Fra Informationsmodel til en DTD*

Steffen Leo Hansen

This work is produced by The Connexions Project and licensed under the
Creative Commons Attribution License †

Abstract

En praf opbygning og brug af en DTD med udgangspunkt i en informationsmodel

1 DTD – DOCUMENT TYPE DEFINITION

1.1 1. Dokument – informationsmodel – DTD

Udgangspunkt for fremstillingen i dette modul er det **XML-dokument** som er vist i fig.1 nedenfor. Vi vil gerne sikre os, at dokumenten er både *velformet* og *gyldigt*. **Velformet** er dokumentet, når det overholder de syntaksregler som gfor XML [1]. At et XML-dokument er **gyldigt**, vil sige at det er i overensstemmelse med en foreliggende standard for opm, den standard som kaldes en **DTD, en Document Type Definition**.

En DTD fastl

- hvilke elementer der forekommer i et XML-dokument
- hvor mange gange det enkelte element forekommer
- i hvilken rølge elementerne forekommer

I det følgende vil vi (1) anskue en DTD som en implementeret **informationsmodel**, dern(2) se npå reglerne for hvordan man skal bygge og notere informationerne i en DTD, og endelig (3) se npå hvordan en DTD bruges sammen med et XML-dokument.

1.2 2. Fra dokument til informationsmodel

1.2.1 2.1 Opm

Udgangspunkt er opmaf nedenstående lydbog (ex1):

*Version 1.4: Feb 9, 2009 6:13 am -0600

†http://creativecommons.org/licenses/by/3.0/

[1] jf. http://www.w3.org/TR/xml/ (<http://www.w3.org/TR/xml/>)

```
<online_katalog>
  <bog>
    <forfatter nation="dk">
      <fornavn>Jørgen</fornavn>
      <efternavn>Leth</efternavn>
    </forfatter>
    <impressum>
      <titel>Det uperfekte menneske</titel>
      <forlag sted="København">Gyldendal</forlag>
      <udgivet>2007</udgivet>
      <ISBN format="lydbog" volumen="11T:36M">978-87-02-06068-3</ISBN>
      <pris>kr. 298</pris>
    </impressum>
    <stof>
      <handling>Cool, elegante, morsomme og selvudslettende
                historier fra et uperfekt liv med litteratur,
                film, sport, kvinder og rejser. Digteren,
                filminstruktøren og Tour de France-kommentatoren
                m.m. har skrevet en erindringsbog af de sj,
                som samtidig tegner et portr af dansk
                kulturliv gennem de sidste 40 år. Jørgen Leths
                karakteristiske stemme fører lytteren gennem en
                af dette årtis mest omtalte bøger.</handling>
      <genre>biografi</genre>
      <original_titel> Det uperfekte Menneske </original_titel>
    </stof>
  </bog>
</online_katalog>
```


Figur 1

Denne opmvil vi gerne bruge som standard ved opmaf bogtitler. Det er derfor vigtigt at give en beskrivelse af denne standard således at alle nye og tilkomne titler følger samme opm. Det krat der udarbejdes en oversigt over hvilke elementer og attributter der indgår i opm. En sådan oversigt kaldes også for en **informationsmodel** og har vi først uadarbejdet en informationsmodel, kan vi bruge denne som grundlag for at udvikle en DTD.

