisering.

Eksempler på identifikatorer [19]:

ISBN

ISBN 0-201-88954-4

ISSN

ISSN 0001-0782

URL

http://www.idi.ntnu.no/index.html

URN

urn:nbn:no-12345678

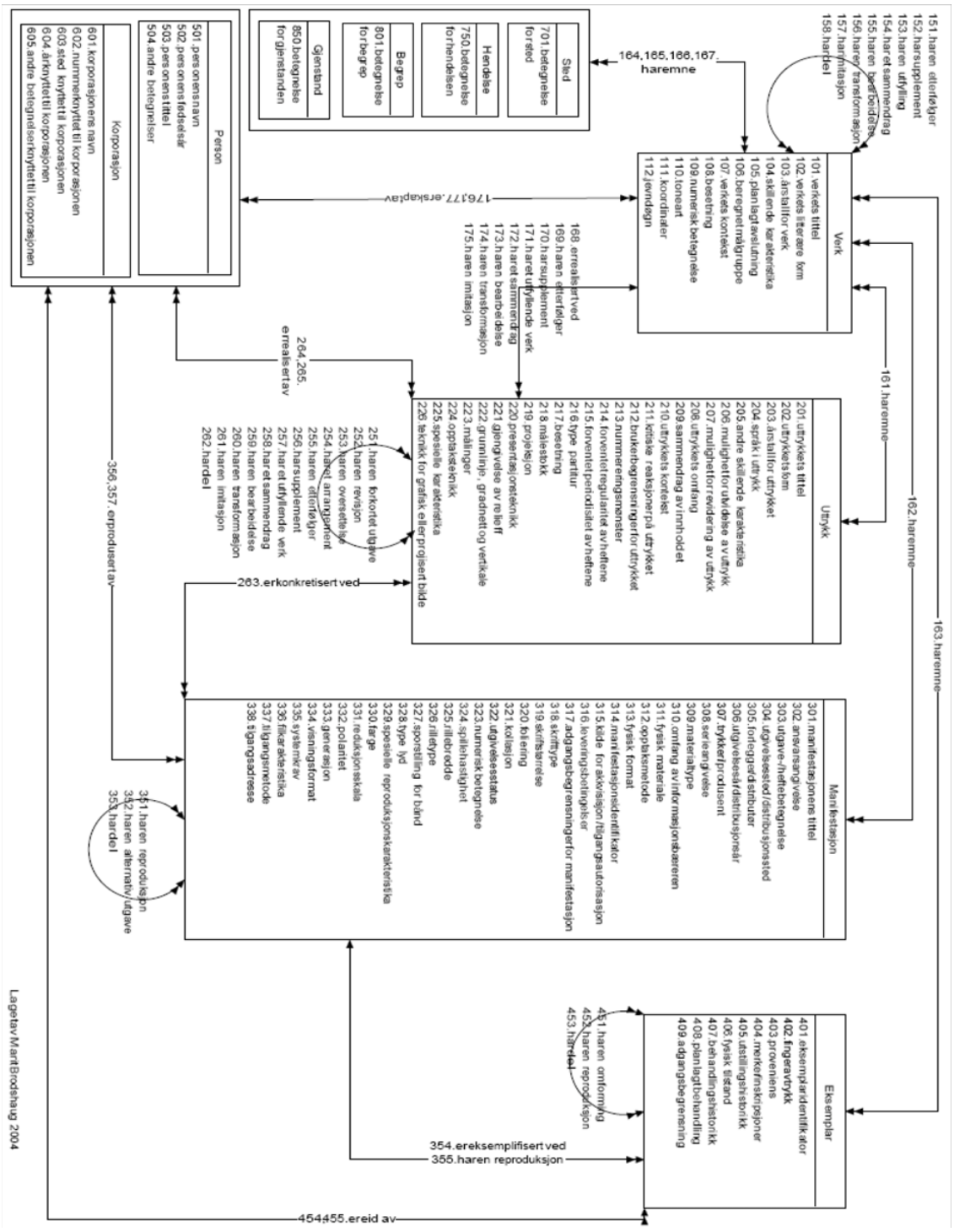
2.6 Identifikatorer for musikkdata

For musikkdata er det utviklet noen egne identifikatorer. Noen av disse er benyttet videre i oppgaven under implementasjonen av emnekart.

Det er utviklet fire ulike kodesystemer som skal støtte fire forskjellige typer av data[20]:

- International Standard Musical Work Code (ISWC), for verk
- International Standard Recording Code (ISRC), for lydopptak
- International Standard Music Number (ISMN), for noter
- International Standard Audiovisual Number (ISAN), for audiovisuelle verk

Dette er ISO-standarder som er internationalt anerkjent, og som det er fornuftig å benytte ved musikkdata. I denne oppgaven er det tatt i bruk de standardene som det har vært mulig og relevant å benytte.



Figur 2.5: FRBR-modell

Kapittel 3

Emnekart (Topic Maps)

Vi så i forrige kapittel på metadata, metadatastandarder og referansemodeller. Vi kan tenke oss emnekart som en referansemodell som kan bli benyttet til å ta i bruk teknikker fra digitale bibliotek for å hjelpe til med gjenfinning av informasjon. Vi nevnte bl.a. at emnekart er et rammeverk bygget på ontologier, men også andre teknikker nevnt tidligere kan bli implementert i emnekart. Ved å benytte emnekart til å representere metadata og emnebasert klassifikasjon, er det mulig å gjenbruke allerede eksisterende klassifiseringer og teknikker, samtidig som emnekart gir mulighet til å beskrive verden mer presist og dermed lette informasjonsgjenfinningen.

Emnekart er en vesentlig del av denne oppgaven siden det i senere kapitler vil bli sett på hvordan ulike metadataformater og modeller kan implementeres i emnekart. I dette kapittelet vil derfor emnekartstandarden bli forklart, og det vil bli sett på hvilke komponenter emnekart består av.

3.1 Bakgrunn

Emnekart, ISO13250, er en teknologi som skal gjøre det enklere å finne informasjon. Emnekart blir sammenliknet med «informasjonsuniversets GPS» [21], fordi man tenker seg at hvis man er ute etter å finne informasjon om et bestemt emne så kan man benytte emnekart som en slags GPS til å peile seg inn til hvor man kan få tak i denne informasjonen.

Man kan også på en forenklet måte sammenlikne emnekart med stikkordregisteret i en bok eller kartet over et land. Hvis man skal finne forklaring på et bestemt ord i en bok, er det naturlig å slå opp i stikkordregisteret hvor ordene er oppført med sidetall som beskriver hvor i boka man kan finne informasjon om det bestemte ordet. Likeledes er det om man skal finne et bestemt sted på et kart. Da slår man gjerne opp i registeret hvor det er beskrevet med koordinater hvor på kartet man finner det bestemte stedet. Disse registrene er med på å lette gjenfinningen av aktuell informasjon og sparer brukeren for tid.

Når det gjelder informasjon på Internett, så finnes det ikke registre på tilsvarende måte som beskrevet over. Internett er ikke avgrenset og strukturert i noen særlig grad, og til nå har det vært vanlig å benytte søkemotorer hvor brukeren skriver inn søkeord og får opp treff. Ofte får man opp altfor mange treff og det kan være vanskelig å sile ut hvilke treff som er relevante og som inneholder det man er ute etter.

Emnekart ble utviklet for å kunne organisere informasjon på Internett, og gjøre det mulig å navigere i informasjonsressursene og dermed også gjøre det lettere og gjenfinne informasjon.

For å få til en slik enklere informasjonsgjenfinning, laget man emnekartstandarden ut fra at man skulle kunne finne all informasjon om et gitt emne på én plass. Man får dermed samlokalisering av data.

Grunnen til at dette er hensiktsmessig er at personer som skal finne informasjon, som regel er ute etter et spesielt emne, og det er ressurskrevende å skulle søke etter informasjon mange plasser. Emnekartteknologien danner en slags kunnskapsvev på den måten at den ramser opp emner og viser hvordan ulike emner henger sammen ved hjelp av assosiasjoner. Emnekartet blir dermed et slags oversiktskart over alle emner, og hvordan emnene henger sammen, i tillegg til å henvise til den reelle informasjonen i form av et dokument eller bilder og lignende.

Eksempler på hvor emnekartstandarden er tatt i bruk, er emneportaler som fungerer som samlested for en gitt type informasjon, som www.forbrukerportalen.no og www.forskning.no. Disse har strukturert nettstedet på basis av en rekke emner som brukeren kan finne informasjon om, og emnene henger sammen ved hjelp av assosiasjoner. Ved hjelp av disse assosiasjonene har man også mulighet for å gjøre nettstedet dynamisk, slik at sidene endrer seg etter hvilke emne du er interessert i. Hvis du klikker på et bestemt emne, kommer det opp linker til emner som er relatert til ditt valgte emne.

Emnekart ble utgitt som en standard i 2000. Standarden var da en grunnmodell basert på SGML-syntaks, og ble kalt HyTM fordi HyTM-standarden ble brukt til lenking. Men det ble fort klart at standarden måtte bli mer rettet mot bruk på Internett. Det ble derfor opprettet en organisasjon kaldt «TopicMaps.org», som skulle lage en syntaks for emnekart basert på XML og URI-er.

XTM 1.0, var resultatet av dette arbeidet, og er i dag hovedstandarden som brukes for å representere og utveksle emnekart. XTM 1.0 er en XML-standard som er spesielt tilrettelagt for oppbyggingen av emnekart, og tar høyde for alle elementene som emnekart består av.

Emnekart er et nettverk bestående av emner (kapittel 3.2), assosiasjoner (kapittel 3.3 på neste side) og forekomster (kapittel 3.4 på side 39), og disse skal bli beskrevet videre i oppgaven. Det skal også sees på identifisering av emner.

3.2 Emne (topic)

Et emne er det et dokument handler om. Et dokument handler ofte om flere ting, og dokumentet har dermed flere emner.

For eksempel har et dokument emnet musikk, hvis det handler om musikk. Mer spesifikt tar dokumentet kanskje for seg jazz og Billie Holiday. Musikk, jazz og Billie Holiday, er dermed emner for dette dokumentet og er med på å fortelle hva dokumentet omhandler.

Problemet blir å skille ut de viktigste ordene i dokumentet som virker hensiktsmessige til å være emner. Disse skal kunne fortelle brukerne hva dokumentet handler om slik at de ser om det er av interesse. Det er ingen begrensninger på hva som kan være et emne, men det er ikke hensiktsmessig å trekke ut altfor mange ord, for da kan brukeren like gjerne lese hele dokumentet.

Emnekart hjelper til med navigeringen blant emner og viser hvordan emnene i emnekartet er relatert til hverandre. Slik emnekart er bygget opp kan man si at alt kan sees på som emner. Dette vil si at også forekomster og assosiasjoner er emner. Alle emner i emnekartet har form som en klikkbar link som fører brukeren videre inn i nettverket av emner. På denne måten kan vi si at emnekart er mer fokusert på hva ressursene handler om (hva emnet for ressursen er og hvordan den er relatert til andre), og ikke den konkrete ressursen.

Emnene representerer ulike temaer (subjects) som informasjonsressursene handler om. Et tema er noe abstrakt som kan gjøres mer konkret ved hjelp av et emne. Et emne er et element i emnekartet vist med <topic>-elementet, og dette representerer (handler om) et tema. Mellom emne og tema skal det være en én-til-én relasjon, dvs. at et tema bare skal representeres ved ett emne, og et emne kun skal representere ett tema. Emner kan også kategoriseres etter type, slik at de kan kategoriseres etter klassen de tilhører.

For å definere emnene i emnekartet slik at det skal kunne håndteres av systemet, benytter man <topic>-elementet. Figur 3.1 viser et emne med navngiving. Et navn har alltid et basisnavn som er basisformen av emnenavnet.I tillegg kan det ha flere variantnavn som kan brukes til fremvisning, sortering osv. Som syntaks benyttes som tidligere nevnt XTM, som er en XML-basert syntaks for emnekart, beregnet for standardisert datautveksling.

```
<topic id="emnenavn">
    <baseName>
        <baseNameString>Emnenavn</baseNameString>
    </baseName>
</topic>
```

Figur 3.1: XTM-syntaks for å opprette et emne

3.3 Assosiasjoner (Associations)

Assosiasjonene viser, som nevnt tidligere, hvordan emnene er relatert til hverandre. Det er de som lager emnekartet som bestemmer hvordan assosiasjonene skal være, men med utgangspunkt i hvordan emnene henger sammen i den virkelige verden, og hva som er naturlig. Man kan også bygge opp relasjonene på bakgrunn av tesaurusrelasjoner, databasestrukturer, eller hierarkiske strukturer [22].

Eksempler på hvordan emner kan kobles sammen ved hjelp av assosiasjoner:

«Don´t explain» er skapt av «Billie Holiday»

«Don´t explain» er realisert ved «lydopptak»

«Lydopptak» er konkretisert ved «CD-spor»

«CD-spor» er eksemplifisert ved «CD-sporet jeg har»

Eksemplene tok utgangspunkt i relasjoner fra FRBR-modellen omtalt i kapittel 2.4.3 på side 28.

Assosiasjonene i emnekart fungerer som lenker, men i motsetning til slik det som oftest fungerer på Internett, hvor linkene ligger i selve informasjonsressursen, er assosiasjonene i emnekartet uavhengig av selve ressursene. Assosiasjonene er i stedet lenker mellom emnene uavhengig av forekomstene som eksisterer til det aktuelle emnet [22].

3.3.1 Assosiasjonstyper

Vi ser fra eksempelet over at et emne kan ha flere forskjellige assosiasjoner knyttet til seg. «Don´t explain» kan både være skapt av noen, og være realisert ved noe. Men det er også slik

at assosiasjoner kan ha mange ulike emner knyttet til seg, siden det ofte er slik at flere emner benytter samme assosiasjon for å kobles sammen. I tillegg til «Don´t explain» så er også «I´m A Fool To Want You» og «For Heavens Sake» skapt av noen. De kan være skapt av noen andre enn Billie Holiday, men uansett så benytter de assosiasjonstypen «skapt av».

Hvis vi tar utgangspunkt i eksempelet over, så har det følgende assosiasjonstyper:

er skapt av

er realisert ved

er konkretisert ved

er eksemplifisert ved

Disse assosiasjonstypene må defineres som egne emner i xml-fila i emnekartet for at de kan taes i bruk. De viser da koblingen mellom emner i tillegg til å kunne gruppere sett av emner som har samme assosiasjoner, se kapittel 6.2.2 på side 62.

3.3.2 Assosiasjonsroller

Alle emner som deltar i en assosiasjon, spiller en rolle i denne assosiasjonen. Vi kan også si at emnene i assosiasjonen sees på som medlemmer av assosiasjonen og får tilegnet en rolle som de passer under. Man kan for eksempel si at Billie Holiday er en person, på lik linje med at Oscar Hammerstein og Sigmund Romberg er det. De er derfor medlemmer under rollen «person».

Eksempelet under viser hvordan forholdet er mellom assosiasjonstyper og assosiasjonsroller:

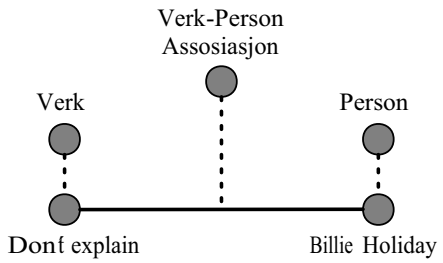
«verk» er skapt av «person»

«verk» er realisert ved «uttrykk»

«uttrykk» er konkretisert av «manifestasjon»

«manifestasjon» er eksemplifisert av «eksemplar»

Dette er også illustrert ved figur 3.2.



Figur 3.2: Assosiasjonseksempel

Assosiasjoner i emnekart skal ikke være rettede, de skal gå begge veier. Hvis A er relatert til B, må også B være relatert til A. Noen ganger er det slik at en assosiasjon er symmetrisk slik at den er lik begge veier, men dette er i mange tilfeller vanskelig å oppnå. Et eksempel på retting

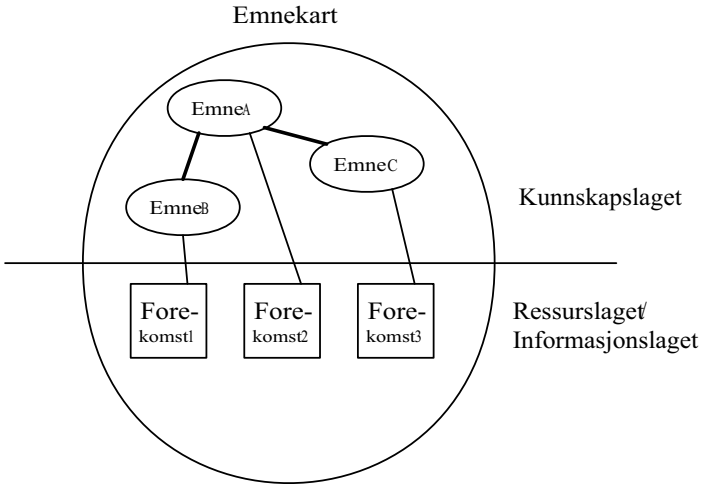
er at det er rett å si at verket er skapt av person. Men å gå andre veien å si at personen er skapt av verket blir feil. For å løse dette problemet benytter man assosiasjonsrollene for å vise hvilke roller de ulike medlemmene har i assosiasjonen.

3.4 Forekomster (Occurrences)

Forekomster er informasjonsressurser som er knyttet til et emne. Alt som behandler eller omtaler det temaet som representeres, er en forekomst [21]. En forekomst kan for eksempel være en artikkel, en omtale, en kommentar, deler av et dokument, en video osv. Som regel er forekomstene eksterne og blir representert i emnekartet ved URI-linker. Man kan for eksempel klikke på linken «Billie Holiday» (som er et emne i emnekartet), og vil da komme til en ekstern side med informasjon om henne.

Det er også mulig å ha forekomster internt i emnekartet. Dette kan være hensiktsmessig hvis man vil gi kommentarer, forklaringer osv. til emner i emnekartet.

3.5 Lagdeling i emnekart



Figur 3.3: Lagdeling i emnekart

Man kan tenke seg at et emnekart består av to lag. Øverst er kunnskapslaget som inneholder ontologien til emnekartet. Det er den som inneholder emnene, og assosiasjonen mellom dem. Det nederste laget er informasjonslaget, hvor man finner selve forekomsten eller dokumentet man er ute etter. Man kan se på kunnskapslaget som et register i en bok, mens informasjonslaget består av innholdet i selve boken. Koblingen mellom lagene skjer ved assosiasjoner i form av URI-er.

På grunn av denne todelingen som illustrert i figur 3.3, er det mulig å bruke det samme emnekartet for flere informasjonssamlinger. Emnene og assosiasjonene er ikke avhengige av noen spesifikke ressurser, og det er derfor mulig å koble emnekartet til flere ressurser. Dette kan gi brukerne mulighet til å få mer informasjon og gi flere synsvinkler på temaer.

3.6 Perspektiv (scope)

Perspektivet til et emne viser i hvilken sammenheng det hører under, og i hvilken sammenheng emnene er relatert til hverandre. Et emne kan ha forskjellig betydning avhengig av hvilket perspektiv det hører under. For eksempel så kan ordet «verk» ha forskjellig betydning ettersom det er tilknyttet FRBR-modellen og handler om åndsverk, eller om det er knyttet til temaet om helse og handler om tannverk osv. Vi sier da at dette er to forskjellige perspektiver. Emnene under de to perspektivene har ikke noe til felles, selv om de har like navn, og holdes adskilt som to forskjellige emner. Disse blir derfor ikke slått sammen ved sammenslåing av emnekart (som vi skal se nærmere på i kapittel 7 på side 71).

