Kompleks data og logik i databasen
Brug af XML dokumenter til en videoapplikation

Stefan Jaensch, Jørgen Bo Arp Ladekjær

Januar 2014 - Marts 2014

Indhold

Kapitel 1

Introduktion

Videoapplikationer som Netflix, Youbio og Viaplay er blevet meget populære i de seneste par år. Dette projekt handler om at skabe nogle de søgefunktioner, som der må være brug for i sådanne videoapplikationer, hvor et stort antal videoer er tilgængelige for brugeren. Disse søgefunktioner skal gøre det muligt for brugeren at navigere rundt i de mange videoer og finde netop det indhold eller den specifikke video, som måtte have interesse for brugeren.

Som datakilde til dette projekt er der valgt et åbent API (Application Programming Interface) fra Danmarks Radio, som stiller stort set alt deres egenproducerede indhold tilgængeligt for brugeren/programmøren via adressen http://www.dr.dk/nu/api . Dette API giver fx mulighed for at hente en total oversigt over alle tilgængelige videoer, i et enkelt XML dokument, samt hente på detaljer omkring specifikke videoer.

I dette projekt hentes disse XML dokumenter ud via en standard webbrowser og gemmes lokalt, hvorefter de indlæses i en BaseX XML database, hvorfra der vil bliver udarbejdet forskellige søgefunktioner til forespørgsler i XML dokumenterne.

I øjeblikket findes der desværre ingen tilgængelig beskrivelse af struktur eller elementer i de tilgængelige XML dokumenter, så derfor starter projektet med en analyse af netop strukturen eller elementerne i de XML dokumenter som er udvalgt til dette projekt.

Kapitel 2

Undersøgelse af XML dokumenterne

I det åbne API fra Danmarks Radio er der mange forskellige XML dokumenter til rådelighed. Nogle af XML dokumenter er forholdsvis små og indeholder kun en sti til en grafik, som fx kan anvendes i en grafisk brugerflade. Til dette projekt er der udvalgt 2 store XML dokumenter, som indeholder beskrivelser af de enkelte videoer og den programserie som de evt. er en del af.

XML dokumenterne er hentet fra nedenstående adresser og deres struktur og indhold beskrives i underafsnittene her under.

- **http://www.dr.dk/nu/api/videos/all.xml**
- **http://www.dr.dk/nu/api/programseries.xml**

2.1 Beskrivelse af dokumentet all.xml

Dette XML dokument indeholder information om alle videoer som er tilgængelige via API'et. Strukturen af dette dokument kan betragtes som værende relativ flad, da dybde i strukturen er relativ lille. Strukturen består udelukkende består af rodelementet ArrayOfProgramSerieVideo, som indeholder elementer af ProgramSerieVideo for hver tilgængelig video der findes. Selve elementet ProgramSerieVideo indeholder en række elementer direkte under sig, som beskriver detaljerne omkring videoen. Disse detaljer er sidste niveau i strukturen, hvilket gør at dybde af strukturen kan betragtes som værende flad. Skulle dybde øges vil det fx være muligt ved at samle elementer som BitrateKbps, Height og Width under et nyt element, fx med navnet TechDetails.

Omkring dokumentets indhold ses det at der er angivet en prolog, som fortæller at dokumentet er XML version 1 og er skrevet med tegnsæt UTF-8. Elementerne under elementet ProgramSerieVideo er ikke beskrevet yderligere i API'et fra Danmarks Radio, så de er i projektet fortolket på følgende måde:

- **Id:** Et unikt id for hver video der findes tilgængelig.
- **Description:** En beskrivende tekst af hver video, som typisk anvendes i en tv-program-guide.

- **ProgramSerieSlug:** Et navn som er tildelt videoer som er en del af en serie af programmer. Dette element kan betragtes som en slags fremmenøgle til elementet slug i XML dokumentet programseries.
- **Title:** En beskrivende overskrift til videoen.
- **Duration:** Videoen længde (ikke set anvendt i endnu)
- **BroadcastTime:** Tidspunkt for først gang videoen blev vist i tv.
- **ExpireTime:** Tidspunkt hvorefter videoen ikke vil være tilgængelig mere.
- **PublishTime:** Tidspunkt hvor videoen blev gjort tilgængelig.
- **Expired:** Er videoen ikke til tilgængelig mere?
- **BroadcastChannel:** Tv-kanal som videoen blev sendt på.
- **VideoManifestUrl:** Link til streamning af videoen.
- **VideoResourceUrl:** Link til streamning af videoen.
- **Premiere:** Er denne video en premierevideo?
- **BitrateKbps:** Datahastighed ved streamning af videoen. (Ikke altid sat)
- **Height:** Videoens højdeopløsning i pixels. (Ikke altid sat)
- **Width:** Videoens bredeopløsning i pixels. (Ikke altid sat)