1.2.2 2.2 Informationsmodellen

Informationsmodellen kommer til at se således ud (ex2):

```
1. BOG prototypen -- strukturelement, container for den samlede beskrivelse af en bog

  1.1 FORFATTER strukturelement; indgang til informationer om den
      aktuelle forfatter

      ATTRIBUT: nation: angiver forfatters nationalitet

      1.1.1 FORNAVN tekstelement; indeholder forfatterens fornavn

      1.1.2 EFTERNAVN tekstelement; indeholder forfatterens efternavn

  1.2 IMPRESSUM strukturelement; indgang til alle informationer
      om bogens udgivelse

      1.2.1 TITEL tekstelement; indeholder bogens titel

      1.2.2 FORLAG tekstelement; indeholder navnet på forlaget

            ATTRIBUT: sted: angiver hjemsted for forlaget

      1.2.3 UDGIVET tekstelement; indeholder året for udgivelsen

      1.2.4 ISBN tekstelement; indeholder bogens ISBN-nummer

            ATTRIBUT: format: angiver udgivelsens format:
            ''heftet'', ''indbundet'' eller ''lydbog''

            ATTRIBUT: volumen: angiver sidetal eller varighed
            for en lydbog

      1.2.5 PRIS tekstelement; indeholder indkøbspris

  1.3 STOF strukturelement; indgang til informationer om en
      bogs handling, genre og orignaltitel

      1.3.1 HANDLING tekstelement; indholder et indholdsresume,
            typisk svarende til indholdet angivet på bogens bagside

      1.3.2 GENRE tekstelement; indeholder angivelse af genre

      1.3.3 ORIGINAL_TITEL tekstelement; indeholder angivelse
            af originaltitel, er specielt relevant for oversat litteratur
```

Figure 2

Med denne oversigt, det vi kalder en formaliseret informationsmodel, har vi dannet os et samlet overblik over elementerne i modellen, over relationerne mellem elementerne, og over mulige attributter i et standard-dokument. Enhver ny bog som skal indgå i ONLINE_KATALOG skal beskrives og opmsom XML-dokument i henhold til modellen i ex2.

Nskridt bliver at implementere beskrivelsen i form af en DTD som vi kan bruge til at **validere** alle nye beskrivelser. At validere et dokument vil sige at fastlom der er tale om et gyldigt XML-dokument, et dokument som lever op til standarden.

1.3 3. DTD: en implementeret standard for opm

En DTD, en *Document Type Definition*, er en standard, en **deklaration**, som angiver *hvilke navngivne elementer* der kan forekomme i en bestemt type XML-dokumenter, *hvor mange gange* elementerne forekommer, og *i hvilken rølge*. På denne måde deklarerer og standardiserer en DTD et s**vokabular**, maf navngivne elementer og eventueller attributter, og en sstruktur, relationen mellem elementer, for forekomster af en type XML-dokumenter.

1.3.1 3.1 Syntaks

Der gen ssyntaks for opbygningen af en DTD (jf. http://www.w3.org/TR/2008/REC-xml-20081126[2]):

1. Alle elementer i en DTD beskrives inden for tegnene < og >, samme notation som gfor angivelsen af tags i XML.
2. Med tegnet '!ELEMENT' angives at der er tale om en PI.
3. Betegnelsen ELEMENT er et reserveret ord i XML, forbeholdt denne funktion i en DTD.

En DTD indeholder en beskrivelse, en **elementdeklaration**, af alle elementer i et tilhørende XML-dokument efter følgende skabelon:

<!ELEMENT elementnavn (indhold) >

Som det første i en elementdeklaration anføres navnet på det element som skal deklareres. Lmtil at navn og ortografi for hvert enkelt element i en DTD skal matche den tilsvarende angivelser i XML-dokumentet og vice versa. Efter elementets navn anføres i parentes elementets indhold. Ved at se på indholdet i parentesen kan man konstatere om der er tale om et strukturelement eller et tekstelement.

Det vil blive illustreret med følgende eksempler:

3.1.1 Rodelementet

<!ELEMENT online_katalog ((bog)+) >

Der findes ikke specielle regler for rølgen af elementdeklarationer i en DTD. Men det kan anbefales at følge rølgen i den formaliserede informationsmodel. Denne rølge svarer til rølgen i det tilhørende XML-dokument, og ved at følge denne er risikoen for at glemme eller overse et element ubetydelig.

Det første element som skal deklareres er derfor rodelementet. Navnet på elementet skal anføres og dern, anført i parentes, den eller de knuder i strukturen som er indlejret i rodelementet.