«Topic map authors must use scopes to distinguish between the different meanings of any name that is used for more than one subject [23].»

Når det blir brukt like navn på flere emner innen emnekart må altså de som lager emnekartet ta i bruk perspektiv får å holde emnene adskilt. På denne måten kan man benytte samme navnet på flere emner uten at de handler om samme tema. Man unngår dermed problemer med homonymi og polynomi.

I andre settinger det er aktuelt å benytte perspektiv er i forbindelse med forskjellige språk. Man kan for eksempel benytte perspektiv for å angi hvilket språk et lands navn skal vises i. Norge har forskjellige navn på forskjellige språk, og det er derfor aktuelt å angi at «Norga» er et gyldig navn for emnet Norge innenfor et samisk perspektiv.

- Norge - Scope: NO
- Norga - Scope: SV
- Norway - Scope: EN
- Norwegen - Scope: DE

Hvis man har et Norsk dokument, hvor perspektivet er NO, vil da det norske navnet «Norge» bli brukt. Men hvis perspektivet er satt til EN vil i stedet «Norway» bli benyttet.

3.7 Temaidentitet og temaindikatorer

Som nevnt tidligere er emnekart en hjelp for å finne den informasjonen man er ute etter på én plass. For å kunne opprettholde dette er det viktig at emnekartet er strukturert slik at det ikke finnes noen emner som er representert flere ganger. For å hjelpe til med dette, definerer man en subject identifier (SI) for hvert tema som er en unik identifisering av et tema. Dette gjør det lettere å holde oversikten over om det er flere emner under det samme temaet. Den unike identifiseringen kan være en adresse til hvordan man får tak i en ressurs, f.eks. en URL (Uniform Resource Locator).

Mange emner har ikke noen klar adresse for hvordan man kan få tak i ressursen. Disse kan kalles ikke-adresserbare. Når det gjelder disse kan man enten finne relevante URL-er, til hjemmesider o.l., eller så kan det være uten hensikt med identifikasjon, slik at SI kan droppes helt. Man kan i stedet URL benytte en temaindikator som beskriver ressursen.

Et moment når det gjelder SI i emnekart er at det helst bør brukes en universell kode. Hvis dette blir overholdt er det mulig å slå sammen flere emnekart med liknende innhold. Hvis to

emnekart har samme SI på sine emner, blir disse slått sammen til et emne for å opprettholde samlokaliseringsprinsippet. For å få til dette benytter man Published subject identifier som skal bli sett nærmere på i kapittel 7.1.2 på side 74.

3.8 Emnekart og Metadata

Emnekart ser på verden som en oppbygging av emner og assosiasjonene for hvordan disse emnene henger sammen. Hvordan strukturen er for emnene og assosiasjonene, blir definert som en modell i bunnen for hvert emnekart ved hjelp av ontologier. På denne måten får emnekart en fleksibel men samtidig presis datamodellering. Fleksibel fordi man kan lage strukturen i emnekartet akkurat sånn man ønsker, og som det er hensiktsmessig for emnene som skal representeres. Presis fordi man bygger en fast ontologi som emnekartet baserer seg på.

Metadataformater har i motsetning til emnekart en mer bestemt struktur. Hvis man tar i bruk en metadatastandard har man lite muligheter for å endre denne, og strukturen blir dermed veldig fast. Dette kan være fordelaktig ved at man vet hva man har å forholde seg til og strukturen blir dermed lik for alle som benytter formatet, også utenfor landegrensene. En ulempe er at det blir vanskeligere å tilpasse seg en spesifikk samling av ressurser. Formatet er kanskje laget med tanke på f.eks. bøker og andre skriftlige uttrykk, og passer dermed kanskje ikke like godt for f.eks. musikkdata. En annen ulempe er at det for enkelte metadataformater kan være vanskelig for vanlige brukere som ikke er opplært i formatet å benytte det. Formatene kan ofte være komplekse og mer myntet på å strukturere og klassifisere innhold og ikke så mye på gjenfinning for en uerfaren bruker.

Emnekart har lagt vekt på å være brukervennlig og kan bygges opp med emner og assosiasjoner slik at mennesket kjenner sammenhengene igjen fra verden, uansett hva temaene omhandler, eller hvilke formater ressursene har. En ulempe er at emnekartene blir forskjellige ettersom hvem som lager disse, og navn på emner og assosiasjoner kan variere fra emnekart til emnekart selv om det omhandler de samme temaene. Ved sammenfletting av emnekart kan dette være problematisk fordi strukturen er forskjellig i emnekartene man fletter sammen. Ved sammenfletting slår man sammen emner som er like, dvs. har like navn eller lik identifikator (se kapittel 7 på side 71). Men hvis emnekartene benytter forskjellige navn og identifikatorer så vil ikke emnene bli slått sammen. Det finnes publiserte identifikatorer som skal hjelpe til å løse dette (se 7.1.2 på side 74), men det skal i denne oppgaven også sees på om det kan være til hjelp å basere emnekartene på metadataformater for å gjøre strukturene i emnekartene likere og dermed gjøre det lettere å opprettholde strukturen i emnekartene ved sammenslåing.

De neste kapitlene skal teste ut hvordan ulike metadataformater/modeller egner seg for å bli implementert i emnekart. Vil man kunne få fordelene fra begge hold? Vil man kunne nyttiggjøre seg av den faste og gode innarbeidede strukturen som finnes i forbindelse med metadataformater og i tillegg ha muligheten til å visualisere det på Internett og gjøre det brukervennlig ved hjelp av emnekart?

Kapittel 4

Dublin Core og emnekart

4.1 Bakgrunn

I dette kapittelet skal det testes hvordan det vil fungere å benytte Dublin Core-elementer i emnekartstandarden. Dublin Core er et metadataformat hvor elementene skal forklare en ressurs, og som ikke inneholder selve ressursen i seg selv. Emnekart inneholder heller ikke i utgangspunktet selve ressursene, men er mer opptatt av muligheten til å navigering blant emner og på denne måten finne ressursene man er på jakt etter, mens Dublin Core baserer seg på å beskrive metadata om en ressurs og dermed gjøre det enklere å kunne søke på dokumentet.

4.2 Forskjellig bruk av Dublin Core i emnekart

Når det gjelder bruk av Dublin Core i emnekart, så er det ulike måter og gjøre dette på. En måte er å implementere Dublin Core som beskrivende metadata for hele emnekartet. Dette fordi det er av interesse å se hvem som har laget selve emnekartet, rettighetene til det, dato for når det er laget osv. Selve emnekartet blir da sett på som en ressurs, og det er naturlig å knytte informasjon til ressursen på samme måte som det er for ressursene inni emnekartet, for å lette forståelsen av innholdet. Metadataene til emnekartet blir derfor implementert som forekomster eller som assosiasjoner. Forekomstene blir brukt for å gi statisk informasjon om emnekartet, og er aktuelt for feltene om rettigheter, dato og beskrivelse. Dette fordi disse feltene kun er en opplysning som brukes som forklaring på dette bestemte emnekartet. De andre DC-feltene er aktuelt å implementere som assosiasjoner. Disse feltene kan ha flere ressurser koblet til seg og det er aktuelt å benytte assosiasjoner for å se sammenhengen mellom ressursene.

I tillegg til å benytte DC som metadata til selve emnekartet, kan det også være aktuelt å bygge hele emnekart basert på Dublin Core-standarden. Man tenker seg da at elementene i Dublin Core er emner i emnekartet. Det kan spesielt være hensiktsmessig å benytte en standard hvis det er aktuelt å skulle slå sammen (merge) flere emnekart. Hvis emnene i de ulike emnekartene er like og følger samme oppbygging, og relasjonene er like mellom emnene, så er det mye større mulighet for å få en bra struktur også etter å ha slått sammen flere emnekart.

Hvis man allerede benytter Dublin Core som metadatastandard på ressursene sine, kan det være av interesse og kunne benytte Dublin Core i andre sammenhenger som i f.eks emnekart. Man har da allerede brukt tid og innsats på å sette seg inn i en standard og det er da en fordel å kunne bruke denne videre.

Dette kapittelet skal se nærmere på hvordan det i praksis vil bli å basere hele emnekartet på

Dublin Core. DC fungerer som et skall med informasjon om en ressurs, og det kan man si om emnekart også. Emnekartet er et kunnskapslager som linker til selve ressursen gjennom en assosiasjon. En forskjell når det gjelder DC og emnekart er at det ikke er noen klare assosiasjoner mellom elementene, og heller ikke mellom elementene og selve ressursene. Vi kan si at Dublin Core er settorientert i motsetning til emnekart som er relasjonsorientert. Dette skal diskuteres nøyere.

4.3 Relasjonsorientert vs settorientert

Emnekartstandarden er i utgangspunktet basert på at det skal finnes relasjoner mellom emner, og standarden er derfor sterkt relasjonsorientert. Dette gjør det enkelt for brukeren å navigere i et nettverk av emner, fordi det tydelig fremgår hvilke emner som henger sammen og på hvilken måte.

Dublin Core er ikke internt basert på en relasjonsorientert tankegang. DC baserer seg på et sett av elementer, og det er ikke fastsatt noen relasjoner mellom disse elementene. Vi kan her tenke oss en settorientering, hvor et sett av elementer med innhold er assosiert til et spesifikt dokument. Hvert enkelt dokument har et slikt sett med elementer knyttet til seg. Det er derfor ikke umiddelbar bruk for implisitte sett-tilhørigheter. Det kan imidlertid være aktuelt med en relasjon mellom de ulike dokumentene for å vise f.eks. at en forfatter har skrevet flere ulike dokumenter. Dette fremgår ikke direkte fra DC, men DC gjør det lettere å søke f.eks. på en spesiell forfatter, og da få opp en liste med alle dokumenter han har skrevet.

Det finnes imidlertid DC-felter som gjør det mulig å vise relasjoner ut av settet, og til andre sett. Feltet «Relasjon» kan benyttes til å vise relasjoner mellom frittstående ressurser som har noe med hverandre å gjøre. Dette kan for eksempel benyttes til å si at en bok er en del av en større boksamling, eller at et enkeltbilde er del av en bildesamling osv.

Et annet felt som kan vise tilknytning mellom ressurser er «Source». Det kan være at en ressurs egentlig stammer fra en annen ressurs, og dette feltet viser denne sammenhengen ved hjelp av en identifikator. Hvis man f.eks. har en PDF-versjon av en bok, så er det aktuelt å skrive opp ISBN-nummeret til den trykte boka som PDF-versjonen stammer fra, i Source-feltet.

4.4 Implementasjon av Dublin Core i emnekart

Ved implementasjonen av Dublin Core i emnekartet var det nødvendig å lage egendefinerte relasjoner, siden relasjoner er påkrevd for at det skal være et emnekart. Alle de 15 elementene ble implementert som separate emner, men på grunn av settorienteringen var det umulig å lage en typisk nettverksstruktur. Det naturlige ble å ha et element som alle de andre elementene relaterte til, slik at det fremdeles fremsto som et sett av elementer til et bestemt dokument. Ulike løsninger av et slikt samlingselement er blitt sett på.

4.4.1 Første løsningsalternativ

Det elementet som umiddelbart skilte seg ut til å være et samlingselement var «identifikator», siden dette elementet identifiserer hvilket dokument det er snakk om. Denne løsningen ble raskt forkastet da det viste seg å være svært lite brukervennlig.

Emnekart er en standard som er basert på visualisering og brukervennlighet. Ved å benytte

«omnigator» som visualiseringsverktøy for implementeringen, viste det seg at «identifikator» egnet seg lite som samlingselement. Identifikatoren er gjerne en kode som er lite lesbar for mennesket, og når denne da blir benyttet som samlingselement vil koden bli vist som emne. Brukeren vil da ikke skjønne hvilket element det er snakk om før han har trykket på linken og letet seg frem til mer informasjon om emnet. Dette kan løses ved bruk av flere navn på identifikatoren, slik at også f.eks. tittelnavnet blir vist i tillegg til koden, eller at man legger inn occurrences. Men jeg har i denne løsningen prøvd å følge standarden i Dublin Core. Jeg valgte derfor å implementere enn annen løsning med «tittel» som samlingselement.

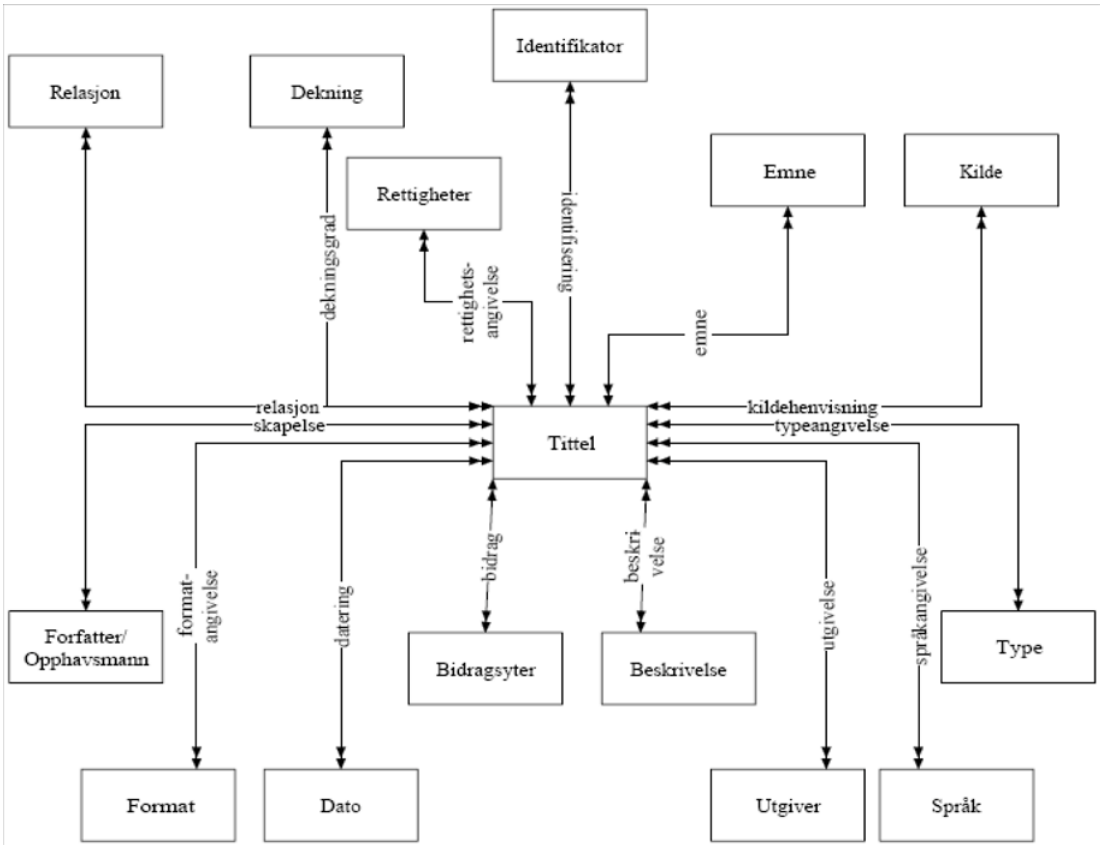
4.4.2 Andre løsningsalternativ

Det ble her valgt å bruke «tittel» som samlingselement, se figur 4.1 på neste side. «Tittel» og «Identifikator» er nært knyttet opp mot hverandre siden identifikatoren er koden, mens tittelen er en tekst som er forståelig for brukeren, men som likevel identifiserer dokumentet selv om det ikke nødvendigvis er en unik tittel. I emnekartet er identifikatoren benyttet som «topic id» for tittel, slik at settet med DC elementer blir unikt for riktig dokument.

Denne løsningen var relativt oversiktlig for brukeren, siden tittelen er lett gjenkjennelig. Brukeren leter kanskje etter en spesiell sang eller bok som han kan tittelen på, eventuelt så sier tittelen noe om hva man kan vente seg hvis han ikke vet noe om tittelen fra før. Det er også noe som er lettere å huske enn en kode.

Assosiasjonene som ble benyttet i dette løsningsalternativet var egendefinerte, og hadde ikke noen tilknytning til Dublin Core standarden, siden denne ikke har støtte for assosiasjoner. Assosiasjonene ble laget ut fra navnet på elementene:

- tittel - identifisering - Identifikator
- tittel - skapelse - Forfatter
- tittel - datering - Beskrivelse
- tittel - utgivelse - Utgiver
- tittel - formatangivelse - Annen bidragsyter
- tittel - datering - Dato
- tittel - emne - Emne
- tittel - typeangivelse - Type
- tittel - formatangivelse - Format
- tittel - kildehenvisning - Kilde
- tittel - språkangivelse - Språk
- tittel - relasjon - Relasjon
- tittel - dekningsgrad - Dekning
- tittel - rettighetsangivelse - Rettigheter



Figur 4.1: Tittel som samlingselement for DC-elementene og med egendefinerte relasjoner

Disse assosiasjonene er relativt intuitive for brukeren, og forteller hva assosiasjon handler om slik at brukeren forstår det. Ulempen er at ordene er egendefinerte og har lite å gjøre med Dublin Core-standarden.

Begge de to første løsningene er i følge Dublin Core umulig å benytte, siden standarden sier at man det ikke er nødvendig å fylle inn informasjon om alle de 15 elementene. Det kan virke rart å ikke skulle ha en tittel eller noe som identifiserer et dokument, men det er i følge standarden lov å droppe disse elementene. Uten å bestemme eksplisitt at «tittel» eller «identifikator» må fylles ut, så kan man ikke benytte disse som samlingselement. De andre elementene vil da ikke ha noe element og assosiere til, og vil ikke bli vist.

4.4.3 Tredje løsningsalternativ

I det tredje løsningsalternativet er det laget et eget samlingselement, som ikke tilhører Dublin Core standarden. Dette elementet er blitt kaldt «DC.record», og skal være et bindeledd for alle de 15 elementene til DC som kan ha informasjon om det gitte dokumentet. Dette elementet må alltid ha innhold, mens alle de andre elementene kan utelates slik det er forespeilet i standarden.

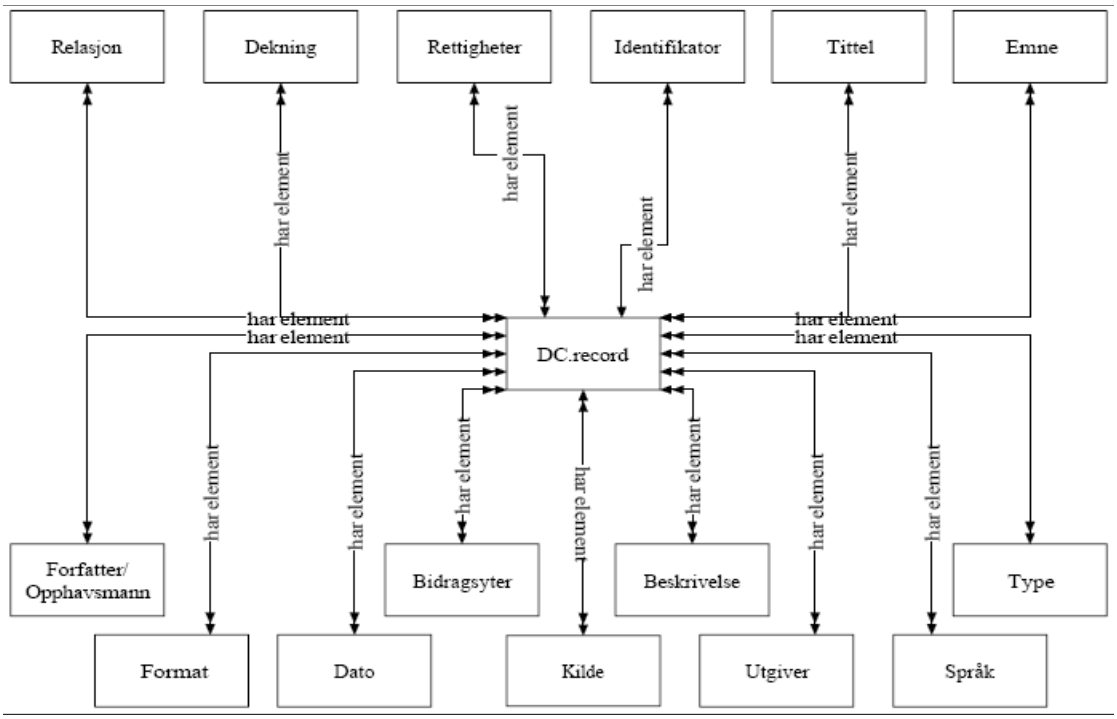
Problemet med et slikt oppdiktet element er å si hva som bør være innholdet. Alle elementer i emnekartet må ha en emneidentifikator, og siden «DC.record» ikke har noe naturlig innhold,

så blir emneidentifikatoren innholdet. Dette kan da være en vilkårlig kode, eller en slags sammensatt kode av «identifikatoren» og noe annet. Ulempen med denne løsningen er at den likner litt på det første løsningsalternativet hvor brukeren kun ser en kode overalt og ikke skjønner hvilket dokument det handler om. Man må derfor ta i bruk ekstra forklarende navn.

