

CPH STL designdokumenter på webben

Jesper Nielsen
Fælledvej 27 3. sal
2200 N
hubertus@diku.dk

1. Indledning

Denne opgave er skrevet i forbindelse med kurset *My favorite software development tool* udbudt på Datalogisk Institut, Københavns Universitet. Opgaven har til formål at vise publikationer og præsentationer skrevet under CPH STL projektet på websiden www.cphstl.dk som link til webserveren der indeholder disse. Dokumenterne kan hentes i forskellige filformater f.eks. .pdf .ps .gz afhængigt af deres tilgængelighed på webserveren. Dokumenterne på webserveren opdateres vha. CVS. Ændringer til CVS repositoret afspejles på websiden dynamisk.

2. Overordnet beskrivelse

Programmet består af to scripter, der behandler hhv. rapporter og præsentationer. Begge scripter genererer .php filer der fortolkes af webserveren efter filens indlæsning. Der er lagt vægt på at siden vises lynhurtigt og at der ingen interaktion, i form af søgning, sker med det underliggende filsystem når websiden vises, dvs .php filerne betragtes som statiske. De to scripter ønskes aktiveret fx én gang dagligt fra den underliggende shell. Scripterne scanner disken for rapporter og præsentationer, ordner og sorterer disse, og generer .php filerne. Scripterne aktiveres af et cronjob, der kører én gang dagligt i den aktuelle opstilling.

3. Rapporter/Artikler

Jeg vil i de følgende afsnit beskrive dokumenternes placering på filsystemet, og hvordan relevante data som forfatter, titel, dato mm. indsamles og vises på websiden, samt det regelsæt der skal være opfyldt for at dokumenterne vises korrekt på websiden i det fastlagte design.

Da jeg havde gjort det klart at scanningprocessen er rimelig I/O intensiv, måtte jeg flytte denne fra frontenden. Ved at benytte en backend løsning der periodisk scanner filerne, og genererer statiske .php filer, kunne jeg undgå scanningprocessen på frontend'en.

3.1 Rapporternes filstruktur

Rapporterne ligger i underkataloger til /Report rodkataloget. Rapporterne forventes være skrevet i tekst formateringssystemet, L^AT_EX, og indholdende hver en .tex fil. Det er ud fra denne fil at information om forfatter, titel, dato (kaldes tags i det følgende) findes, og vises på websiden. De fleste underkataloger indeholder hver én .tex fil og én eller flere .pdf .ps-filer + en evt. makefile, der genererer .pdf .ps-filer ud fra .tex filen.

Eksempel på en fil struktur for rapporter:

Roddir	1.subdir
<hr/>	
/Report	/Bitset
	cphstl-2001-1.tex
	cphstl-2001-1.pdf
	cphstl-2001-1.ps
	makefile
	/Deque
	cphstl-2000-6.tex
	..

På websiden vises følgende struktur:

Reports 2002
Reports 2001
Reports 2000
Scientific articles

CPH STL Reports 2002

yyyy-no forfattere: *titel* (.ps) (.pdf) (.gz)

..

Reports 200x

..

Scientific articles

forfattere: *titel*, bibinfo

..

Der sorteres og grupperes efter årstal, hvis årstallet ikke er tilgængeligt (eller ikke korrekt angivet), vil disse artikler blive vist over de grupperet. Som bilag er vedlagt en udskrift fra hjemmesiden.

3.2 Regler for navngivning/beliggenhed af filer

- Der må kun ligge én .tex fil i hvert underkatalog.
- De filer der ønskes vist på websiden (.pdf .ps .gz) skal have samme navn som .tex filen.

Udlades .tex filen i et katalog, vil intet blive vist på websiden fra kataloget. Udlades ovenstående navngivning af (.ps .pdf .gz) filer bliver disse ikke vist, findes ingen, bliver .tex filen ikke behandlet.

Bemærk! Findes der flere .tex filer i samme katalog, vil disse blive læst i rækkefølge, som Perl funktionen "readdir" indlæser dem, men kun den sidst fundne bliver behandlet.

3.3 Tag i .tex filen

Eksempel fra en .tex fil:

```
\authorhead{navne}
\titlehead{titel}
\dates{CPH STL Report yyyy-no, April 2001.}
\bibinfo{{.. .. (2001) .. .. . }
```

Dokumentklassen `\documentclass\{DIKU-article}` forventes benyttet. Tagene kan evt. udkommenteres med "%" hvis ønskeligt, fx. hvis artiklen ikke er skrevet med `DIKU-article` som layoutklasse. Alternativt kan benyttes `\author` og `\title` (se nedenstående).