I det anførte eksempel er der kun ét indlejret element: elementet BOG. Der er mindst én forekomst, men forhåbentlig mange flere, af dette element. Det angives med en **hyppighedsoperator**: + som betyder: én eller flere forekomster af det element som står umiddelbart foran. Ved at bruge operatorer for hyppighed, kan man angive hvor mange gange et element skal forekomme. Står der ikke en operator efter et element, forekommer det én og kun én gang.

Vi kan lnotationen i fig.3 på følgende måde: elementet ONLINE_KATALOG er for(eng. *parent*) til BOG, som er barn af (eng. *child*) ONLINE_KATALOG. BOG forekommer én eller flere gange.

3.1.2 Strukturelementet

Elementet BOG er et strukturelement. Det fremgår af indholdet i parentesen som kun indeholder elementer. Det deklareres i princippet efter samme opskrift som ovenfor:

[2] http://www.w3.org/TR/2008/REC-xml-20081126

<!ELEMENT bog (forfatter, impressum, stof) >

Dette element har som vist tre indlejrede elementer, de elementer som er anført i parentesen. Der er i dette eksempel ikke nogen hyppighedsoperator anført for disse elementer, og det betyder at de forekommer én – og kun én gang. Rølgen af elementer i parentesen foreskriver rølgen af elementer i XML-dokumentet. Vi kan lnotationen på følgende måde: elementet BOG er fortil børnene FORFATTER, IMPRESSUM og STOF. Disse tre elementer er børn af BOG og de er tillige søskende (eng. *siblings*).

3.1.3 Tekstelementet

Som eksempel på et tekstelement kan vi bruge elementet FORNAVN:

<! ELEMENT fornavn (#PCDATA) >

Deklarationen siger i dette eksempel at elementet FORNAVN indeholder #PCDATA. Det betyder at indholdet i elementet er ren tekst. **PCDATA** betyder *Parsed Character Data*, data som skal undersøges n, parses, med henblik på at sikre at der ikke forekommer ulovlige tegn i teksten, det vil sige tegn som indgår i XMLs standardvokabular.

3.1.4 Attributter

Der gen ssyntaks for angivelse af attributter i en DTD. De bygges over følgende skabelon:

<!ATTLIST *elementnavn attributnavn typevforekomst* **>**

Et eller flere attributter anføres i en **attributliste**: ATTLIST. Som det første angives hvilket element attributtet er associeret med, dernnavnet på attributtet, hvilken type der optrsom vfor attributtet, og endelig en karakteristik af forekomsten, om der for eksempel er tale om et obligatorisk eller optionelt attribut.

Elementet FORFATTER indeholder et attribut:

<!ELEMENT forfatter (fornavn, efternavn) >
<!ATTLIST forfatter
nation CDATA #REQUIRED
>

FORFATTER er fortil FORNAVN og EFTERNAVN, som dermed også er søskende. FORFATTER er med andre ord et *strukturelement*. Men derudover kan vi se at der er et attribut knyttet til elementet. Det angives med en attributliste: ATTLIST – det kan jo tat der er mere end et attribut – efterfulgt af navnet på det element som indeholder et attribut efterfulgt af navnet på selve attributtet. Af ovenstående fremgår altså at elementet FORFATTER har et attribut kaldet NATION.

Herefter skal der følge yderligere to angivelser: en angivelse af hvilken type oplysning der kan forekomme som vfor attributtet, og en angivelse af hvornår attributtets vskal vudfyldt.

I det aktuelle eksempel: **nation CDATA #REQUIRED,** står der umiddelbart efter attributtets navn som angivelse af typev: **CDATA**. Denne oplysning betyder *Character Data*. Det angiver at der er tale om data som – i modstil PCDATA – ikke skal parses af XML-processoren (hvad det vil sige, vender vi tilbage til i modul 4). Som sidste oplysning finder vi angivelsen: **#REQUIRED**. Det betyder at der *skal* ven vangivet for dette attribut, ver med andre ord obligatorisk, og der kan ikke forekomme en forfatter i vores onlinekatalog uden at der samtidig er en oplysning om vedkommendes nationalitet.