Som assosiasjonsnavn er det i denne løsningen blitt sett mer på hvordan standarden til DC er utformet. DC er jo bygget opp av elementer, slik at når vi tar i bruk samlingselementet «DC.record» så kan man si at dette «har element» - «elementnavn». Se figur 4.2.

Eksempel:

- DC.record - har element - identifikator
- DC.record - har element - tittel
- DC.record - har element - format
- osv.



Figur 4.2: DC.Record som samlingselement for DC-elementene

Emnetypene, assosiasjonstypene og assosiasjonsrolletypene som ble implementert i dette løsningsalternativet er vist i tillegg B på side 95.

4.4.4 Fjerde løsningsalternativ

I de tidligere løsningsalternativene har alle de 15 elementene blitt tatt med som separate emner. Dette har vært for å følge DC-standarden mest mulig, men det er ikke nødvendigvis det mest brukervennlige. Flere av DC-elementene egner seg ikke spesielt bra som separate emner i et emnekart. Dette gjelder i hovedsak «beskrivelse» og «emne». Dette er elementer som er

forskjellige fra dokument til dokument, og det er derfor ikke naturlig å ha dette som emner som andre emner kan linke til. Andre elementer det kan være unaturlig å ha som egne emner er «Relasjon» og «Source», siden disse viser til assosiasjoner, og dermed ikke er typiske emner. Disse DC-feltene kan ikke utelates fra emnekartet, men kan bli implementert som forekomster til emner i i stedet for å være emner i seg selv.

4.5 Implementasjon av kvalifikatorer

4.5.1 Kvalifikatorer

Det ble tidligere nevnt at det blir benyttet kvalifikatorer i Dublin Core for å ha mulighet til å legge til tilleggsinformasjon til elementene. Dette kan i DC gjøres på to forskjellige måter. Enten så kan man utvide med et tilleggsnavn bak et punktum i elementnavnet (refinement). Man presiserer da betydningen av hva som hører til elementet. Eller man kan kreve at innholdet i elementet skal ha en bestemt syntaks, eller følge et kontrollert vokabular. Man benytter seg da av skjemaer (scheme), eller systemer for å ha en strukturert og komplett oversikt over hva som kan være innholdet.

4.5.2 Hvordan støttes kvalifikatorer i emnekart?

Det er ikke tenkt på kvalifikatorer for emnekartstandarden, men det er ulike måter å løse bruk av kvalifikatorer på. Skjemaer og retningslinjer kan følges for hva som skal inngå i de ulike emnene. Disse skjemaene vil da være eksterne.

En annen løsning er å ha forklaringen på hva som skal inngå i emnene som forekomst. Forekomsten kan da være en forklarende post, slik at brukeren lett kan se hva innholdet skal være når han er inne på et emne. Informasjonen blir da intern i emnekartet, noe som letter informasjonsinnhentingen for brukeren.

Det er også mulig å endre de opprinnelige elementnavnene og tillegge et ekstra navn bak det opprinnelige i emnenavnet, og på den måten følge standarden for refinement i DC.

Ytterligere en mulighet er å benytte seg av scope. Man tenker seg da at man kan se elementet fra forskjellige perspektiver og elementet kan da vise forskjellig innhold ettersom hvilket perspektiv man er ute etter.

4.6 Evaluering

Dublin Core er en enkel standard som hjelper til med å strukturere informasjon om nettdokumenter, og tar form som et sett av metadata som knyttes til et spesielt dokument. Den tar ikke høyde for relasjoner mellom elementene, og heller ikke i stor grad på tvers av dokumenter. Siden emnekart er svært relasjonsorientert, ved at hele standarden baserer seg på å lage assosiasjoner mellom emner, blir det vanskelig å kombinere disse standardene.

For å kunne ta i bruk DC i emnekart må man definere egne relasjoner, som ikke inngår i DC-standarden. Man må også finne en løsning på hvordan man kan få alle DC-elementene, som er blitt gjort om til emner, til å danne en helhet for dokumentet de til sammen representerer, i form av et samlingselement eller forekomster. Noen forskjellige løsninger er vist i dette kapittelet.

Får å basere emnekartet fullt og helt på Dublin Core, må det gjøres veldig mange tilpasninger og endringer bort fra DC-standarden, og det blir derfor naturlig å tenke at det ikke egner seg å gjøre det på denne måten. Man kan i stedet lage et emnekart bygget på emner som mer naturlig følger emnekartstandarden, og som baserer seg på en mer relasjonsorientert tenkemåte.

Kapittel 5

Dewey og emnekart

5.1 Bakgrunn

Det vil i dette kapittelet bli testet hvordan Deweys klassifikasjonssystem kan implementeres i emnekartstandarden.

Dewey er hierarkisk oppbygget og baserer seg på tallkoder med tilhørende navn. Denne inndelingen er med på å klassifisere ressurser slik at det er mulig å finne ressurser som omhandler samme tema på samme plass. Dette likner i store trekk på hva emnekartstandarden går ut på, som også strukturerer ressurser ettersom hvilket emne de tilhører for at de skal kunne finnes på én plass. Men Dewey har i utgangspunktet blitt utviklet for trykte medier plassert i hyller på et bibliotek, mens emnekartstandarden er laget for internett. Det er derfor interessant å se om det er hensiktsmessig å ta i bruk Dewey på internett ved hjelp av emnekartstandarden.

5.1.1 Emner

Et grunnleggende prinsipp i Dewey er at dokumenter ordnes etter fag og ikke etter emne. Dette er fordi et emne som regel ikke er plassert kun én plass i systemet. I tillegg til å finne emner om musikk under 700-tallet, kan man se musikk fra sosiologisk synsvinkel under 306.484, musikk i forbindelse med grunnskolen under 372.87 og musikk i forhold til kristendom under 246.7.

Dewey blir benyttet for å kunne sette bøker i biblioteker i system etter innhold. På denne måten er det lettere for brukeren å finne kunnskap om et spesielt fag/emne han er ute etter. Han kan da peile ut den hylla hvor innholdet han er ute etter står, og se om han finner noe interessant under det bestemte faget/emnet.

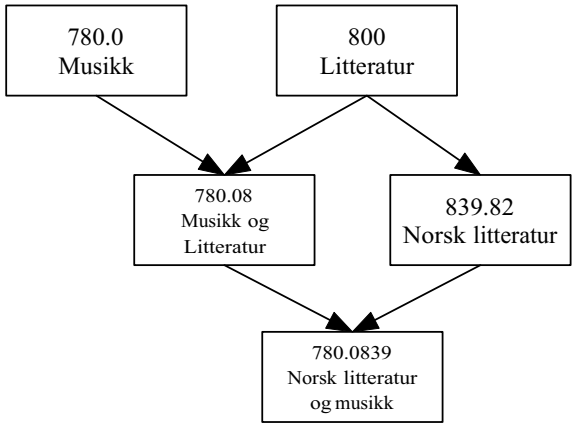
Vi kan her trekke paralleller til hvordan emnekart fungerer. Emnekart baserer seg på datateknologi og omhandler ofte digitale ressurser, i motsetning til Dewey som i hovedsak er myntet på fysiske bøker i et bibliotek. Men begge systemene går ut på å sortere kunnskap etter hvilket emne/fag ressursene handler om.

I Dewey er emnene/fagene fast bestemt og det er et system som må følges. Emnekart har i utgangspunktet ingen slike faste emner. Emnekartstandarden sier at alt kan være emne, og det er opp til den som bygger emnekartet hvilke emner som skal inngå. Det er derfor interessant å tenke seg at man kan bestemme seg for å lage et emnekart basert på de gitte emnene i Dewey.

5.1.2 Assosiasjoner

Relasjon mellom musikk og andre emner i henhold til Deweystandarden

Dewey er hierarkisk oppbygget, og relasjonene representerer i hovedsak stigende og synkende rang i hierarkiet. Men det er også mulig å lage relasjoner på tvers av dette oppsettet. Her er et eksempel på dette innen musikkens verden: Basisnummeret for musikk er som tidligere vist 780.0. Men man kan til dette tallet legge til numrene 001-999, men ikke flere enn tre sifre. Vi kan for eksempel se hvordan det blir å koble sammen musikk og litteratur. Litteratur har basisnavn 800, og nummeret blir derfor 780.08. Vi kan i tillegg tenke oss en klassifisering av musikk og norsk litteratur. Norsk litteratur finner vi under tallet 839.82 og de to emnene sammen får tallet 780.0839, siden det ikke er lov å legge til mer enn tre siffer. Dette er illustrert i figur 5.1



Figur 5.1: Assosiasjoner i Dewey

5.1.3 Forekomster

Noter

I Dewey er det mulig å legge til noter (forklaringer) til tallene. Dette hjelper til med å gi en mer utfyllende informasjon som ikke klassenummeret eller klassebetegnelsen viser av seg selv. Notene kan brukes til å:

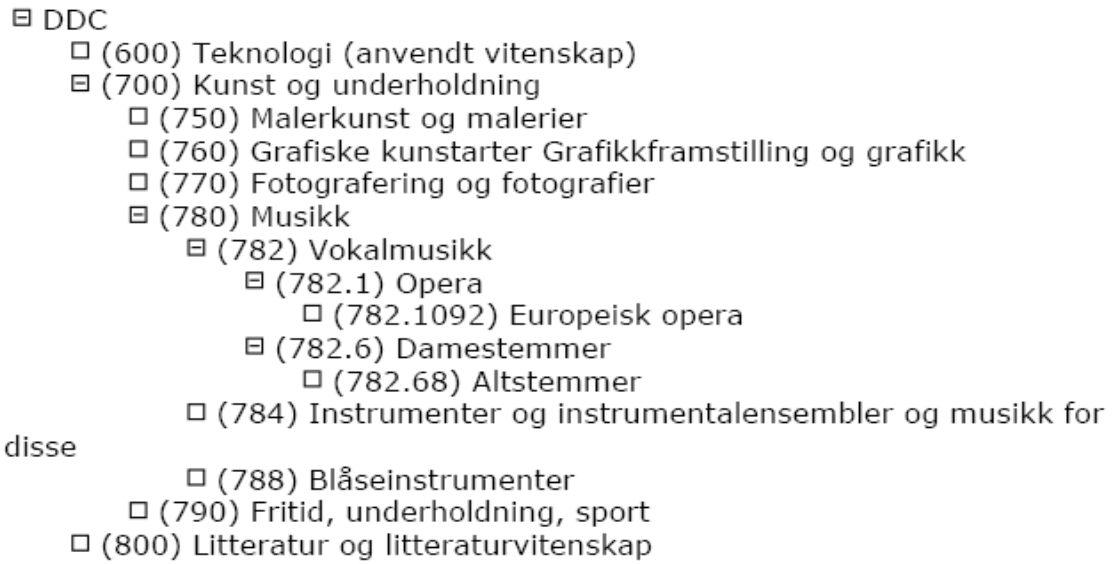
A forklare hva som hører inn under klassen og dens underinndelinger

B liste opp emner som ikke har nok litteraturbelegg til å ha eget nummer

C beskrive hva som hører inn under andre klasser

D forklare endringer i tabellene

Under A kan det finnes «her-noter» som viser viktige emner som inngår i klassen. Slike noter fungerer som forklaringer til klassene og det er derfor naturlig å implementere disse som interneforekomster i emnekart.



Figur 5.2: Hierarkisk visning i omnigator

5.2 Implementasjon

Ved implementasjonen tok jeg utgangspunkt i subsettet av Deweyemner som jeg plukket ut fra standarden og illustrerte i figur 2.2 på side 23. Emnene jeg valgte å implementere var derfor som vist i figur 5.2, og emnetypene representeres ved Deweytallene kombinert ved emnenavnene.

```
<association>
    <instanceOf><topicRef xlink:href="#superclass-subclass"/></instanceOf>
    <member>
        <roleSpec><topicRef xlink:href="#superclass"/></roleSpec>
        <topicRef xlink:href="#dewey782.1"/>
    </member>
    <member>
       <roleSpec><topicRef xlink:href="#subclass"/></roleSpec>
       <topicRef xlink:href="#dewey782.1092"/>
    </member>
</association>
```

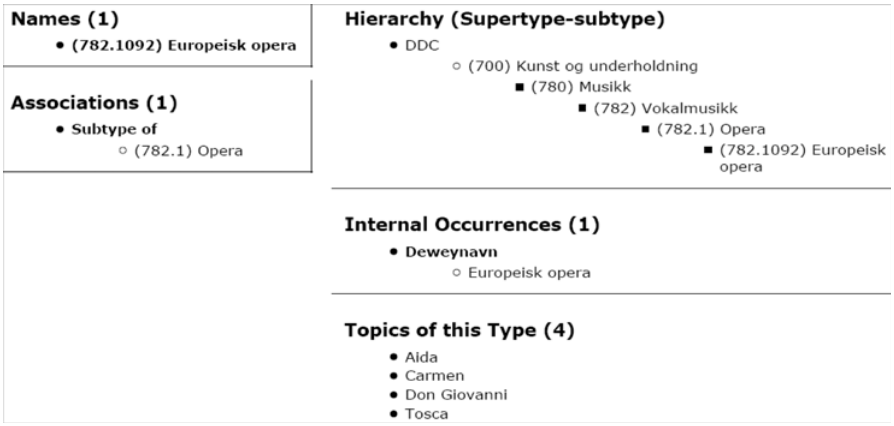
Figur 5.3: Kodeeksempel på superklasser og subklasser i emnekart

Assosiasjonene som er implementert er også de samme som vist som streker i figur 2.2 på side 23. Det er derfor et rent hierarkisk emnekart, uten kryssreferanser av noe slag. Emnekartstandarden er i utgangspunktet nettverkspreget, hvor emner er linket til hverandre på kryss og tvers. Men i Omnigator er det allikevel laget et eget oppsett for å implementere hierarkier bestående av superklasser og subklasser. Denne muligheten er i utgangspunktet laget som et supplement til nettverksstrukturen som emnekart vanligvis består av, men siden Dewey i utgangspunktet er hierarkisk, virket det naturlig å benytte denne funksjonaliteten.

I kodeeksempel 5.3, ser man et utdrag fra XML-koden som viser hvordan assosiasjonen mellom

en superklasse og subklasse blir kodet. Vi ser her at «dewey782.1092» som er topic-navnet for et emne, er subklassen av en superklasse med navn «dewey782.1». I omnigator vises dette som i figur 5.4. For å få opp dette skjermbildet er det trykket på linkene nedover i hierarkiet på 700-tallet. Til venstre i skjermbildet ser man at navnet på emnet linker til 782.1092, og at dette emnet er subtypen til 782.1. Til høyre i skjermbildet ser man alle emnene som er over i hierarkiet.

Øverst i hierarkiet er det plassert en emnetype med navn «DDC». Dette elementet er ikke direkte med i Deweystandarden, men jeg har valgt å opprette dette for å få en toppnode i hierarkiet. «DDC» inneholder ingen emner, og består kun for at de øverste tallene i hierarkiet skal ha et samlingselement, slik at de henger sammen, og det dermed dannes et hierarki. Uten dette elementet måtte man ha valgt et av de øverste tallene til samlingselement, noe som blir feil fordi man ikke kan si at 700 er over 600 osv. I Deweystandarden er begge disse tallene hovedklasser og på likt nivå i hierarkiet.



Figur 5.4: Deweyemner

Emnetypene, assosiasjonstypene, assosiasjonsrolletypene og forekomsttypene som er implementert i emnekartet basert på Dewey, er vist i tillegg C på side 97.

5.3 Evaluering

Dewey er et klassifikasjonssystem som tar utgangspunkt i at ressurser struktureres etter hvilket fag de omhandler. Fagene er bestemt i standarden og strukturert hierarkisk etter kode og navn. Disse fagene kan lett implementeres som emner i emnekartstandarden.

I implementasjonen av emnene ble først kun tallkodene til Dewey benyttet i basisnavnet til emnekartet. Dette var forklarende for å se hvordan tallene var hierarkisk strukturert, men var lite brukervennlig fordi de færreste mennesker har Deweytallene i hodet. Det ble derfor i stedet implementert en kombinasjon av tallene og de tilhørende navnene. På denne måten kom fremdeles tallene tydelig frem i emnekartet, men det ble i tillegg vist forståelige navn for mennesket.

Emnekart strukturerer emnene i nettverk, mens Dewey strukturerer fagene hierarkisk, som er en form for nettverksstruktur. Emnekart har mulighet for implementasjon av hierarkier, og nettverksstrukturen i Dewey kan bestå uendret ved en implementasjon.

Som interne forekomster ble «notes» implementert, og det er her mulighet for å legge inn små merknader. De eksterne forekomstene er link til den konkrete ressursen.

Totalt kan man si at Dewey-standarden på en relativt enkel måte kunne implementeres i et emnekart uten at strukturen ble forandret. Men standarden får ikke utnyttet mulighetene emnekartet har for nettverksstruktur optimalt, siden Dewey kun er hierarkisk.

Kapittel 6

FRBR-modellen og emnekart

6.1 Bakgrunn

FRBR-modellen er et rammeverk for bibliografiske elementer, og et grunnlag for datasystemer som kan la brukerne navigere i bibliografiske poster på en enkel og strukturert måte (se kapittel 2.4.3 på side 28). Dette er ikke så ulikt hva vi kan si om emnekart, som også er utviklet for at brukeren skal kunne navigere å gjenfinne informasjon på en enkel måte. Men emnekart legger større vekt på at brukerne skal kunne navigere seg frem til informasjonsressurser på internett (se kapittel 3 på side 35), mens FRBR-modellen i større grad er en underliggende modell for strukturering av informasjonsressurser. Det er derfor interessant å se om det kan ha en hensikt å benytte FRBR-modellen som en struktur som emnekartet kan baseres på slik at man følger en standard for organiseringen av informasjonsressursene, samtidig som man kan benytte seg av den brukervennlige navigeringsløsningen til emnekartet.

I kapittel 4 på side 43 og kapittel 5 på side 51 er det sett på andre standarder i forbindelse med emnekart, men disse har hatt en relativt enkel oppbygging. FRBR-modellen er en kompleks modell, og det er uvisst hvordan emnekart takler slike komplekse modeller. Det er derfor relevant å undersøke hvordan et emnekart oppbygget på FRBR-modellen vil fungere i praksis.