2.2 Beskrivelse af dokumentet programseries.xml

Dette XML dokument indeholder information om alle serier af programmer, som er tilgængelige via API'et. Ved serier forstås programmer som er opdelt i mange afsnit. Strukturen af dette dokument er lidt dybere end dokumentet all.xml, men ellers er den overordnet struktur identisk. Rodelementet hedder nu ArrayOfProgramSerie og indeholder elementer af ProgramSerie, som beskriver detaljer omkring selve serien af det specifikke program. Der hvor dokumentet adskiller sig dybdemæssigt i forhold til dokumentet all.xml er i elementet Labels som kan indeholde en til mange elementer af String, som er en kategorisering af seriens emne.

I dokumentets indhold ses en prolog, som er identisk for det forrige dokuments prolog. Igen er elementerne under elementet ProgramSerie ikke beskrevet yderligere i API'et fra Danmarks Radio, så de er i projektet fortolket på følgende måde:

- **Slug:** Unik nøgle et den enkelte serie af programmer. Denne nøgle anvendes som en slags fremmenøgle i det forrige dokument (all.xml).
- **Title:** En beskrivende overskrift til serien af programmet.

```
<?xml version="1.0" encoding="utf-8"?>
<ArrayOfProgramSerieVideo
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ProgramSerieVideo>
    <Id>5062</Id>
    <Description>Det allerførste So ein Ding program ser på HP Touch Smart IQ 500.
    <ProgramSerieSlug>so-ein-ding</ProgramSerieSlug>
    <Title>Touch Smart skærme { So ein Ding</Title>
    <Duration />
    <BroadcastTime>2009-02-03T20:30:00</BroadcastTime>
    <ExpireTime>3000-01-01T00:00:00</ExpireTime>
    <PublishTime>0001-01-01T00:00:00</PublishTime>
    <Expired>false</Expired>
    <BroadcastChannel>DR2</BroadcastChannel>
    <VideoManifestUrl>http://www.dr.dk/Forms/Published/PlaylistGen.aspx?qid=1946138
    <VideoResourceUrl>http://www.dr.dk/handlers/GetResource.ashx?id=853642</VideoRe
    <Premiere>false</Premiere>
    <BitrateKbps>0</BitrateKbps>
    <Height>0</Height>
    <Width>0</Width>
  </ProgramSerieVideo>
</ArrayOfProgramSerieVideo>
```

Figur 2.1: Eksempel på dokumentet all.xml

- **Description:** En beskrivende tekst af hver serie, som typisk anvendes i en tv-program-guide.
- **ShortName:** (Ikke set anvendt)
- **NewestVideoId:** En slags fremmenøgle til id'et på den nyeste video i dokumentet all.xml.
- **NewestVideoPublishTime:** Tidspunkt for udgivelse af den nyeste video i serien af programmet.
- **VideoCount:** Antal af videoer i denne serie af programmer.
- **Labels:** Kategorisering af series indhold.
- **String:** En kort beskrivende tekst til kategorisering under Labels.

TODO: Insert stuff about study xml documents
Description of the selected XML documents
What documents we will use for this project?
What is the structure of the XML document?
What elements are there and what is described in the element?
Proposals for changes in the structure or elements? (Is there anything that can be improved) (Attributes might be used instead of some of the elements)

```
<?xml version="1.0" encoding="utf-8"?>
<ArrayOfProgramSerie
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ProgramSerie>
    <Slug>so-ein-ding</Slug>
    <Title>So ein Ding</Title>
    <Description>Det bliver ikke nemmere. Den allersidste Ding er ...</Description>
    <ShortName />
    <NewestVideoId>96149</NewestVideoId>
    <NewestVideoPublishTime>2014-01-09T11:39:28</NewestVideoPublishTime>
    <VideoCount>157</VideoCount>
    <Labels>
      <string>tech og viden</string>
    </Labels>
    <WebCmsImagePath />
  </ProgramSerie>
</ArrayOfProgramSerie>
```