Et eksempel mere:

<!ELEMENT ISBN (#PCDATA) >
<!ATTLIST ISBN
format CDATA (tryk | lydbog) "tryk"
volumen CDATA #REQUIRED
>

Elementet ISBN er et tekstelement. Det er angivet som: ISBN (#PCDATA), hvor PCDATA betegner data som parses, nemlig tekststrengen som angiver ISBN nummeret. Der må med andre ord ikke vtegn indeholdt i denne streng som er reserveret til brug i XML.

Elementet ISBN har desuden, som det fremgår af ATTLIST, to attributter, attributterne FORMAT og VOLUMEN. For begges vedkommende gat typever defineret til at vcharacter data. Men der er en vforskel i angivelsen af vforekomst. Som det umiddelbart fremgår, skal der anføres en vfor attributet VOLUMEN – i XML-dokumentet vil det ven angivelse af enten et sidetal eller en lydbogs varighed. Denne ver obligatorisk, hvad der fremgår af notationen #REQUIRED.

Vfor attributet FORMAT er derimod angivet som en oplistning af mulige v(eng. *enumerated type values*): **format CDATA (tryk | lydbog) "tryk"**

Vkan venten *tryk* eller *lydbog* svarende til angivelsen i parentesen: (tryk | lydbog), altså en bog i enten papirformat eller en lydbog. Endvidere er der angivet en default-v, det vil sige en vsom automatisk vil g, hvis opmhar glemt at anføre en v. Default er tryk, hvad der nok er det mest almindelige, trods alt.

Vi kan konstatere at vi nu er i stand til at bygge en DTD med baggrund i en formaliseret informationsmodel.

Den samlede DTD ser ud som nedenfor vist i (ex3):

```
<!ELEMENT online_katalog ((bog+))>
<!ELEMENT bog ((forfatter, impressum, stof))>
<!ELEMENT forfatter ((fornavn, efternavn))>
<!ATTLIST forfatter
nation CDATA #REQUIRED
>
<!ELEMENT fornavn (#PCDATA)>
<!ELEMENT efternavn (#PCDATA)>
<!ELEMENT impressum ((titel, forlag, udgivet, ISBN, pris))>
<!ELEMENT titel (#PCDATA)>
<!ELEMENT forlag (#PCDATA)>
<!ATTLIST forlag
sted CDATA #REQUIRED
>
<!ELEMENT udgivet (#PCDATA)>
<!ELEMENT ISBN (#PCDATA)>
<!ATTLIST ISBN
format (tryk | lydbog) "tryk"
volumen CDATA #REQUIRED
>
<!ELEMENT pris (#PCDATA)>
<!ELEMENT stof ((handling, genre, original_titel))>
<!ELEMENT handling (#PCDATA)>
<!ELEMENT genre (#PCDATA)>
<!ELEMENT original_titel (#PCDATA)>
```

Figur 3

1.3.2 4. Validering af et XML-dokument med en DTD

Som det sidste skal vi se på hvordan man kan skabe en forbindelse mellem en DTD og et tilhørende XML-dokument således at det bliver muligt at validere opmi dokumentet.

Et XML-dokument er gyldigt hvis det matcher beskrivelsen af elementstruktur og indhold i en tilhørende DTD. Det vil med andre ord sige, at elementernes navne matcher, at rølgen de forekommer i er den samme som beskrevet i DTD'en, og at de forekommer med samme hyppighed som angivet i DTD'en.

En DTD kan v**intern** i forhold til et XML-dokument eller den kan v**ekstern**.