6.2 Oppbygging av emnekartet basert på FRBR-modellen

For å teste og vise hvordan emnekart basert på FRBR-modellen vil fungere i praksis har jeg implementert dette, og utførelsen vil bli beskrevet videre i kapittelet. Det er tatt utgangspunkt i en samling av musikk-metadata (informasjon om innspillinger/sanger med Billie Holiday), og informasjonsressursene om denne samlingen er strukturert etter FRBR-modellen. Det er både forklart den tekniske implementasjonen av Billie Holiday-data i FRBR-modellen i tillegg til hvordan dette er implementert som XTM-kode/emnekart og vist i visualiseringsverktøyet Omnigator.

FRBR-modellen består av entiteter, attributter og relasjoner. Når disse komponentene skal overføres og sammenfattes i henhold til komponentene i emnekartstandarden, er det nærliggende å tenke seg at entitetene blir emner, relasjonene blir assosiasjoner og attributtene blir interne forekomster.

6.2.1 Emner

Entitetene i FRBR-modellen er de komponentene som er av kjerneinteresse for brukerne av bibliografiske data (se kapittel 2.4.3 på side 28). På samme måte kan vi si at emnetypene i emnekartet skal fremheve det som er av kjerneinteresse for brukerne av emnekartet, nemlig emnene. Ved å implementere entitetene fra FRBR-modellen som emnetyper, kan man dermed benytte emnekartet som en hjelp til å navigere blant komponentene som brukerne er mest interessert i.

Emnene i emnekartet blir dermed det samme som entitetene i FRBR-modellen:

- verk
- uttrykk
- manifestasjon
- eksemplar
- person
- korporasjon
- begrep
- objekt
- hendelse
- sted

Selve teorien om hva disse entitetene representerer står forklart i kapittel 2.4.3 på side 28, men hva som dukker opp av problemstillinger ved en implementasjon skal beskrives nærmere. Det er blant annet ikke alltid like enkelt å vite hva som skal plasseres under hvilke entitet, og det dukker opp tilfeller ved en implementasjon hvor man må velge en best mulig avgjørelse av flere mulige.

verk

Figur 6.1: Verk

```
<topic id="T-070.012.870-6">
   <instanceOf>
      <topicRef xlink:href="#verk"/>
   </instanceOf>
   <baseName>
      <baseNameString>But Beautiful</baseNameString>
   </baseName>
   <occurrence>
      <instanceOf>
         <topicRef xlink:href="#tittel"/>
      </instanceOf>
      <resourceData>But Beatiful</resourceData>
   </occurrence>
   <occurrence>
   <instanceOf>
         <topicRef xlink:href="#form"/>
      </instanceOf>
      <resourceData>Musikk</resourceData>
   </occurrence>
</topic>


<topic id="verk">
   <subjectIdentity>
       <subjectIndicatorRef xlink:href=
       "http://www.idi.ntnu.no/~maribro/FRBR/psi.html#verk"/>
   </subjectIdentity>
   <baseName>
      <baseNameString>Verk</baseNameString>
   </baseName>
</topic>
```

Figur 6.2: XTM-kode for verk

Når det gjelder verk kan det være vanskelig å si hva som skal være et verk, og hva som i stedet skal være et uttrykk som realiserer verket. Dette er en diskusjon som stadig pågår i forum som omtaler FRBR. Svarene kan være forskjellige fra kultur til kultur ettersom hvordan man ser på ulike ting. Denne kategoriseringen av verk er et problem når det gjelder bøker, men det er ennå vanskeligere å se skillet når man begynner å blande sammen flere medier. Hvis vi benytter boken Lés Miserables av Victor Hugo som eksempel, så har denne boken vært opphav for blant annet en musikal med samme navn, i tillegg til en rekke filmer. Selv om disse er sterkt knyttet til hverandre og baserer seg på den samme historien, så vil de likevel være ulike verk på bakgrunn av at de alle er sprunget ut av et selvstendige, intellektuelle arbeid som er svært forskjellige fra hverandre.

Et vanskeligere dilemma innenfor musikkverdenen, gjelder sanger som består av både tekst og melodi. Vi kan tenke oss at teksten og melodien er separate verk fordi de har forskjellig uttrykk, kan være laget av forskjellige personer til forskjellig tid, osv. Et dikt kan f.eks. være skrevet av en forfatter, og er da et typisk «verk». En komponist kan flere år etterpå finne ut at han vil lage melodi til diktet, og lager da en melodi som et selvstendig verk. I dette tilfellet er det naturlig at diktet og melodien er separate verk. Vanskeligere å skille er det når en musiker sitter å prøver seg frem på gitaren og lager melodi og tekst samtidig. Det er da mer naturlig å tenke på både tekst og melodi som et samleverk.

Under utviklingen av emnekartet for denne oppgaven var det i utgangspunktet meningen å skille

melodi og tekst som forskjellige verk, men det viste seg at dette ble vanskelig å gjennomføre. Dette er ingen form som vanligvis blir fulgt i andre sammenhenger, slik at det var vanskelig å oppdrive nok informasjon. Hvis man har en sang, er det alltid oppgitt navn på den eller de som har laget den, men det er vanskelig å finne ut hvem som laget melodien og hvem som laget teksten. Ofte er personene oppgitt som både forfattere og komponister, og det er grunn til å tro at sangene har blitt utviklet i form av samarbeid. Det blir derfor også naturlig å si at sangen er *ett* verk.

Ett annet dilemma er å bestemme om sangene som forekommer på en manifestasjon i form av en CD, er separate verk, eller om hele CD-en kommer fra et verk. Det vil i de fleste tilfeller være naturlig å si at hver sang springer ut fra et intellektuelt arbeid, og er separate verk. Et typisk eksempel på dette er samleCD-er, som inneholder sanger fra mange forskjellige plater. Sangene tilsammen danner ikke noen typisk enhet, og det er liten tvil om at disse er separate verk. Vanskeligere er det med for eksempel «Eine Alpensinfine» komponert av Richard Strauss. Dette må sees på som *ett* verk, selv om den er delt opp i forskjellige satser, som blir konkretisert som separate spor på en CD. Grunnen til at dette er *ett* verk er at satsene henger sammen, og danner en helhet, en symfoni. Dette kan sammenliknes med kapitlene i en bok. Kapitlene vil ikke bli sett på som separate verk, men hjelper til med oppbyggelsen av boken som tilslutt er det totale verket.

I denne oppgaven er det sanger sunget av Billie Holiday som er implementert som verk, og i figur 6.1 på side 58 ser vi et utklipp fra omnigator for hvordan verkene er visualisert som en liste med emner. Kodeeksempel 6.2 på forrige side viser hvordan verkene er notert i XTM-syntaksen. Øverst er den unike ID´en for emnet. Deretter er det sagt at dette emnet skal være av typen «verk», og skal hete «But Beautiful». Det er også lagt til noen interne forekomster som tittel og at verket har en form av typen musikk. Selve emnetypen «Verk» er spesifisert som eget emne fordi alle sanger av typen verk vil spesifisere at de er av denne typen.

Uttrykk og Manifestasjon

Uttrykk skal vise til hva som realiserer et verk. Når det gjelder bøker kan dette typisk være forskjellige oversettelser av en bok. For musikk kan dette være forskjellige lydopptak som verket har blitt realisert ved. Problemet er å skille mellom hva som er et uttrykk og når det i stedet blir en manifestasjon.

Er for eksempel en konsert et uttrykk eller en manifestasjon? En konsert kan bli sett på som en virkeliggjøring, og danner et uttrykk av et verk på den måten at publikum kan oppleve musikken. Vi kan også si at notene som blir spilt, og eventuell koreografi, er et uttrykk. Men man kan også si at en konsert blir gjort fysisk tilgjengelig for publikum som betaler for å se en fremføring, og at det da kan bli sett på som en manifestasjon. Man kan jo si at publikum betaler for en fysisk fremføring i form av en konsert, på samme måte som de betaler for en CD. Forskjellen er at konserten ikke er et manifest i den forstand at man kan se den identiske konserten på nytt, det er en engangshendelse som ikke er lagret på noe medium.

Hvordan man vil løse dette avhenger av hvilken kultur man tilhører. Det er en vanlig oppfatning at en manifestasjon er noe som kun trenger å «sanses», men en motstridende definisjon hentet fra digitale bibliotek sier:

Manifestation: «Form given to an expression of a work, e.g., by representing it in digital form.»[24]

I følge denne definisjonen er det tydelig at en konsert ikke er en manifestasjon. Konserten må bli tatt opp slik at det er representert på digital form, og dette digitale opptaket blir da en manifestasjon, mens selve fremføringen av konserten forblir et uttrykk.

Eksemplar

Eksemplar er den konkrete tingen man kan holde i hånda, og her er det ikke så mange rom for diskusjoner. En spesiell CD er et typisk eksemplar, og det eneste som kan spesifiseres er om hvert spor på CD´en også skal spesifiseres som egne eksemplarer.

Hvis man skal låne en CD på biblioteket er det ikke naturlig å tenke på hvert spor som et eksemplar, og at man låner mange spor, da låner man den spesielle CD-en. Hvis man derimot eier en CD, som man kopierer inn på datamaskinen i form av mp3-filer, blir CD-en delt opp i større grad, og det kan være naturlig å se på hver sang, hver mp3-fil, som et eksemplar.

I emnekartet i denne oppgaven, er hvert spor implementert som separate eksemplarer i tillegg til at den totale CD-en også er et eksemplar. Det er tatt utgangspunkt i at emnekartet i stor grad skal representere innholdet av musikkfiler som brukeren har på sin egen maskin, og at brukeren da kan strukturere filene etter eget ønske og ikke nødvendigvis etter hele CD-er. Det er ikke sikkert brukeren har innholdet til hele CD-er på datamaskina, men kun et spor i form av en mp3-fil. Det er da ønskelig at emnekartet kan vise relasjonen til hvilke CD filen tilhører, og hvor denne CD-en befinner seg, eventuelt hvor den kan lånes fra.

6.2.2 Assosiasjoner

Assosiasjon defineres i bokmålsordboka som en forbindelse mellom tanker og forestillinger (el. sanseinntrykk), ord som vekker assosiasjoner. Mens relasjoner defineres som en henvisning til en beslektet ressurs.

I praksis når det gjelder implementasjon av digitale ressurser så går disse to begrepene over i hverandre. Begge begrepene går ut på å lenke sammen to ressurser som har noe med hverandre å gjøre, og det er mest naturlig å kalle det relasjon. Men når det gjelder emnekart så er emnekarttankegangen preget av at man tenker mer på de menneskelige aspektene og skal vise hvordan emner henger sammen utfra hvordan det oppfattes i virkeligheten. Det er derfor naturlig at disse har blitt kalt assosiasjoner. Ved en kombinasjon av FRBR-modellen og emnekart-standarden, så overlapper assosiasjons- og relasjonsbegrepene hverandre, og har lik betydning.

FRBR-modellen består av entiteter og bestemte navngitte relasjoner som binder disse entitetene sammen. Når det skal lages emnekart basert på FRBR-modellen er det derfor naturlig å benytte disse definerte relasjonen for å holde seg til standarden. Relasjonene i FRBR-modellen er vist i figuren i kapittel 2.5 på side 34. Disse relasjonene blir dermed implementert som assosiasjoner i emnekartet.

XTM-koden i figur 6.3 på neste side er et eksempel på hvordan en FRBR-relasjon kan implementeres i et emnekart. Her er det en assosiasjon med navn «konkretisering» som linker sammen et uttrykk med id: T-070.012.870-6-u1 og en manifestasjon med id: CK65144-009-1997-BB-COLUMBIA. Assosiasjonene i emnekartene er i hovedsak ikke rettede, og skal kunne gå begge veier mellom emnene som er assosiert med hverandre. Derfor bør man velge et assosiasjonsnavn som er retningsnøytralt, slik som «konkretisering» er. For å ha mulighet for å spesifisere retning tar man i bruk assosiasjonstyper og assosiasjonsroller, og disse begrepene blir sett nærmere på.

```
<association id="T-070.012.870-6-u1-BB-relasjon">
   <instanceOf>
      <topicRef xlink:href="relasjoner.xtmm#konkretisering"/>
   </instanceOf>
   <member>
      <roleSpec>
         <topicRef xlink:href="relasjoner.xtmm#uttrykket"/>
      </roleSpec>
         <topicRef xlink:href="uttrykk.xtmm#T-070.012.870-6-u1"/>
   </member>
   <member>
      <roleSpec>
         <topicRef xlink:href="relasjoner.xtmm#manifestasjonen"/>
      </roleSpec>
         <topicRef xlink:href=
        "manifestasjon.xtmm#CK65144-009-1997-BB-COLUMBIA"/>
   </member>
</association>
```

Figur 6.3: «Konkretisering»-relasjon

Assosiasjonstyper

Assosiasjonstypene blir spesifisert i XTM-standarden som vist i figur 6.4 på neste side. Assosiasjonstypene i dette eksempelet er: «er konkretisert ved» og «konkretiserer». Disse assosiasjonstypene viser mulige retninger på assosiasjonen. Vi ser av XTM-koden at «uttrykket» *er konkretisert ved* «manifestasjonen», og «manifestasjonen» *konkretiserer* «uttrykket».

I FRBR-modellen er det i tillegg til relasjonene mellom entitetene også mange relasjoner som går fra en entitet til den samme entiteten, for eksempel fra «manifestasjon» til «manifestasjon» se figur 6.5 på neste side.

I kodeeksempel 6.6 på side 64, ser man på «har del»-relasjonen som er spesifisert i FRBR-modellen. Men for at relasjonen skal bli implementert i emnekartet bør assosiasjonen være retningsnøytral og selve assosiasjonen blir i eksempelet derfor kaldt «samlecd». Retningen blir spesifisert ved hjelp av assosiasjonstypene som har navnene «har del» og «er en del av», se kodeeksempel 6.7 på side 65.

Eksempelet kommer fra behovet for å vise hvordan tre CD-er er samlet sammen og lager en manifestasjon i form av en samleboks. Hver CD er en manifestasjon, men samleboksen som består av disse CD-ene er også en manifestasjon. Det er derfor naturlig å se på samleboksen som en «helhet» av CD-er, mens hver enkelt CD er en «del» av denne samleboksen.

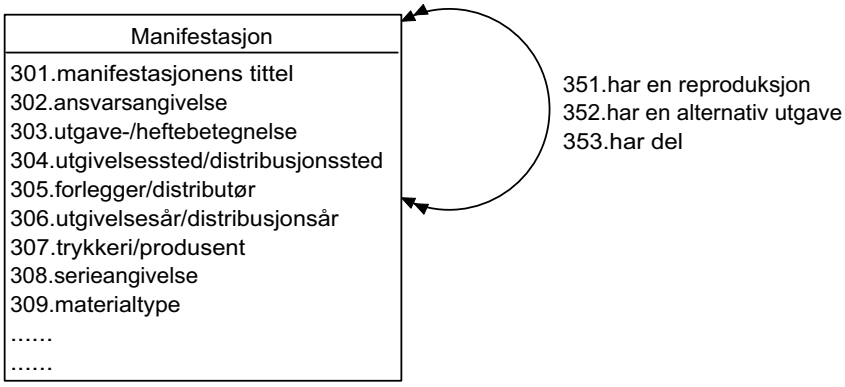
Assosiasjonsroller

Assosiasjonstypene gir navn på retningen til en assosiasjon, men for å kunne forklare hvilken retning som er riktig i hvilke sammenheng trenger vi assosiasjonsroller. I eksempel 6.3, er rolletypene som er benyttet: «uttrykket» og «manifestasjonen». Disse rolletypene er spesifisert som emner i XTM-fila som vist i kodeeksempel 6.8 på side 65.

Rollene gjør det slik at når man manøvrerer seg rundt i emnekartet og er på et emne som

```
<topic id="konkretisering">
   <baseName>
      <baseNameString>Konkretisering</baseNameString>
   </baseName>
   <baseName>
      <scope>
         <topicRef xlink:href="relasjoner.xtmm#uttrykket"/>
      </scope>
      <baseNameString>er konkretisert ved</baseNameString>
   </baseName>
   <baseName>
      <scope>
         <topicRef xlink:href="#manifestasjonen"/>
      </scope>
      <baseNameString>konkretiserer</baseNameString>
   </baseName>
</topic>
```

Figur 6.4: «konkretisering»-relasjonstype



Figur 6.5: «manifestasjon til manifestasjon»-relasjon

er spesifisert med rolletype «uttrykket» så vil det komme opp at dette emnet er konkretisert ved en spesifikk manifestasjon. Hvis man i stedet står på manifestasjonen så vil det komme opp at denne konkretiserer uttrykket. Rolletypene gjør det altså mulig å bytte mellom de ulike assosiasjonstypene ettersom hvilken retning som er tiltenkt.

I praksis vil det si at emner som inngår i en assosiasjon er tilknyttet en rolletype. Når det gjelder implementasjonen av FRBR-modellen vil det derfor si at alle entitetene må ha en rolletyper knyttet til seg, slik at rollene kan bli brukt for å forklare retningen til assosiasjonene. Rolletypenavnene er i utgangspunktet vidtomspennende begreper som kan dekke flere emnetyper, f.eks. rolletype «person» kan dekke emnetypene: «komponist», «forfatter», «ingeniør». Når det gjelder entitetene i FRBR-modellen er disse begrepene så generelle i utgangspunktet at det er vanskelig å gi dem en ennå mer overordnet rolle. En mulighet hadde vært å gitt entitetene et beskrivende rollenavn utifra forklaringene til hva hver enkelt entitet står for.

Eksempel på dette kan være:

```
<association id="hel-del-relasjon">
   <instanceOf>
      <topicRef xlink:href="relasjoner.xtmm#samlecd"/>
   </instanceOf>
   <member>
      <roleSpec>
         <topicRef xlink:href="relasjoner.xtmm#helhet"/>
      </roleSpec>
         <topicRef xlink:href="manifestasjon.xtmm#GLD25311"/>
   </member>
   <member>
      <roleSpec>
         <topicRef xlink:href="relasjoner.xtmm#del"/>
      </roleSpec>
         <topicRef xlink:href="manifestasjon.xtmm#GLD25311-1"/>
         <topicRef xlink:href="manifestasjon.xtmm#GLD25311-2"/>
         <topicRef xlink:href="manifestasjon.xtmm#GLD25311-3"/>
   </member>
</association>
```

Figur 6.6: «hel-del»-relasjon

- verk = abstrakt arbeid
- uttrykk = virkeliggjøring
- manifestasjon = konkret utforming
- eksemplar = konkret eksemplar

Dette blir fort uoversiktlig, og kompliserer modellen for brukere som ikke nødvendigvis kan FRBR-modellen fra før. Det blir fort vanskelig å skjønne sammenhengen til entitetstypene, og det blir bare flere vanskelige ord å sette seg inn i.

Ved bruk av FRBR-modellen i emnekart kan det være mest hensiktsmessig å ikke legge informasjon i navnene på rolletypene, men kun holde seg til modellen og ha med roller fordi det er nødvendig for å vise retninger for assosiasjonene. Det er i denne oppgaven derfor valgt å benytte navn som er nesten identiske med entitetstypene.

- verk = verket
- uttrykk = uttrykket
- manifestasjon = manifestasjonen
- eksemplar = eksemplaret

Det mest brukervennlige hadde vært å ha helt identiske navn (verk = verk), og jeg vil anbefale dette ved senere implementasjoner av FRBR-modellen i emnekart. Jeg valgte å ha en liten forskjell på ordene for å lettere kunne skille i XTM-syntaksen hvor det var spesifisert roller og hvor det var emner, for læringsprosessens og visualiseringens del.