Figur 2.2: Eksempel på dokumentet programseries.xml

Kapitel 3

Validering af XML dokumenterne

Som beskrevet i det foregående kapitel, så er der ikke i API'et ikke udstillet noget validerings-skema eller nogen indbygget DTD (Document Type Definition) til de XML dokumenter der er til rådighed. Den manglende mulighed for at validere XML dokumenterne betyder, at videoapplikation må stole på at XML dokumenterne altid overholder samme struktur. Ændres strukturen pludseligt vil det betyde at de søgefunktioner, som udarbejdes i dette projekt, ikke vil kunne fungere længere. Det er dermed sagt, at der kun gives garanti for at alle de funktioner, som udarbejdes i dette projekt kun kan anvendes såfremt at de XML dokumenter, som hentes fra Danmarks Radio's API, kan valideres imod de XML skemaer, som beskrives i dette kapitel.

Nogle af fordelene ved at anvende XML skema fremfor DTD er, at XML skema er meget mere kraftfuldt end DTD da XML skema eksempelvis understøtter datatyper og desuden er selve XML skemaer selv beskrevet med XML Syntax og derfor er relative lette at læse.

3.1 Beskrivelse af XML skema til all.xml

Skriv om anvendelse af sequence Skrive om valg af datatyper. Skriv om fx BroadcastChannel (findes kun i nogle programmer)

TODO: Insert stuff about xml schemas

Description of how we create them and the choice of types.

XML validation testing.

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
        <xs:element name="ArrayOfProgramSerieVideo">
                <xs:complexType>
                        <xs:sequence>
                                <xs:element ref="ProgramSerieVideo"
                                            maxOccurs="unbounded" />
                        </xs:sequence>
                </xs:complexType>
        </xs:element>
        <xs:element name="ProgramSerieVideo">
                <xs:complexType>
                        <xs:sequence>
                                <xs:element name="Id"
                                            type="xs:nonNegativeInteger" />
                                <xs:element name="Description"
                                            type="xs:string" />
                                <xs:element name="ProgramSerieSlug"
                                            type="xs:string" />
                                <xs:element name="Title"
                                            type="xs:string" />
                                <xs:element name="Duration"
                                            type="xs:string" />
                                <xs:element name="BroadcastTime"
                                            type="xs:dateTime"
                                                        nillable="true" />
                                <xs:element name="ExpireTime"
                                            type="xs:dateTime" />
                                <xs:element name="PublishTime"
                                            type="xs:dateTime" />
                                <xs:element name="Expired"
                                            type="xs:boolean" />
                                <xs:element name="BroadcastChannel"
                                            type="xs:string"
                                                        minOccurs="0"
                                                        maxOccurs="1" />
                                <xs:element name="VideoManifestUrl"
                                            type="xs:anyURI" />
                                <xs:element name="VideoResourceUrl"
                                            type="xs:anyURI" />
                                <xs:element name="Premiere"
                                            type="xs:boolean" />
                                <xs:element name="BitrateKbps"
                                            type="xs:nonNegativeInteger" />
                                <xs:element name="Height"
                                            type="xs:nonNegativeInteger" />
                                <xs:element name="Width"
                                            type="xs:nonNegativeInteger" />
                        </xs:sequence>
                </xs:complexType>
        </xs:element>
</xs:schema>
```

Figur 3.1: XML skema til dokumentet all.xml

Kapitel 4

XQuery søgning

TODO: Insert stuff about xquery search

Description of each search and how it is solved

Find all kinds of labels for prorgams. Find the number of programs for each label.

Find all kinds of broadcasting channels. Find the number of programs for each channel.

Find programs between specific date intervals.

Find all videos from a particular series of programs sorted by date.

Kapitel 5

Fuld tekst søgning

TODO: Insert stuff about full-text search
Using Full-text search in BaseX
Find relevant videos based on full text search.
Sorting results by relevance (score).
Creating and using stop-word list
Creating af word cloud (image before stop-word list and after stop-word list)

Kapitel 6

Join imellem XML dokumenter

TODO: Insert stuff about joining data
Which XML documents to be joined and why?
Compare the same XML document different dates. (Is something added,removed,updated ? list results)

Kapitel 7

XML i PostgreSQL

TODO: Insert stuff about xml in postgresql
- Importing xml into PostgreSQL
- Adding full-text search
- Compare full text performance between BaseX and PostgreSQL

Kapitel 8

Konklusion

chapter:conclusion TODO: Insert stuff about conclusion

Bilag A

Installations guide

TODO: Insert stuff about installation guide