1. En intern DTD er placeret øverst i XML-dokumentet, DTD og opmer i et og samme dokument eller i en og samme fil.
2. En ekstern DTD opbevares i et dokument for sig, i en sfil med efternavnet *.dtd, for eksempel online.dtd. Den DTD som er vist ovenfor i fig. 3 er et eksempel på en ekstern DTD.

Om man nu vat bruge en intern eller en ekstern DTD, afhaf mange ting. Vman en intern DTD, følges DTD og opmaltid ad. Det kan undertiden vpraktisk, specielt hvis der er tale om mindre XML-dokumenter som er afsluttede helheder og som ikke skal udbygges yderligere. Vman en ekstern DTD, opnår man mange fordele. Først og fremmest kan man distribuere sin DTD og dermed muliggøre at flere personer kan sidde samtidigt forskellige steder og gennemføre en ensartet og konsistent opmaf dokumenter, som afslutningsvis kan samles i ét større dokument.

Forbindelsen mellem et *XML-dokument* og *en ekstern DTD* etableres i den del af dokumentet som også kaldes for prologen. Det er her i et XML-dokument at man finder alle deklarationerne. Det kommer til at se således ud:

1: <?xml version="1.0" encoding="UTF-8"?>
2: <!DOCTYPE online_katalog
SYSTEM "d:\home\data\Altova Projects\online.dtd">

I første linie har vi den traditionelle XML-erkl. I linie 2 følger et navn samt en angivelse af at der er en DTD knyttet til dette XML-dokument og at denne ligger på systemniveau: SYSTEM, og at den findes på det sted som er angivet i den efterfølgende sti. Lmtil at denne deklaration følger den traditionelle PI-syntaks: <!.....> og at den indeholder to reserverede ord: DOCTYPE og SYSTEM.

Lendelig som det sidste mtil at vores DTD har et internt navn: **online_katalog**, det navn som står lige efter DOCTYPE, og at **dette navn er og skal videntisk med navnet på rodelementet i XML-dokumentet**.

Efter denne prolog følger resten af XML-dokumentet. Det kan nu udvides med flere forekomster af elementet BOG, og hver gang vi åbner dokumentet med en browser eller en anden form for XML-processor, bliver opmvalideret. Det kan gøres for hver enkelt opmdecentralt og for hele kataloget centralt. Derfor er det en god idé at bruge en ekstern DTD!

Det nuvdokument, klar til udvidelser og validering, ser således ud (ex.4):

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE online_katalog
SYSTEM "d:\home\data\Altova Projects\nynetbog_online.dtd">

<online_katalog>

  <bog>

    <forfatter nation="dk">

      <fornavn>Jørgen</fornavn>

      <efternavn>Leth</efternavn>

    </forfatter>

    <impressum>

      <titel>Det uperfekte menneske</titel>

      <forlag sted="København">Gyldendal</forlag>

      <udgivet>2007</udgivet>

      <ISBN format="lydbog" volumen="11T:36M">978-87-02-06068-3</ISBN>

      <pris>kr. 298</pris>

    </impressum>

    <stof>

      <handling>Cool, elegante, morsomme og selvudslettende
                     historier fra et uperfekt liv med litteratur,
                     film, sport, kvinder og rejser. Digteren,
                     filminstruktøren og Tour de France-kommentatoren
                     m.m. har skrevet en erindringsbog af de sj,
                     som samtidig tegner et portr af dansk
                     kulturliv gennem de sidste 40 år. Jørgen Leths
                     karakteristiske stemme fører lytteren gennem en
                     af dette årtis mest omtalte bøger.</handling>

      <genre>biografi</genre>

      <original_titel> Det uperfekte Menneske </original_titel>

    </stof>

  </bog>
```



```xml
</online_katalog>
```

Test dig selv

Vores XML-dokument,online_katalog.xml, og vores formaliserede informationsmodel repren og samme tr.

Hvad vil det sige at der er tale om en tr?

I hvilken forstand kan man sige at der er tale om en repraf en og samme tr?

Tegn den trsom er implementeret i de tre forskellige repr