Ved implementasjon hvor man bestemmer seg for å ha like navn på emnetypene som rolletypene må man også være klar over at XTM-syntaksen ikke godtar at flere emner kan ha samme navn. Man må derfor spesifisere rolletypene og emnetypene som hvert sitt emne, med likt basisnavn men med ulik identifikator.

```
<topic id="samlecd">
   <baseName>
      <baseNameString>Samlecd</baseNameString>
   </baseName>
   <baseName>
      <scope>
         <topicRef xlink:href="#helhet"/>
      </scope>
      <baseNameString>har del</baseNameString>
   </baseName>
   <baseName>
      <scope>
         <topicRef xlink:href="#del"/>
      </scope>
      <baseNameString>er en del av</baseNameString>
   </baseName>
</topic>
```

Figur 6.7: «samle-CD» - relasjon: Typer

```
<topic id="manifestasjonen">
   <baseName>
      <baseNameString>Manifestasjonen</baseNameString>
   </baseName>
</topic>

<topic id="uttrykket">
   <baseName>
      <baseNameString>Uttrykket</baseNameString>
   </baseName>
</topic>
```

Figur 6.8: Relasjon konkretisering: Rolletyper

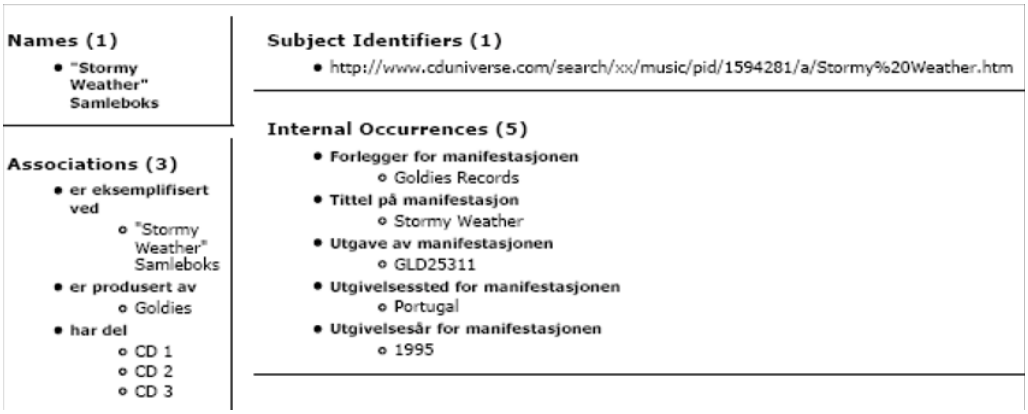
6.2.3 Forekomster

Når det gjelder FRBR-modellen så er entitetene og assosiasjonene veldig lett overførbare til emner og relasjoner i emnekart. Attributtene er ikke fullt så innlysende hvordan skal implementeres.

Det er ingen tydelig støtte for attributter fra ER-modeller i emnekart, men det mest nærliggende er å tenke på attributter som interne forekomster siden attributtene inneholder forklarende ressurser om emnene. Eksempler på attributter i FRBR-modellen er: Tittel, årstall, skillende karakteristika, toneart, besetning osv. Dette er typiske metadata hvor man kan legge inn informasjon om en ressurs, og er ikke ressurser i seg selv. Det er ikke naturlig å linke til attributtene og hente dem som ressurser, og vil derfor ikke være eksterne forekomster.

Attributtene kunne vært implementert som separate emnetyper, men dette ville fort blitt uoversiktlig. Det er veldig mange attributter som er representert i FRBR-modellen, og disse har ingen naturlig nettverksstruktur seg i mellom og linker kun til entiteten de representerer. Entitetene i FRBR-modellen er det naturlige valget når det gjelder emnetypene i emnekartetet, og det blir uoversiktlig å skulle implementere entiteter og attributter som emnetyper på lik linje med hverandre.

I denne oppgaven ble derfor attributtene i FRBR-modellen implementert som interne



Figur 6.9: Attributter som interne forekomster

forekomster i emnekartet, se utklipp fra omnigator i figur 6.9.

Det kan også være naturlig å ha eksterne forekomster knyttet til FRBR-modellen, men det er ikke for alle entitetene dette er like relevant. «Eksemplar» er en entitet som henviser til et konkret eksemplar man kan se på/lese på osv, og hvor det kan være hensiktsmessig å knytte til en ekstern ressurs. Entiteten «person» kan f.eks. ha en internettside om seg selv det kan være aktuelt å linke til, og gruppe 3 som inneholder objekter kan ha videre beskrivelser på eksterne ressurser. Men «verk», «uttrykk» og «manifestasjon» viser bare sammenhengene videre til selve eksemplaret, og henviser ikke i utgangspunktet til konkrete ressurser.

Emnetypene, assosiasjonstypene, assosiasjonsrolletypene og forekomsttypene som er implementert i emnekartet basert på FRBR-modellen, er vist i tillegg D på side 99.

6.2.4 Temaidentitet

Publiserte temaindikatorer og temaidentifikatorer

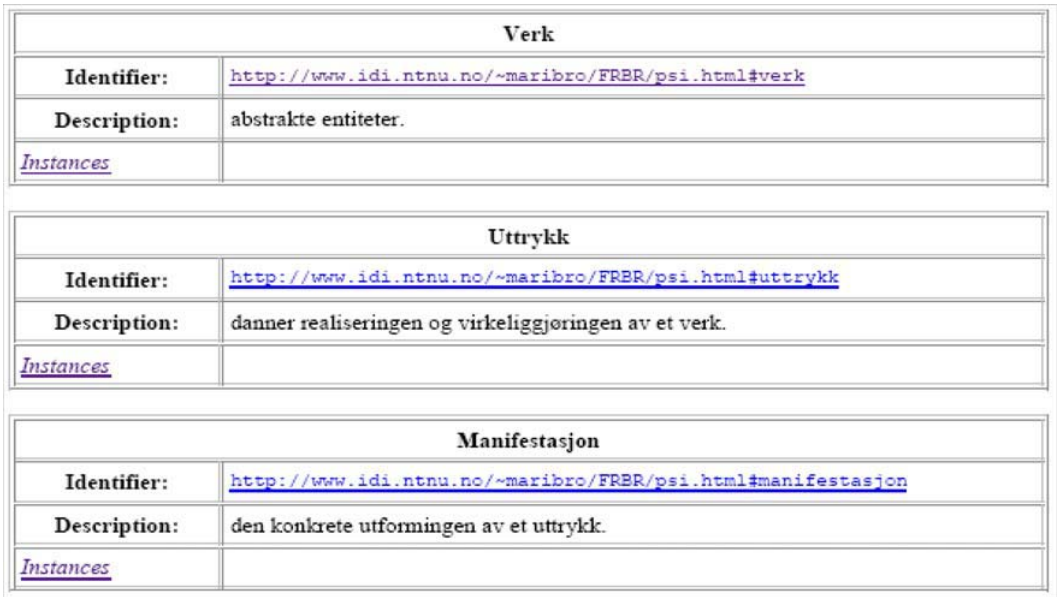
Det er i emnekartet i oppgaven knyttet PSI-er (Published subject identifier) til alle entitetstypene som forekommer. Dette er identifikatorer i form av URI´er, og danner en unik adresse for hver entitetstype. I oppgaven er alle PSI-ene samlet og publisert på en internettside, slik at de i teorien kan være mulige å benytte for andre som skal lage emnekart for samme emner. Dette er nok lite aktuelt fordi min samling ikke er spesielt stor og ikke gjort spesielt tilgjengelig, og fungerer mest som et hjelpemiddel for å holde oversikten over identifikatoren i denne oppgaven. Det er også en liten forklaring til PSI-ene på internettsiden som er lettere å forstå for mennesket, og forteller hva identifikatoren identifiserer. I figur 6.11 på neste side er det et lite klipp fra internettsiden.

Eksempel på hvordan en PSI er implementert i koden vises i figur 6.10 på neste side. Vi ser her at PSI-linken er plassert inni en subjectIdentity-tag.

Det er i emnekartet i oppgaven knyttet PSI-er (Published subject identifier) til alle entitetstypene som forekommer. Dette er identifikatorer i form av URI-er, og danner en unik adresse for hver entitetstype som lenker til en publisert internettside (se kapittel 3.7 på side 40). I oppgaven tar denne internettsiden form som en samling av alle PSI-er som er i bruk i emnekartet, med en

```
<topic id="verk">
   <subjectIdentity>
       <subjectIndicatorRef xlink:href=
       "http://www.idi.ntnu.no/~maribro/FRBR/psi.html#verk"/>
   </subjectIdentity>
   <baseName>
      <baseNameString>Verk</baseNameString>
   </baseName>
</topic>
```

Figur 6.10: PSI implementert i koden



Verk	
Identifier:	http://www.idi.ntnu.no/~maribro/FRBR/psi.html#verk
Description:	abstrakte entiteter.
Instances	

Uttrykk	
Identifier:	http://www.idi.ntnu.no/~maribro/FRBR/psi.html#uttrykk
Description:	danner realiseringen og virkeliggjøringen av et verk.
Instances	

Manifestasjon	
Identifier:	http://www.idi.ntnu.no/~maribro/FRBR/psi.html#manifestasjon
Description:	den konkrete utformingen av et uttrykk.
Instances	

Figur 6.11: Internettside med PSI

tilhørende liten forklaring som er lett forståelig for mennesket.

ID verk

For «verk» finnes ISO-standarden, ISO 15707[20] «International Standard Musical Work Code» som er utviklet spesielt for dette formålet. Dette er en kode som gir varige og unike identifikatorer til musikkverk, og som gjør det mulig å skille ulike verk fra hverandre. Det er flere identifikatorer som kunne blitt brukt for entiteten «verk» i FRBR-modellen, men siden implementasjonen i denne oppgaven har fokus på musikkdata var det naturlig å velge ISWC.

Måten slike ISWC-er er bygget opp på, er at alle koder starter med bokstaven «T», noe som indikerer at det er en ISWC. Deretter følger ni tall, fra 00000001 til 999999999, som kan være delt med punktum, og som tilsammen danner et unikt tall. Til slutt kommer et sjekktall som er knyttet til de andre tallene med en bindestrek.

ISWC-identifikatoren til verket «I´m A Fool To Want You», laget av: Wolf, Herron, Sinatra, ser ut på denne måten: T-070.903.671-4

Identifikatorene samles i en database som er tilgjengelig verden over, noe som gjør det lettere å administrere musikkverk på et internationalt plan. Denne databasen er offentlig tilgjengelig, slik at alle kan benytte samme kode på samme verk.

I denne oppgaven ble ISWC-databasen [25] benyttet for å finne unike identifikatorer til verkene i emnekartet. Ulempen er at databasen er under utvikling, slik at ikke alle verk er registrert der ennå. Det var ikke alle verkene som er tatt med i oppgavens emnekart som fantes i ISWC-databasen, og disse har derfor egne identifikatorer. Disse verkene har sangtittel som ID, noe som ikke er en fullgod løsning siden det kan finnes helt uavhengige og forskjellige verk, som kan ha samme tittel. Denne løsningen er allikevel valgt fordi det stort sett fremkommer av relasjonene om hvem som har skapt verket, at det er ulike verk. Eksempel på sangtittel benyttet som ID: <topic id=«if-you-were-mine»>

ID uttrykk

Det er ingen av CIS-standardene som er en fullgod identifikator for «uttrykk». Men når man skal identifisere «uttrykk» av musikkverk kan man bruke koden kalt International Standard Recording Code (ISRC)ISO 3901 [26], som er utviklet for spesielt for inspillinger. Ulempen med denne er at det ikke finnes noen tilgjengelig database for ISRC på samme måte som for ISWC. Det er kun plateselskaper o.l. som får se disse kodene. Ved å kontakte Granmo, et firma i Norge som har ansvaret for ISRC, ble det hevdet at en slik åpen database var noe som kom til å komme, men at det foreløpig ikke var mulig å se den. I denne oppgaven er derfor ikke ISRC blitt tatt i bruk. Det er i stedet laget en egen kode for å identifisere de ulike uttrykkene fra hverandre, og denne er basert på ISWC-standarden som ble benyttet for verk.

ISWC-standarden er benyttet som en basis, i tillegg til å legge til en utvidelse. På denne måten ser man lett sammenhengen mellom verket og uttrykket som verket er realisert ved.

Eksempel på dette:

- T-070.903.671-4-u1
- T-070.903.671-4-u2
- T-070.903.671-4-u3

Første delen av identifikatoren, T-070.903.671-4, er den samme som beskrevet i «verk». Endelsen u1, u2 og u3 skiller de ulike uttrykkene fra hverandre, og man kan lett se at dette er et verk som er realisert ved tre forskjellige uttrykk.

ID manifestasjon

For manifestasjoner kan man benytte alle ID-er som kan identifisere et spesielt produkt. De fleste manifestasjoner som publiseres har identifikatorer knyttet til seg som f.eks. CD-id for musikk, ISBN-nr for bøker osv.

I implementasjonen i denne oppgaven er det laget en egendefinert identifikator for manifestasjonene. Eksempel på dette er:

CK65144-001-1997-IAFTWY-COLUMBIA

Forklaring til koden:

- CK65144 = CD-ID oppgitt på CD

- 001 = spor-nummer som sangen er plassert på
- 1997 = årstall for når cd ble spilt inn
- IAFTWY = forbokstaven til alle ordene i tittelen på sangen. I dette tilfellet er bokstavene hentet fra sangtittelen «I´m A Fool To Want You»
- COLUMBIA = plateselskapet som har utgitt sangen

En mulighet hadde også vært å bruke International Standard Music Number (ISMN) hvis det var snakk om manifestasjoner i form av noter. Men disse identifikatorene er ikke tilgjengelige i offentlige databaser. Men noe av strukturen på koden er kjent, slik at man eventuelt kan prøve å lage egne nummer.

Eksempel: M-2306-7118-7

- M = skiller ISMN fra andre standardnummer
- 2306 = utgiver-ID
- 7118 = note-ID
- 7 = sjekknummer

ID eksemplar

I dette kapittelet er det for «eksemplar» fulgt samme mønster når det gjelder identifisering som det er for uttrykk. Det er valgt å bruke uttrykksidentifikatoren som en stamme, og legge til en tilleggskode.

Vi ser i eksempelet under at verket, T-070.903.671-4, har tre forskjellige uttrykk, som beskrevet ovenfor. Hvert av de tre uttrykkene er eksemplifisert ved hvert sitt eksemplar -e1.

- T-070.903.671-4-u1-e1
- T-070.903.671-4-u2-e1
- T-070.903.671-4-u3-e1

Hvis det er flere eksemplarer av samme uttrykk vil disse få ID som slutter på -e2, -e3, - e4 osv.

6.3 Evaluering

FRBR-modellen var relativt lett å implementere som emnekart fordi de to standardene hadde komponenter som kunne kobles sammen. Entiteter kunne implementeres som emnetyper, relasjoner som assosiasjonstyper og attributter som interne forekomster. Det var derfor ikke nødvendig å avvike fra FRBR-modellens struktur, og komponenter kunne opprettholdes slik det var tiltenkt fra standarden i utgangspunktet.

Innholdet det ble tatt utgangspunkt i var musikk-metadata, og implementasjonen ble dermed også en test for hvordan slike data kunne plasseres i FRBR-modellen. Modellen er kompleks og det var ikke alltid like enkelt å vite hvilke data som skulle under hvilke entitet. Det ble valgt å se på den abstrakte melodien og sangteksten i sammenheng som et verk. Lydopptaket, eller

en annen form for realisering av verket, ble satt under uttrykk. Videre ble en konkret CD, eller lydfil, plassert under manifestasjon. Eksemplaret henspeilet til den konkrete CD-en som noen kunne ha stående i hyllen. Alle musikkverkene som ble implementert i FRBR-modellen var sunget eller skrevet av Billie Holiday, og metadata om henne ble lagt under person, sammen med andre tekstforfattere og komponister.

Entitetene «verk», «uttrykk», «manifestasjon», «eksemplar», «person» osv. fra FRBR-modellen, ble implementert som emnetyper i emnekartet. Videre ble relasjonene implementert som assosiasjoner. Det ble i tillegg til hovedrelasjonene mellom entitetene implementert en «hel-del»-relasjon som gikk fra manifestasjon til manifestasjon for å relatere tre CD-er til hverandre for å vise at de var del av en samle-CD (helhet).

Attributtene ble implementert som interne forekomster. Det ble her kun lagt inn de attributtene som var av relevans for musikkdataene som ble benyttet. FRBR-modellen inneholder en rekke attributter under hver entitet som skal dekke mange ulike bruksområder, hvorav de fleste er spesielt tilpasset metadata om litteratur. Men det er også attributter som passer spesielt for musikk-metadata, i tillegg til at noen attributter passer for alle typer data som f.eks. «tittel» og «årstall».

Når det gjelder identifikatorer så ble det benyttet standarder for musikkidentifikasjon der dette var mulig. Ellers ble det laget egendefinerte identifikatorer for emnene. Emnetypene fikk PSI-er tilknyttet seg, som viste til en internettside.

FRBR-modellen er relativt kompleks, men den gir rom for å kunne strukturere mye informasjon på en ryddig måte. Modellens komponenter er lett overførbare til emnekartstandarden, og det er derfor mulig å kombinere den gode strukturingen av data med emnekartstandardens gode navigeringsmuligheter.

Kapittel 7

Sammenfletting av emnekart

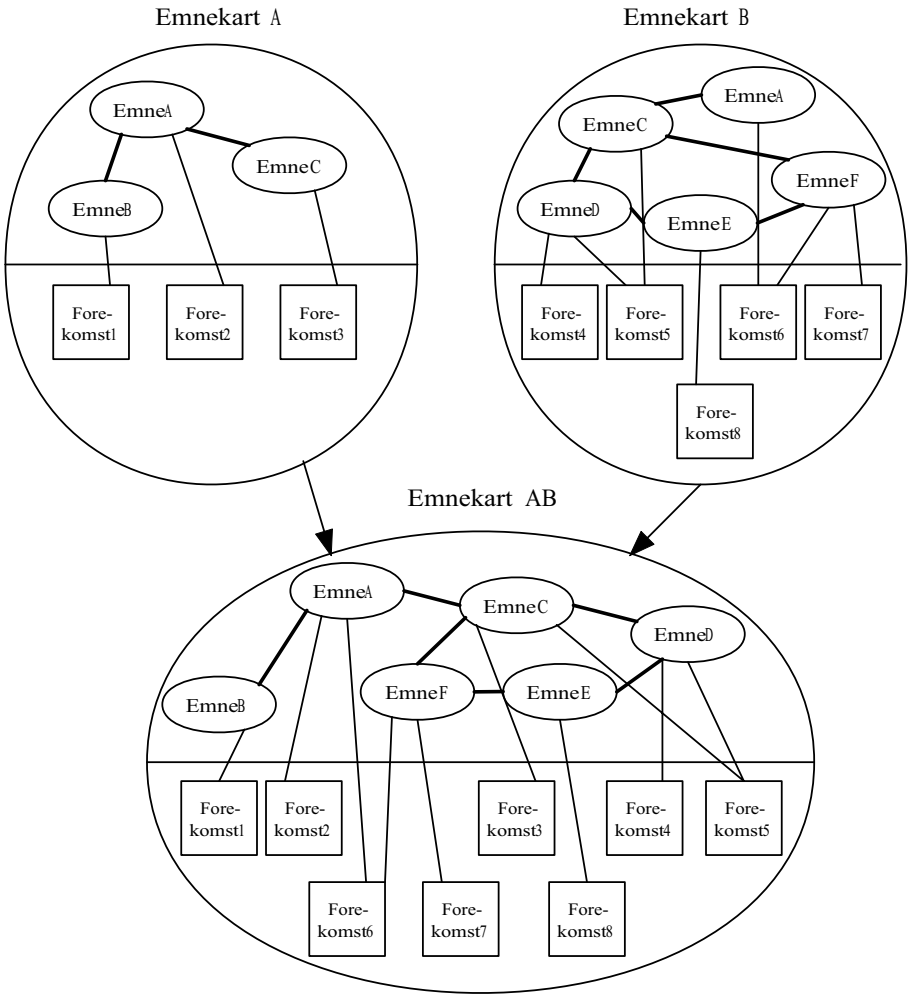
Det er flere store nettsteder som har tatt i bruk emnekart, for eksempel forskning.no og forbrukerportalen.no. Men disse portalene har flere like emner, med liknende innhold, begge har blant annet emner som omhandler «helse». Dette resulterer i at brukeren fremdeles må gå inn på begge portalene for å finne all info om dette emnet.

Målet med emnekart er å kunne samle all informasjon om et tema på en plass. Vi så i kapittel 3.7 på side 40, hvordan temaidentitet hjelper til så vi unngår å ha flere like emner i et emnekart. Men dette hjelper kun på å ha samlet all informasjon om et emne innenfor et emnekart, hva når det finnes flere emnekart som omhandler samme emne?

For å løse dette problemet er det i emnekartstandarden mulig å sammenflette og koble sammen flere emnekart slik at de til sammen utgjør et felles emnekart. Disse emnekartene kan hver for seg ha både like og forskjellige emner, men når de flettes sammen kan vi tenke oss at emnekartene blir lagt oppå hverandre og duplikater av emner og assosiasjoner blir fjernet.

Figur 7.1 på neste side, viser hvordan vi slår sammen to emnekart, «Emnekart A» og «Emnekart B». Emnekart A inneholder tre emner som er assosiert med hverandre, i tillegg til at emnene er koblet sammen med forekomster. Emnekart B inneholder fem emner i tillegg til assosiasjoner. Og vi ser at et emne kan være assosiert med flere forekomster og flere emner. Vi ser at begge emnekartene har to like emner: «Emne A» og «Emne C».

Ved sammenslåing så blir alle emner, forekomster og assosiasjoner fra begge emnekartene slått sammen slik at de danner et emnekart. Men som vi ser i figuren, så blir «Emne A» og «Emne C» kun vist en gang. Det er fordi emnene med like navn blir slått sammen for å unngå duplikater. Emnet som da består er en union av de to opprinnelige emnene og vil være tilkoblet alle assosiasjoner og forekomster som de to separate emnene tidligere hadde. Dette ser vi i figuren ved at «Emne A» var tilknyttet «Forekomst 2» i emnekart A, mens «Emne A» var tilknyttet «Forekomst 6» i emnekart B. Når emnekartene da blir slått sammen så blir «Emne A» tilknyttet både «Forekomst 2 og 6. Det samme gjelder assosiasjoner mellom emner. «Emne C» er assosiert med «Emne A» i emnekart A, og i emnekart B er det assosiert med «Emne A», «Emne D» og «Emne F». Ved sammenslåing er «Emne C» dermed assosiert med «Emne A», «Emne D» og «Emne F». I kapittel 7.2 på side 76, er det sett nærmere på eksempler som viser hvordan dette fungerer i praksis.



Figur 7.1: Sammenfletting av emnekart A og B

7.1 Metoder

I emnekartstandarden XTM 1.0 [27] er det angitt tre måter som bestemmer når to emner representerer det samme temaet. Og det gjør de dersom:

1. emnene har samme ressurs som tema. Dette innebærer at temaet er en adresserbar ressurs.

2. to emner som bruker samme ressurs for å beskrive et tema, representerer det samme temaet. Det vil si at de to emnene bruker samme identifikator.

3. basisnavnet for to emner består av samme streng og basisnavnene er i samme perspektiv.

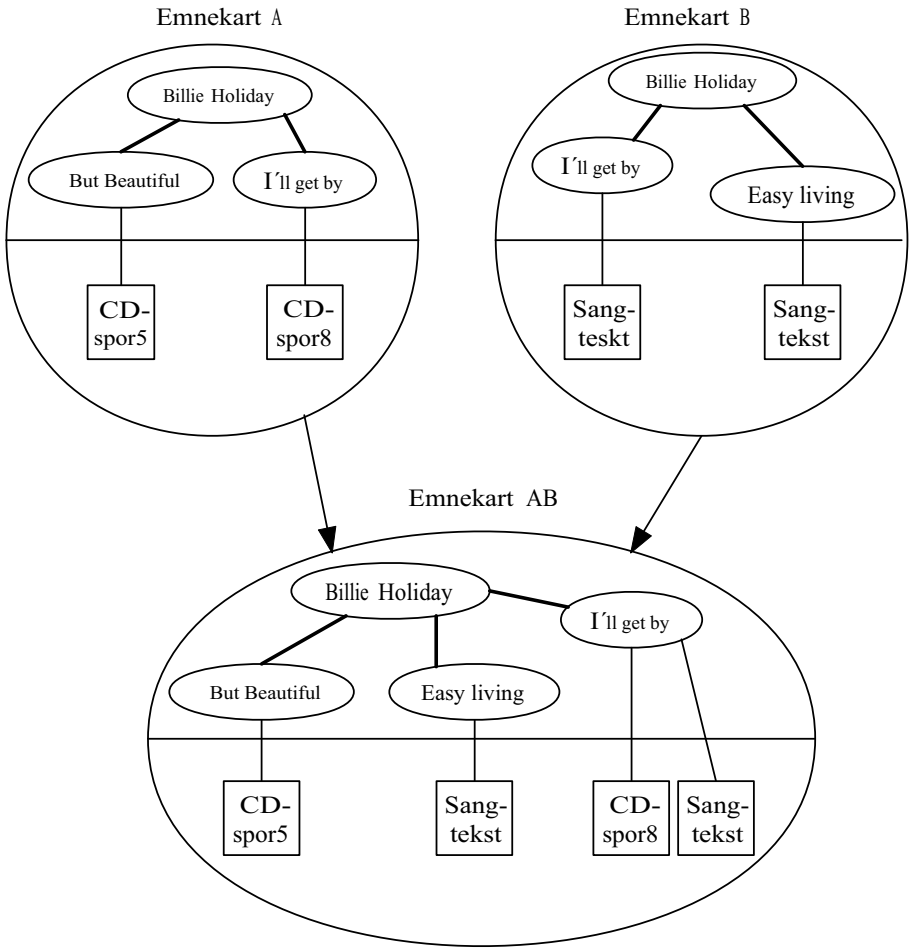
Punkt 1 og 2 blir løst ved metoden temabasert sammenfletting. Mens punkt tre inngår i metoden om navnebasert sammenfletting. Disse to metodene skal bli forklart nærmere.

7.1.1 Navnebasert sammenfletting

Navnebasert sammenfletting går ut på at man tar utgangspunkt i basisnavnene til emnene ved sammenfletting. Alle emner har basisnavn og hvis to emner har samme basisnavn antas det at disse representerer det samme temaet, og de blir da slått sammen. For eksempel så blir et emne med basisnavn «Don´t explain», slått sammen med et annet emne med samme basisnavn i et annet emnekart.

Figur 7.2 viser to separate emnekart, A og B, med hver sine emner og tilhørende forekomster. Av figuren ser man at begge emnekartene inneholder en emnetype med navnet «Billie Holiday». Under emnetypen «I´ll get by». Dette emnenavnet er identisk i begge emnekartene og ved sammenslåing blir dette emnet slått sammen slik at de kun representeres en gang.

Alle emnene er tilknyttet forekomster. Emnene i emnekart A hadde forekomster representert ved CD-spor, mens emnekart B hadde tekstene til sangene som forekomster. Ved sammenslåing ble dermed emnet, som var representert i begge emnekartene, tilknyttet både tilhørende CD-spor og sangtekst.



Figur 7.2: Navnebasert sammenfletting

Problemer med navnebasert sammenfletting er at det ikke er sikkert at emnene har helt identiske basisnavn selv om emnene omhandler samme tema. Synonymer, homonymer og polysemer i språket gjør det vanskelig med slik sammenfletting, og forskjellig ordbruk vil fort

forekomme når forskjellige personer lager emnekartene. Hvordan kan f.eks. en datamaskin vite at «genmodifisert mat» dreier seg om det samme som «genmodifiserte matvarer»? Det blir også umulig å slå sammen emnekart basert på forskjellig språk. En datamaskin vil ikke ha mulighet til å se at «genetically modified food» er det samme som «genmodifiserte matvarer.

Et aspekt man må være klar over ved navnebasert sammenslåing av emnekart, er håndteringen i forbindelse med perspektiv (kapittel 3.6 på side 40). Perspektivet spiller en stor rolle, og følger samme regel som for duplikate basisnavn innad i et emnekart. Hvis to emnekart blir slått sammen ved å benytte navnebasert sammenfletting, og har emner med identiske basisnavn under samme perspektiv, vil disse bli slått sammen. Hvis emnene derimot er plassert under forskjellig perspektiv, vil de bestå separat under hvert sitt perspektiv.

I følge XTM 1.0 [27], representerer emnene det samme temaet hvis basisnavnene for to emner består av samme streng og så lenge basisnavnene er i samme perspektiv.

Et problem når det gjelder perspektiv og navnebasert sammenfletting er at hvis man f.eks. har flere bøker med identisk tittel og dermed identiske basenames så kan disse bli slått sammen selv om bøkene har forskjellig forfatter.

For å løse problemene ved navnebasert sammenfletting har man tatt i bruk temabasert sammenfletting.

7.1.2 Temabasert sammenfletting

Temabasert sammenfletting går ut på at man benytter temaidentitet for å se om to emner er like og kan flettes sammen. Alle emnene må da ha hver sin temaidentifikator som unikt identifiserer de ulike temaene slik at man kan si om to emner representerer det samme temaet, og dermed skal slås sammen.

Temaidentitet vil si at man tar i bruk unike identifikatorer for hvert emne, og datamaskina trenger dermed ikke å bry seg om navn. Identifikatorer er noe som går igjen i alle databasesystemer for å kunne skille ting fra hverandre. Problemet er at alle bruker forskjellige identifikatorer og mange har flere betydninger, f.eks. kan identifikatoren «no» stå for både Norge eller norsk.

Krav til en god identifikator [28]:

- Må være unik
- Må være global
- Må være enkelt å definere
- Må være enkelt å bruke

En identifikator som dekker disse kravene for emnekartstandarden er Publiserte temaer (subjects) som er en distribuert mekanisme for tildeling av unike, globale identifikatorer.

Publiserte temaindikatorer og temaidentifikatorer

Published subject identifier (PSI) er en subject identifier (SI) i form av en URL som beskrevet i kapittel 3.7 på side 40, men er i tillegg publisert og gjort tilgjengelig for andre. På denne måten er det mulig for de som skal lage emnekart å se om andre har laget emnekart med samme emner tidligere, og de kan dermed benytte de samme PSI-ene. Ved sammenslåing av flere emnekart vil dermed like emner ha identisk identifikator og like emner vil bli slått sammen.

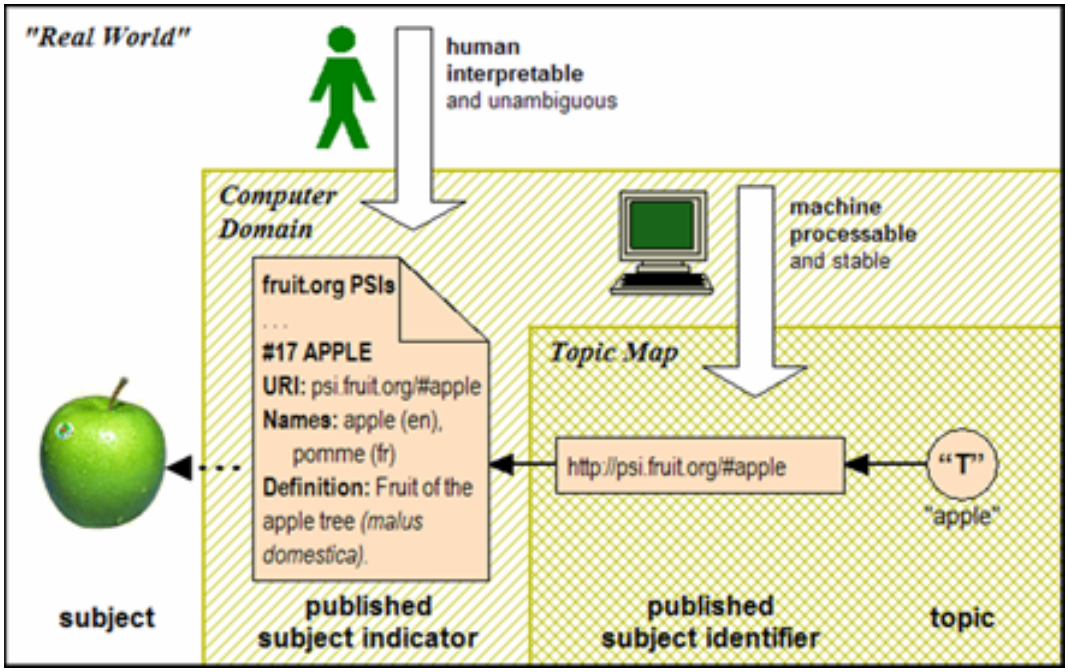
Eksempel på PSI-er som datamaskina sammenlikner og fletter sammen hvis de er like [28]:

`http://psi.kulturnett.no/museum/ibsen-museet`
`http://psi.kulturnett.no/museum/ibsen-museet` = OK

`http://psi.kulturnett.no/museum/ibsen-museet`
`http://psi.kulturnett.no/museum/ibsen-huset` = ikke OK

Vi ser av disse PSI-ene at de er lesbare for datamaskiner, men også at de er relativt forståelige for mennesket. Vi ser at det er en URL som viser til en side for PSI-er for kulturnett.no, og emnet er «ibsen-museet» som hører under emnetypen «museum». De øverste URL-ene er like og vil bli slått sammen, mens de nederste to er forskjellige, siden det nederste emnet i i stedet er «ibsen-huset», og vil ikke bli slått sammen.

Ikke alle URL-er er like forståelige for mennesket som eksempelet over, og eksempelet gir heller ikke mye info til brukeren om det f.eks. handler om Ibsen-museet i Oslo, eller Henrik Ibsen Museum i Skien. For å hjelpe til med dette kan man knytte indikatorer til PSI-en. Indikatorer er et dokument som forklarer betydningen av identifikatoren på en forståelig måte. Og dette dokumentet kommer opp ved å klikke på URL-linken.



Figur 7.3: Temaidentitet [3]

Figur 7.3 illustrerer hvordan PSI-ene, indikatordokumentet, selve emnet og verden henger sammen. I den virkelige verden finnes det et eple som skal bli omtalt som et emne i emnekartet. Emnet «apple» i emnekartet identifiseres ved en PSI: «`http://psi.fruit.org/#apple`». Denne PSI-en er prossesserbar for datamaskina. Hvis man klikker på PSI-en kommer man til et lite dokument som er en indikator for hva emnet omhandler og som er forståelig for mennesket, og mennesket kan der se at emnet handler om frukten fra et epletre.

Alle kan lage PSI, på lik linje med at alle kan lage en URL. På denne måten er det veldig enkelt å ta i bruk, og ved at alle publiserer sine identifikatorer kan andre ta i bruk allerede eksisterende PSI-er for sine identiske emner. Dette gjør sammenflettingen bedre ved at like emner blir slått

sammen.

En ulempe er at utviklere av emnekart må lete etter allerede publiserte temaidentifikatorer for hvert emne de lager, noe som kan være ressurskrevende. URL-er er svært enkelt å lage, og det er grunn til å tro at mange vil velge å lage nye egne PSI-er for sitt emnekart i stedet for å benytte seg av identifikatorer som er publisert av andre. For at utviklere skal bruker PSI i stedet for å lage egne identifikatorer bør det finnes et sterkt ønske om å kunne slå flere emnekart sammen. Det bør også lages et godt system for å publisere temaindikatorer slik at alle PSI-ene ligger på samme plass, og er lette og finne frem i.

Det er uklart hvem som skal stå for en slik samling av publiserte temaidentifikatorer og temaindikatorer. Steve Pepper [29] sier at det skal være et dugnadsarbeid å stå for slik publisering, men at noen med store samlinger av emner og autoritet bør gå foran. Instanser av en sånn art kan være Nasjonalbiblioteket eller Norsk Digitalt bibliotek, som har enorme samlinger. Hvis disse hadde hatt PSI-er på alle sine ressurser offentlig tilgjengelig, og i et system som var lett å finne frem i, ville dette gjort det mer aktuelt at alle benyttet samme PSI på samme emner. Men det er usikkert om Nasjonalbiblioteket og Norsk Digitalt bibliotek vil være villige til å publisere alle emnene de har. De har investert store ressurser i samlingene sine, og er kanskje ikke interessert i at de skal ligge åpent for alle, uten kontroll.

7.2 Testcase

For å se hvordan sammenfletting fungerer i praksis er det aktuelt å teste ut hvordan det vil fungere med sammenfletting av emnekartene som er implementert tidligere i oppgaven. Det vil først bli sett på sammenfletting av emnekart basert på Dublin Core, Dewey og FRBR-modellen. Deretter vil det testes hvordan det vil fungere med kombinert sammenfletting på tvers av de ulike standardene. Det interessante er å se om strukturen i emnekartene blir opprettholdt og om like emner blir slått sammen slik det er tiltenkt, og hva som fungerer best av navnebasert og temabasert sammenfletting.

Ved implementasjonen av emnekartene ble programmet, Omnigator, benyttet. Dette programmet har en merge-funksjonen som automatisk slår sammen de emnekartene man ønsker. Denne funksjonen benytter i utgangspunktet temabasert sammenfletting, og slår sammen emner som har lik PSI. Men det finnes også et alternativ for å legge til navnebasert sammenfletting i tillegg. Omnigator slår da først sammen emner basert på lik PSI, og slår deretter sammen emner med identiske navn. Omnigator slår ikke automatisk sammen duplikater av forekomster og assosiasjoner med samme navn. Men hvis man vil unngå slike duplikater har Omnigator en ekstra funksjon for å slette disse som heter «Suppress».

7.2.1 Sammenfletting av emnekart basert på Dublin Core

Hva skal testes

Vi skal her se på hvordan sammenfletting av forskjellige emnekart basert på Dublin Core fungerer i praksis. Det er her aktuelt å skulle undersøke både hvordan det vil fungere med temabasert sammenfletting alene og med navnebasert sammenfletting i tillegg.

Hva testes det på

I kapittel 4.4 på side 44, ble det vist ulike løsningsalternativer for hvordan implementasjon av Dublin Core kan utføres. Det er her aktuelt å finne ut hvordan det vil fungere å sammenflette to av disse alternativene.

Tredje løsningsalternativ som ble beskrevet i kapittel 4.4.3 på side 46, hvor alle DC-elementene er assosiert til et samlingselement kalt «DC.record» slåes sammen med andre løsningsalternativ beskrevet i kapittel 4.4.2 på side 45 hvor det er laget egne navn på assosiasjonene, og hvor alle elementene har assosiasjon til «tittel».

De to emnekartene har i utgangspunktet forskjellig innhold, men det er blitt lagt inn noen felles emner for å kunne vise hvordan like emner blir slått sammen under merging. Ved dette eksempelet får vi testet hvordan strukturen blir opprettholdt ved emnekart basert på samme standard, selv om emnekartene er bygget opp litt forskjellig. Vi får i tillegg testet hvordan like emner og forskjellige emner blir håndtert, og om duplikater blir fjernet.

Utførelse

Det ble først benyttet kun temabasert sammenfletting, og dette fungerte veldig bra. DC-elementene har PSI-er spesifisert av Ontopia. Disse PSI-ene ligger lett tilgjengelige for alle som skal lage emnekart basert på DC, og PSI-ene vil derfor alltid være identiske for elementene i DC og disse vil bli slått sammen ved sammenslåing av flere emnekart. Eksempel på en ferdig definert PSI fra Ontopia for elementet «title», er vist i figur 7.4.

```
<topic id="Title">
   <subjectIdentity>
            <subjectIndicatorRef xlink:href=
            "http://purl.org/dc/elements/1.1/title"/>
      </subjectIdentity>
      <baseName>
       <baseNameString>Title</baseNameString>
   </baseName>
</topic>
```

Figur 7.4: PSI for DC

Navnebasert sammenslåing av DC-elementene hadde også fungert bra, siden navnene er så fastsatt i standarden og vil være identiske uansett hvem som lager emnekartet. Men dette vil kun fungere for sammenslåing av emnekart med samme språk. Ved sammenslåing av et norsk og et engelsk emnekart vil ikke emnene bli slått sammen selv om standarden er lik.

«baseName» = Tittel er ulik «baseName = Title»

7.2.2 Sammenfletting av emnekart basert på Dewey

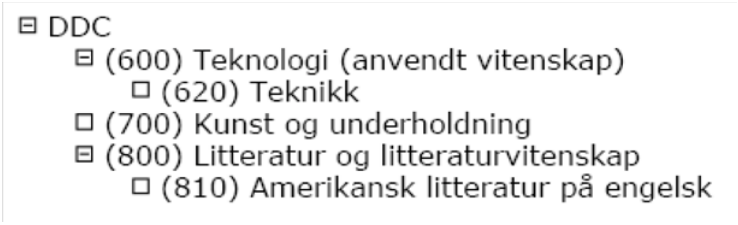
Hva skal testes

Her skal det sees på hvordan sammenfletting av forskjellige emnekart basert på Dewey fungerer. Emnekartene basert på dewey ble bygget opp hierarkisk og skiller seg derfor litt ut fra de øvrige emnekartene. Det blir her derfor naturlig å se om det er noen

forskjell på sammenflettingen når man benytter denne oppbyggingen i stedet for vanlig nettverksoppbygging.

Hva testes det på

Det blir her benyttet det testcaset som ble laget tidligere i oppgaven. Dette eksempelet inneholdt det øverste nivået av Dewey-emnene 600, 700 og 800, og hvor det videre var tatt med endel underemner til 700-tallet. Dette eksempelet blir her sammenflettet med et nytt testcase som bygger videre på 600 og 800-tallene. Se figur 7.5



Figur 7.5: Nytt testcase for Dewey

Utførelse

Dewey benytter tall for å klassifisere hvilke emner det er snakk om. Disse tallene er bestemt i klassifikasjonssystemet og er dermed ufravikelige. Det blir derfor alltid benyttet like tall ettersom hvilket emne det er i alle emnekart basert på Dewey. Dette gjør det enkelt og entydig ved sammenslåing, og fører til at strukturen blir opprettholdt så lenge man benytter samme versjon av standarden.

Resultater

Emnene i DDC-testcasene hadde ikke fått PSI-er, og ved merging kun basert på PSI ble emnekartene naturlig nok doblet uten at det ble noen forbindelse mellom dem. Det var derfor nødvendig å ta i bruk navnebasert sammenslåing. Jeg hadde først basert basenamene på Deweytallene, uten noe tekst, og det var da ikke mulighet for andre ordkombinasjoner, og tallene er like på alle språk, så det var da uproblematisk med slik sammenfletting. De like emnene hadde identiske basisnavn og ble slått sammen uten problemer og man unngikk duplikater. Strukturen ble også opprettholdt og det fungerte veldig bra med slik sammenfletting.

Men i kapittel 5 på side 51, valgte jeg i stedet å lage basenamene utifra en kombinasjon av både Deweytallene og emnenavnene for å gjøre det mer brukervennlig ved visualisering. Brukeren trenger en tekstlig forklaring i tillegg til tallene for å forstå hvor han skal lete etter emnet han er på jakt etter. Dette kompliserte den navnebaserte sammenflettingen. Så lenge man holder seg til den samme versjonen av Dewey-standarden, så skal teksten være identisk og det skal da være lett å slå sammen emnekartene. Ulempen er at det er mange versjoner, Dewey-standarden er stadig i endring og navnene endres litt på fra versjon til versjon, og dette kompliserer sammenfletting basert på navn.

En annen ulempe er at man har mange valgalternativer for hvordan man skal skrive tall og emnenavn i kombinasjon. Valgalternativer er:

- (620) Teknikk
- 620 - Teknikk
- Teknikk - 620
- osv.

Jeg valgte det alternativet som er skrevet først, selv om alle de tre alternativene som er vist stort sett er like forklarende for brukeren. Denne skrivemåten ble benyttet både i hovedemnekartet og testcaset, og det ble brukt navn fra samme versjon på begge. Sammenslåingen jeg gjorde basert på basename i dette caset ble derfor vellykket, og resultatet er vist i figur 7.6. Vi ser av figuren at emnene 620 og 810 er lagt til hovedemnekartet, og 600, 700 og 800 forekommer kun en gang.

Figur 7.6: To sammenslåtte DDC-emnekart

7.2.3 Sammenfletting av emnekart basert på FRBR

Hva skal testes

Her vil det bli sett på hvordan sammenfletting av forskjellige emnekart basert på FRBR-modellen fungerer i praksis. Det er her aktuelt å skulle undersøke både hvordan det vil fungere med kun temabasert sammenfletting og hvordan det blir med navnebasert sammenfletting i tillegg.

Hva testes det på

Emnekartet som ble laget i kapittel 6 på side 57 blir benyttet som utgangspunkt for sammenflettingen. Det er i tillegg laget et subsett av dette emnekartet som i stor grad er likt,

men som også inneholder flere ulike emner. Strukturen i emnekartene er like siden de begge er bygget opp på FRBR-standarden og følger denne i forhold til navn og assosiasjoner.

Utførelse

Jeg valgte først kun temabasert sammenfletting. Dette førte til at alle emner med identisk PSI ble slått sammen og emnet som sto igjen hadde arvet forekomstene og assosiasjonene fra begge de foregående emner. Det ble imidlertid en del duplikate emner siden ikke alle emnene hadde fått tildelt PSI-er. Temabasert sammenfletting krever at alle emner har PSI, hvis ikke har den ikke noe sammenlikningsgrunnlag.

Ved å benytte navnebasert sammenfletting i tillegg, løste dette seg imidlertid veldig bra. Denne funksjonen gikk over emnekartet etter at den temabaserte sammenflettingen var ferdig, og sammenliknet alle basisnavnene. Emnene med like basisnavn ble slått sammen. Det var nå ingen duplikate emner igjen.

En stor fordel når det gjelder sammenfletting basert på FRBR, er at strukturen er så fast. Alle emnekart basert på FRBR-modellen har akkurat den samme oppbyggingen hvor emner og assosiasjoner er de samme uansett. Ved sammenfletting vil altså ikke strukturen endre seg, og det vil fortsette å være like oversiktlig. Det eneste som skjer er at emnekartet får mer innhold og flere forekomster, det vil inneholde flere verk og flere personer osv.

Men det kan være ulemper knyttet til å basere seg på navnebasert sammenfletting. Så lenge det gjelder emnene som er fastsatt av standarden, som «verk», «person» osv. er det lite slingringsmonn på basisnavnene, og disse vil bli slått sammen. Det er også stor grunn til å tro at navn på personer ikke forandre seg så mye, men navnet kan skrives på forskjellig form, f.eks. hvilke rekkefølge fornavn og etternavn skal stå i. Det er derfor ikke sikkert at disse vil bli slått sammen. Andre emner kan også inneholde nøyaktig det samme, men når forskjellige personer har laget emnekartet er det grunn til å tro at man kan ha laget litt forskjellige navn, hvis man ikke er enige om en standard. Det er heller ikke sikkert at emnekartene som er aktuelle å slå sammen er basert på samme språk. Hvis basisnavnene i det ene er på norsk og det andre er på engelsk, vil disse ikke bli slått sammen selv om de betyr akkurat det samme.

Ennå et problem når det gjelder sammenslåing basert på basisnavn er at man må være veldig konsekvent når man bygger emnekartet så man tenker etter hvilke perspektiv/scope man tar i bruk og hvordan konsekvens dette har for sammenslåingen. Det kan være naturlig i FRBR-modellen og ha flere emner som heter det samme, men som hører under forskjellige perspektiv og derfor ikke bør slåes sammen. «Tittel» er et eksempel på et emne som kan forekomme flere ganger. Det kan være tittel for verk, eller tittel for uttrykk. Disse har ikke samme innhold og bør ikke slåes sammen. Dette kan løses slik jeg har gjort det med å i stedet kalle dem «verkets tittel» og «uttrykkets tittel», men det er ikke overalt dette er naturlig. En assosiasjon som går igjen i FRBR-modellen er «har emne», denne assosiasjonen går fra «verk» og til både «uttrykk», «manifestasjon» og «eksemplar». Dette er altså tre unike assosiasjoner men med samme navn. Disse bør heller ikke slåes sammen med navnebasert sammenfletting.

Navnebasert sammenfletting er enkel med tanke på at alle emner har basisnavn og man slipper å spesifisere PSI-er, men man må ha klart for seg hvilke perspektiver de ulike emnene er i. I FRBR-modellen kan det lønne seg å basere seg mest mulig på tema-basert sammenfletting, siden det er så mange navn som kan være aktuelle at går igjen, selv om de ikke har identisk innhold.

Resultater

Ved sammenslåingen av to emnekart basert på FRBR-modellen holdt strukturen seg når det gjaldt emner og assosiasjoner, og det nye emnekartet så nesten identisk ut som de to foregående, bortsett fra flere emner som var lagt til i strukturen.

Men det ble duplikater når det gjaldt enkelte forekomster. I det ene emnekartet var det en forekomst med navnet: «Navn», mens det i det andre emnekartet var brukt «Navn på person», mens innholdet under begge forekomstnavnene var likt og kom derfor to ganger. Dette ble imidlertid løst ved bruk av ekstrafunksjonaliteten i omnigator «suppress», som tar bort alle duplikater. Personsnavnene kom derfor kun en gang, og det overordnede navnet skiftet fra «Navn» til «Navn på person» ettersom hvilket emnekart som var valgt først i mergeprosessen.

7.3 Kombinert sammenfletting

Vi har til nå sett på sammenfletting av emnekart basert på like standarder, noe som har fungert bra siden strukturene har blitt bevart og oversikten opprettholdt. Det har kun blitt tilført flere emner, assosiasjoner og forekomster til den allerede eksisterende strukturen.

I dette kapittelet vil det bli sett på hvordan emnekart med forskjellige strukturer vil arte seg ved sammenslåing.

7.3.1 Sammenfletting av emnekart basert på Dublin Core og FRBR

Hva skal testes

Dublin Core er et sett av bestemte elementer og FRBR-modellen er en gitt modell. Disse formatene er veldig forskjellige og har hver for seg lite slingringsmonn i oppbyggingen. Det er derfor interessant å undersøke hvordan strukturen blir ved å slå emnekart basert på disse standardene sammen i ett emnekart.

Et annet moment er at samtidig som standardene er ulike, så har de også visse fellestrekk. Det er flere elementer som går igjen i forskjellige metadatastandarder slik som «tittel», «forfatter», «emne» osv. Slike felleselementer finnes også mellom FRBR-modellen og Dublin Core, og det er interessant å se hvordan disse vil bli håndtert ved sammenslåing. Blir FRBR-tittel lik som DC-tittel ved sammenslåing? og blir de plassert lik plass i emnekartet?

Hva testes det på

I denne testen er det benyttet det tredje løsningsalternativet for Dublin Core, hvor alle emnene er assosiert med DC.record. For FRBR-modellen er subsettet til FRBR-testcaset benyttet.

De to emnekartene har veldig ulikt innhold, og få emner vil antageligvis bli slått sammen, men det er likevel interessant å se hvordan strukturen blir.

Resultater

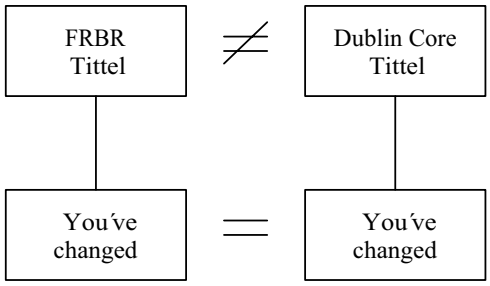
Når de to emnekartene ble slått sammen var det lite kobling mellom de to. De hadde stort sett ulike emner og resultatet ble i hovedsak to separate strukturer i samme emnekart. Det vil si at

når man klikket på et emne kom man til den strukturen som dette emnet opprinnelig hørte til, enten det var FRBR-modellen eller DC.

Et emne var likt og det var «title». «title» er et av de 15 elementene i DC, og et vanlig emne i DC-strukturen, mens i FRBR-emnekartet fremsto «title» som en intern forekomst. Denne koblingen ble derfor ikke god. I FRBR-modellen forteller «title» hvilke rolle de ulike personene har: sanger, komponist osv., og er kun en forklarende forekomst. Mens i DC-emnekartet inneholder «title», titler på manifestasjonene. Ved navnebasert sammenslåing ble likevel disse to emnetypene slått sammen. Titlene på tidsskriftene fra DC-emnekartet ble vist som emner, og deres tilhørende forekomster ble en oppramsing av personene fra FRBR-modellen med tilhørende titler (sanger, komponist osv.). Tidsskriftene og personene hadde ingen ting med hverandre å gjøre siden de kom fra forskjellige emnekart med helt forskjellig innhold, så denne sammenslåingen ble helt feil. Dette er et moment man må være klar over ved navnebasert sammenfletting.

For å lage bedre struktur på det sammenslåtte emnekartet må man forhindre at elementer i forskjellige emnekart som ikke har noe med hverandre å gjøre, blir slått sammen. For å få til dette må man ta i bruk perspektiv. Ved å angi hvilket perspektiv de ulike emnene hører til under, slipper man at emner som beskriver helt forskjellige ting blir slått sammen.

Som vi ser i figur 7.7, så inneholder begge de to emnekartene tittelen «You´ve changed». Disse vil ikke bli slått sammen selv om de er identiske og begge er tittelen på samme sang, så lenge det overordnede perspektivet er forskjellig. I dette tilfellet bør man sørge for at perspektivet er likt, mens i det første eksempelet hvor tittelen gjaldt helt forskjellige ting (tittel som forekomst og tittel for manifestasjon), bør man sørge for at perspektivet er forskjellig så de ikke blir slått sammen.



Figur 7.7: Perspektiv

7.3.2 Sammenfletting av emnekart basert på Dublin Core, Dewey og FRBR

Hva skal testes

Vi skal nå gå et steg videre å legge sammen alle de ulike emnekartstrukturene som er blitt gjennomgått i oppgaven. Det interessante er å se om det kan ha noen hensikt å slå sammen emnekart basert på helt forskjellige strukturer.

Hva testes det på

Her blir testcasene fra sammenslåingen av FRBR-modellen og DC benyttet, i tillegg til testcaset for DDC. Figur 7.8 på side 84, viser alle emnene fra de tre emnekartene etter sammenslåing.

Resultater

Ved sammenslåingen mistet flere av emnekartene den opprinnelige strukturen. Og koblinger som fantes tidligere ved mindre sammenslåinger ble borte. DDC-emnekartet mistet den hierarkiske strukturen, og ble vist likt som de andre ikke-hierarkiske emnekartene. «title»-feltet som ble slått sammen ved sammenslåing mellom FRBR-modellen og DC, ble nå ikke slått sammen.

Alle de ulike emnekartene ble stående separat hver for seg, selv om de var i det samme emnekartet. Strukturene ble ikke flettet sammen, i tillegg til at strukturen innad i de opprinnelige emnekartene ble dårligere.

Det kan derfor konkluderes med at det er lite hensiktsmessig å slå sammen emnekart basert på forskjellig strukturer. Det blir uoversiktlig og med liten nytteverdi. Men det kan i stedet sees mer på hvordan det kan benyttes en felles måte å implementere på. Denne oppgaven var bare et lite innspill i debatten.

Topic Types (36)

- (600) Teknologi (anvendt vitenskap)
- (700) Kunst og underholdning
- (750) Malerkunst og malerier
- (760) Grafiske kunstarter Grafikkframstilling og grafikk
- (770) Fotografering og fotografier
- (780) Musikk
- (782) Vokalmusikk
- (782.1) Opera
- (782.1092) Europeisk opera
- (782.6) Damestemmer
- (782.68) Altstemmer
- (784) Instrumenter og instrumentalensembler og musikk for disse
- (788) Blåseinstrumenter
- (790) Fritid, underholdning, sport
- (800) Litteratur og litteraturvitenskap
- Begrep
- Contributor
- DDC
- Description
- Hendelse
- Hierarchical relation type
- Identifier
- Korporasjon
- Language
- Manifestasjon
- Objekt
- Person
- Record
- Rights
- Sted
- Subject
- Subordinate role type
- Superordinate role type
- Title
- Uttrykk
- Verk

Figur 7.8: Emnene for alle de sammenslåtte emnekartene

Kapittel 8

Oppsummering og konklusjon

8.1 Oppsummering av arbeidet

Oppgaven har tatt for seg hvordan man kan ta i bruk metadatastandarder, klassifikasjonssystemer og konseptuelle referansemodeller i forbindelse med emnekart. Det ble først gjort en gjennomgang av teorien for utvalgte metadatastandarder, systemer og modeller. Deretter fulgte et teorikapittel om emnekart. Teorien ble videre tatt i bruk ved at det ble laget emnekart for metadatastandarden Dublin Core, klassifikasjonssystemet Dewey og den konseptuelle referansemodellen FRBR. Det ble på bakgrunn av disse implementasjonene sett hvordan slike faste strukturer kunne fungere i forbindelse med emnekart.

Til slutt i oppgaven ble det undersøkt hvordan slike faste strukturer kan være til nytte ved sammenfletting av flere emnekart. Det ble gjort sammenfletting av emnekart som var bygget opp på samme struktur i tillegg til sammenflettinger på tvers av strukturer.

8.2 Resultater

Dublin Core er en settorientert metadatastandard som ikke er basert på nettverkstankegangen slik emnekartstandarden er. Det var derfor nødvendig å legge til egenkomponerte assosiasjoner for at Dublin Core skulle fungere i kombinasjon med emnekart. Dublin Core kan derfor ikke benyttes uredigert som standard, og det er naturlig å tenke at det er bedre å lage emnekart bygget på formater som har en mer relasjonsorientert tenkemåte, og som er bedre tilpasset emnekartstandarden.

Dewey er et klassifikasjonssystem som tar utgangspunkt i fag/emner, og her er emnene i størst grad satt i sammenheng ved hierarkier. Emnekart har mulighet for implementasjon av hierarkier, og strukturen i Dewey kan bestå uendret ved en implementasjon. Men den utpregede nettverksstrukturen til emnekart ble begrenset ved bruk av Dewey.

FRBR-modellen er en kompleks modell, men har komponenter som er nært knyttet opp mot komponentene i emnekartstandarden. Entitetene kunne implementeres som emner, relasjonene som assosiasjoner og attributtene som interne forekomster. Det var derfor relativt uproblematisk å benytte emnekart og FRBR-modellen i kombinasjon. Det var derfor ikke nødvendig å avvike fra FRBR-modellen og struktur og komponenter kunne opprettholdes slik det var tiltenkt fra standarden i utgangspunktet, og data slapp å gå tapt.

Ved sammenfletting av like standarder kom det frem at strukturen i emnekartet ble beholdt på

en god måte. Ved sammenfletting av emnekart på kryss av standardene derimot, ble strukturen i liten grad overlappende, og det ble i praksis to separate emnekart som hang sammen i noen få ledd som besto av like emner i de to emnekartene.

Ved sammenslåingen av emnekart på kryss av strukturene, mistet flere av emnekartene den opprinnelige strukturen. Koblinger som fantes tidligere ved mindre sammenslåinger ble borte. DDC-emnekartet mistet den hierarkiske strukturen, og ble vist likt som de andre ikke-hierarkiske emnekartene. Title»-feltet som ble slått sammen ved sammenfletting mellom FRBR-modellen og DC, ble nå ikke slått sammen.

Alle de ulike emnekartene ble stående separat hver for seg, selv om de var i det samme emnekartet. Strukturene ble ikke flettet sammen, i tillegg til at strukturen innad i de opprinnelige emnekartene ble dårligere.

Det er derfor lite hensiktsmessig å slå sammen emnekart basert på forskjellig strukturer. Det blir uoversiktlig og med liten nytteverdi. Sammenfletting av like standarder derimot bør sees mer på. En felles struktur for implementasjon kan være til hjelp for sammenflettinger av emnekart ved at man har en felles struktur på emnekartene som blir bevart også etter sammenfletting.

8.3 Evaluering av arbeidet

Det er i denne oppgaven sett på hvordan Dublin Core, Dewey og FRBR-modellen var egnede til å bli implementert i emnekartstandarden. Ved disse tre strukturene var det mulig å få frem forskjellige aspekter, siden det gjaldt et metadataformat, et klassifikasjonssystem og en modell. Men det er likevel aktuelt i videre arbeider å se på flere typer formater for å få ennå større bredde på testresultatene.

Det ble laget dobbelt med emnekart for hver struktur, slik at to emnekart med samme struktur kunne sammenflettes. Disse emnekartene var relativt like hverandre siden det ekstra emnekartet for hver struktur var et subsett fra det opprinnelige. Sammenflettingen gikk derfor kanskje bedre enn om de hadde hatt totalt forskjellig innhold. Det hadde også antageligvis kommet tydeligere frem ulike måter å implementere strukturen på hvis emnekartene var laget av forskjellige personer, og med forskjellig oppbygging ville heller ikke sammenflettingen av like standarder gått like bra. Dette hadde særlig gjort utslag hvis personene var fra forskjellige land og benyttet forskjellig språk, versjoner av standardene osv.

For emnekartene er det implementert PSI-er for emnetypene. For Dublin Core, ble PSI-ene som Ontopia har publisert benyttet, mens det for FRBR-modellen og Dewey, ble laget egne PSI-er som ble publisert på en nettside. Denne nettsiden var på mitt hjemmeområde på skolen, og PSI-ene er derfor ikke publisert eller lagt vekt på med tanke på at andre skal benytte dem, og ble kun laget for testens skyld. I videre utvikling, kunne PSI-ene blitt lagt mer vekt på.

8.4 Videre arbeid

Det kan være aktuelt å se på flere formater for å få større bredde.

Det er også interessant å se mer på sammenflettingsbiten. Både med tanke på like standarder og med tanke på bruk av PSI-er.

Brukervennligheten er ikke diskutert i større grad i denne oppgaven, og er et viktig moment for videre utvikling.

Et annet tema som kan diskuteres videre er hvordan språkforskjeller kan takles. Hvordan blir det i praksis ved sammenslåing av emnekart basert på forskjellige språk?

Emnekart er en relativt ny standard som stadig blir tatt mer i bruk i forskjellige sammenhenger, og utvikles deretter. Det vil derfor hele tiden dukke opp nye temaer i forbindelse med emnekart som det vil være aktuelt å se på.

Bibliografi

[1] Lars Marius Garshol. *Metadata? Thesauri? Taxonomies? Topic Maps!* Ontopia, http://www.ontopia.net/topicmaps/materials/tm-vs-thesauri.html, Oktober 2004. Sist sett: 05.11.2005.

[2] Trond Aalberg Knut Hegna. *Arkitektur for digitale bibliotek*. BIBSYS, http://www.bibsys.no/BDB/arkitektur/ArkDigBib.pdf, 2000. ISBN 82-7729-027-6.

[3] Steve Pepper. *Published Subjects: Introduction and Basic Requirements*. OASIS Published Subjects Technical Committee Recommendation, http://www.ontopia.net/tmp/pubsubj-gentle-intro.htm, Juni 2003. Sist sett: 19.10.05.

[4] Ole Husby. *Metadata. Foredrag ved Kunnskapsorganisasjonsdagene 1997, Høgskolen i Oslo*. BIBSYS, http://www.bibsys.no/meta/korg97.html, 1997. Sist sett: 23.11.2005.

[5] Standardiseringen i norge. *Enklere å finne musikk på nettet*. http://www.standard.no/imaker.exe?id=615:. Sist sett 07.11.2005.

[6] NTNU, http://www.idi.ntnu.no/emner/it2802/forelesingsnotater/. *Metadata*, oktober 2005. Sist sett: 07.11.2005.

[7] Diane Hillmann. *Using Dublin Core*. Dublin Core Metadata Initiative, http://dublincore.org/documents/usageguide, August 2003. Sist sett: 23.11.2005.

[8] Dublin Core Norge, http://www.dublincore.no. *Hva er Dublin Core*, Juni 2001. Sist sett: 08.08.05.

[9] Ole Husby og Knut Hegna. *Dublin Core Metadata Element Set: Norsk oversettelse av referansedokument*. BIBSYS, http://heim.ifi.uio.no/ knuthe/dok/DCref.html, Juli 1999. Sist sett: 08.08.05.

[10] Andy Powell. *Expressing Dublin Core in HTML/XHTML meta and link elements*. The Dublin Core Metadata Initiative, http://dublincore.org/documents/dcq-html/, November 2003. Sist sett: 08.08.05.

[11] Melvil Dewey. *Deweys desimalklassifikasjon*. Nasjonalbiblioteket, ISBN 82-7965-059-8, 5.norske forkortede utgave ved isabella kubosch. basert på ed.21 edition, 2002.

[12] Nasjonalbiblioteket, http://www.nb.no. *Pressemelding: Viktig begivenhet for norske bibliotek: Lansering av ny norsk Dewey-utgave*, Juni 2004. Sist sett: 30.06.05.

[13] Lisbeth Eriksen. Fra tesaurus til ontologi - fra agrovoc til agricultural ontology service (aos).

[14] ICOM, http://cidoc.ics.forth.gr/. *The CIDOC Conceptual Reference Model*, Mai 2004. Sist sett: 05.11.2005.

[15] Jane Hunter Carl Lagoze. *The ABC Ontology and Model*. http://jodi.ecs.soton.ac.uk/Articles/v02/i02/Lagoze/. Sist sett: 23.11.2005.

[16] International Federation of Library Associations and Institutions (IFLA), http://www.ifla.org/VII/s13/frbr/frbr.pdf. *Functional requirements for bibliographic records*, final report edition, 1998.

[17] Oversatt av: Liv Aasa Holm. *Funksjonskrav til bibliagrafiske poster*. International Federation of Library Associations and Institutions (IFLA), http://www.nb.no/katkom/frbr/4nbmkap1.htm, norsk utgave edition, Februar 2001. Utgitt av: Nationalbiblioteket.

[18] Carol van Nuys. Funksjonskrav til bibliografiske poster. Presentasjon, Nationalbiblioteket, http://www.jbi.hio.no/bibin/nolug/FRBR.ppt, 13.12 2002. Sist sett: 30.09.2004.

[19] Trond Aalberg. *Identifikatorer*. NTNU, Februar 2004. Forelesningsfoil for IT3803.

[20] ISWC Administrator. *ISO 15707:2001. Information and documentation - International Standard Musical Work Code (ISWC)*. ISO/TC 46/SC 9, http://www.collectionscanada.ca/iso/tc46sc9/standard/15707e.htm, 24.02 2004. Sist sett: 17.11.2004.

[21] Steve Pepper. *The TAO of Topic Maps*. Ontopia AS, http://www.ontopia.net/topicmaps/materials/tao.html, April 2002. Sist sett: 23.11.2005.

[22] Ingvild Kongsbakk. *Sømløs kunnskap: Om bruk av emnekart*. ABM-utvikling, http://www.abm-utvikling.no/publisert/ABM-skrift/2004/emnekart.pdf, 2004. Først utgitt: Diplomoppgave ved Høgskolen i Oslo, Avdeling for journalistikk, bibliotek- og informasjons-fag, Oslo 2003.

[23] Steve Newcomb Michel Biezunski, Martin Bryan. *ISO/IEC FCD 13250:1999 - Topic Maps*. http://www.y12.doe.gov/sgml/sc34/document/0058.htm, April 1999. Sist sett 19.10.05.

[24] William Arms. *The online edition of Digital Libraries*. http://www.cs.cornell.edu/wya/DigLib/new/glossary.html, 2000. Sist sett: 23.07.2005.

[25] ISWC Administrator. *ISWCNET*. ISWC International Agensy, http://www.iswc.org/iswc/en/html/ISWCdbEN.html, november 2004. ISWC-database. Sist sett: 21.11.04.

[26] International ISRC Agency. *ISO 3901:2001. Information and documentation - International Standard Recording Code (ISRC)*. ISO TC 46/SC 9, http://www.collectionscanada.ca/iso/tc46sc9/standard/3901e.htm, 02.04 2002. Sist sett: 21.11.04.

[27] Members of the TopicMaps.Org Authoring Group;. *XML Topic Maps (XTM) 1.0, TopicMaps.Org Specification*. http://www.topicmaps.org/xtm/1.0/, 1.16 edition, August 2001. Sist sett:14.08.05.

[28] Steve Pepper. *Sømløs kunnskap for offentlig og privat*. Ontopia, http://66.249.93.104/search?q=cache:6Ul7z3NMai8J:www.emnekart.no/2004/konferanse, Oktober 2004. Sist sett: 27.11.05.

[29] Steve Pepper. *Sømløs kunnskap. Emnekart som portalløsning*, 16.03 2004. Emnekartsemminar for ITEA.

Tillegg A

Deweyklasser

Deweys hovedklasser og nivået under: [11]

000 Generelle emner

010 Bibliografi

020 Bibliotek- og informasjonsvitenskap

030 Generelle encyklopedier og leksika

050 Generelle periodika

060 Generelle organisasjoner og museumskunnskap (museologi)

070 Dokumentarmedier, undervisningsmedier, nyhetsmedier; journalistikk; publisering

080 Generelle samlinger

090 Håndskrifter(manuskripter), sjeldne bøker, annet sjeldent, trykte materiale

100 Filosofi, overnaturlige fenomener, psykologi

110 Metafysikk

120 Erkjennelsesteori, årsakssammenheng, mennesket

130 Overnaturlige fenomener

140 Bestemte filosofiske skoler og retninger

150 Psykologi

160 Logikk

170 Etikk (Moralfilosofi)

180 Oldtidens og middelalderens filosofi, orientalsk filosofi

190 Vestens filosofi i nyere tid og annen ikke-orientalsk filosofi

200 Religion

210 Filosofi og teori innen religion

220 Bibelen

230 Kristendom Kristen teologi

240 Kristen etikk (moral)og oppbyggelig teologi

250 Den lokale kirke og kristne religiøse ordener

260 Kristen sosialteologi og kirketeologi

270 Historisk eller geografisk behandling, personer knyttet til kristendom Kirkehistorie

280 Trossamfunn og sekter i den kristne kirke

290 Sammenlignende religionsvitenskap og andre religioner enn kristendom

300 Samfunnsvitenskap

310 Generelle statistikksamlinger

320 Statsvitenskap

330 Økonomi

340 Rettsvitenskap

350 Offentlig administrasjon og militærvesen

360 Sosiale problemer og tjenester;foreninger

370 Utdanning og pedagogikk

380 Handel, kommunikasjon, samferdsel.

390 Skikker, etikette, folkeminne

400 Språk og språkvitenskap

410 Språkvitenskap (lingvistikk)

420 Engelsk og gammelengelsk (angelsaksisk)

430 Germanske språk Tysk

440 Romanske språk Fransk

450 Italiensk, sardinsk, dalmatisk, rumensk, retoromanske språk

460 Spansk og portugisisk

470 Italiensk språk Latin

480 Greske språk Klassisk gresk

490 Andre språk.

500 Naturvitenskap og matematikk

510 Matematikk

520 Astronomi og lignende

530 Fysikk

540 Kjemi og lignende vitenskaper

550 Geovitenskaplige fag

560 Paleontologi Paleozoologi

570 Biologiske fag Biologi

580 Planter

590 Dyr

600 Teknologi (anvendt vitenskap)

610 Legevitenskap (medisin)

620 Teknikk (ingeniørfag) og lignende fagområder

630 Landbruk og lignende fagområder

640 Husholding og familieliv

650 Administrasjon og ledelse

660 Kjemiteknikk og lignende teknologi

670 Industriell produksjon

680 Produksjon av produkter med bestemte bruksområder

690 Husbygging

700 Kunst og underholdning

710 Arealplanlegging og landskapsarkitektur

720 Arkitektur

730 Plastisk kunst Skulptur

740 Tegnekunst og kunsthåndverk

750 Malerkunst og malerier

760 Grafiske kunstarter Grafikkfremstilling og grafikk

770 Fotografering og fotografier

780 Musikk

790 Fritid, underholdning, sport

800 Litteratur og litteraturvitenskap

810 Amerikansk litteratur på engelsk

820 Engelsk og gammelengelsk (angelsaksisk) litteratur

830 Germanske språks litteraturer Tysk litteratur

840 Romanske språks litteraturer Fransk litteratur

850 Italiensk, sardinsk, dalmatisk, rumensk, retoromanske språks litteraturer

860 Spansk og portugisisk litteratur

870 Italienske språks litteraturer Latinsk litteratur

880 Greske språks litteraturer Klassisk gresk litteratur

890 Andre bestemte språks og språkgruppers litteraturer

900 Geografi, historie og deres hjelpefag

910 Geografi og reiser

920 Biografi, genealogi, insignier

930 Oldtidens historie (til ca. 499)

940 Europas historie Vest-Europa

950 Asias historie Orienten Fjerne Østen

960 Afrikas historie

970 Nord- og Mellom-Amerikas historie

980 Sør-Amerikas historie

990 Andre deler av verden og himmellegemer utenfor jorda Stillehavsøyene

Tillegg B

Emnekart basert på Dublin Core

Emnetyper, assosiasjonstyper og assosiasjonsrolletyper for emnekartet basert på Dublin Core:

Figur B.1: Emnekart basert på Dublin Core

Tillegg C

Emnekart basert på Dewey

Emnetyper, assosiasjonstyper, assosiasjonsrolletyper og forekomsttyper for emnekartet basert på Dewey:

Topic Types (19)

- (600) Teknologi (anvendt vitenskap)
- (700) Kunst og underholdning
- (750) Malerkunst og malerier
- (760) Grafiske kunstarter Grafikkframstilling og grafikk
- (770) Fotografering og fotografier
- (780) Musikk
- (782) Vokalmusikk
- (782.1) Opera
- (782.1092) Europeisk opera
- (782.6) Damestemmer
- (782.68) Altstemmer
- (784) Instrumenter og instrumentalensembler og musikk for disse
- (788) Blåseinstrumenter
- (790) Fritid, underholdning, sport
- (800) Litteratur og litteraturvitenskap
- DDC
- Hierarchical relation type
- Subordinate role type
- Superordinate role type

Association Types (1)

- Supertype-subtype

Association Role Types (2)

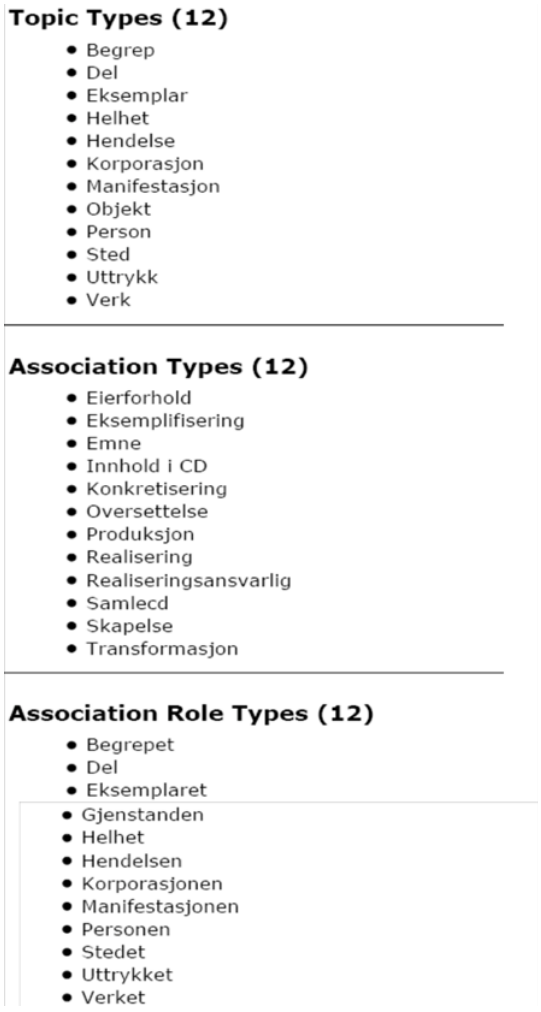
- Subtype
- Supertype

Figur C.1: Emnekart basert på Dewey

Tillegg D

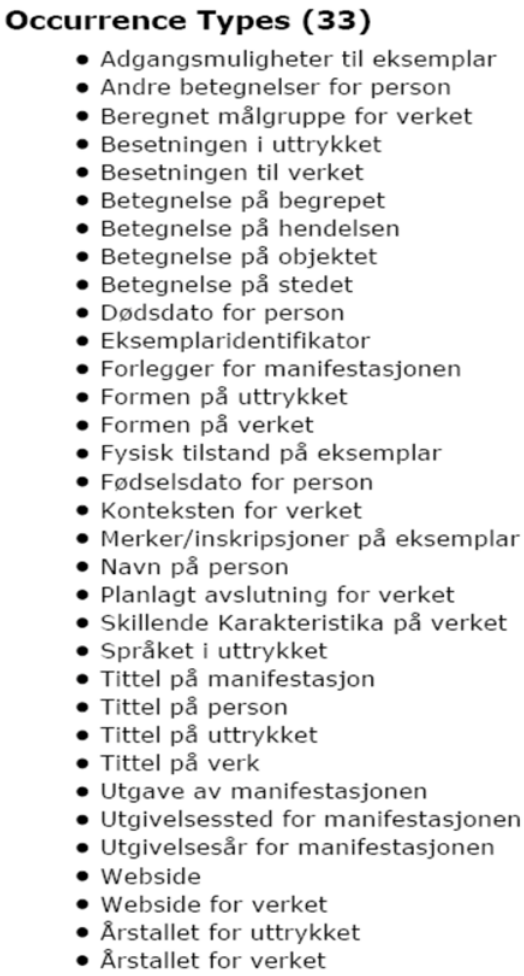
Emnekart basert på FRBR-modellen

Emnetyper, assosiasjonstyper og assosiasjonsrolletyper for emnekartet basert på FRBR-modellen:



Figur D.1: Emnekart basert på FRBR-modellen

Forekomsttyper for emnekartet basert på FRBR-modellen:



Figur D.2: Forekomster for emnekart basert på FRBR-modellen