Udlades et tag vil N/A blive vist i stedet. Udlades yyyy-no vil hele dates-taget blive vist på websiden uden sortering, og vil derfor ikke kunne grupperes. Tags `\titlehead` `\authorhead` har 1. prioritet. Tags `\title` `\author`, behandles kun hvis førstnævnte ikke findes Hvis der forekommer flere lineskift i et tag - vil kun første line til et linebreak blive behandlet. Hvis `\bibinfo` findes bliver rapporten behandlet som en Scientific article, og vil blive vist på websiden under denne type.

3.4 Implementeringsstrategi

Efter at have indsamlet navne på filerne for hvert underkatalog i en passende datastruktur bliver disse behandlet efter ovenstående regler. Ugyldige navne på filer bliver slettet fra datastrukturen, efter ovenstående regelsæt for navngivning. Alle .tex filer, vil derefter blive åbnet og tagene parset og informationer filteret vha. regulære udtryk og lagt i datastrukturen.

3.4.1 Sortering efter dato

En central del ligger i at parse `\dates` og sortere datastrukturen efter disse. F.eks 2001-1 og 2000-10 referere til henholdsvis rapport no.1 og rapport no.10. Problemet opstår når disse skal sorteres, 2000-10 vil blive betragtet som værende skrevet efter 2001-1 fordi tallet gemmes som hhv. 200010 og 20011. Dette undgås ved 0-padding, fx 2001-1 -> 2001-01. Selve sorteringen forgår først når php siden skal bygges.

3.4.2 Generering af PHP filer

Efter at alle tags er indsamlet efter gældende regler, genereres selve php siden, der består af 3 filer: Rapporterne grupperet efter aftagende årstal og stigende rapport-no indenfor hvert år, Scientific articles efter aftagende år -

og en indeks fil. Selve udtrækningen af elementer fra datastrukturen sker i sorteret orden efter dato. Endnu en sortering foregår ved at "vende" om på rapport-no, så disse vises i stigende orden. Videnskabelige articles behandles for sig, indikeret i datastrukturen, indeks filen genereres "on the fly" idet datastrukturen iterativt gennemløbes.

3.4.3 Parsning af \LaTeX tag

I nuværende implementering bliver \LaTeX tagene (kommandoerne) parset som ren tekst og vist som HTML på websiden. Dog har jeg valgt at konvertere `\texttt(x)` til HTML tag `<tt>x</tt>`. Desuden konverteres "`<`" og "`>`" til `<` og `>`, dette er escape koden for "`<`" og "`>`" i HTML. Jeg har begrænset mig til kun at behandle første linie, og alle `\titlehead` `\authorhead` indeholder blot en linie. En fuld parsning af tag `author{...}` for at filtrere navnet kunne være problematisk, da dette felt i de fleste tilfælde også indeholder informationer om sted, e-post mm. At skulle filtrere blot navnet, kunne let lede til inkonsistente resultater, hvis ikke et stramt regelsæt benyttes, men dette finder jeg uønskeligt. En anden måde kunne være at parse hele `author`, `title`, `dates` tagene, inklusiv alle \LaTeX kommandoer, konvertere dette til HTML eller ren tekst, og "gemme" dette i en `` tag i HTML (vises så ved at løbe musen over linket).

4. Slide præsentationer

Jeg skulle finde en metode der på en konsistent måde samler informationer omkring dato, titel og forfatter(e) udfra slide præsentations filerne, deres type, beliggenhed, navngivning mm.

4.1 Præsentationer - filstruktur

Præsentationerne ligger tilgængeligt enten som enkelte (.ps .pdf .gz) filer, eller som skrevet i \LaTeX , grupperet i underkataloger.

Eksempel fil struktur på slide præsentationer:

Rootdir	1.subdir	2.subdir
/Presentation	/1st workshop	/Deque
	Navn,title,dato.{.ps .pdf .gz}	navn,title,dato.{ps pdf gz}

	/2.workshop	/Project-status
	Navn-dd.mm.yyyy.tex	..
	Navn-dd.mm.yyyy.ps	..
	Navn-dd.mm.yyyy.pdf	..

På websiden vises følgende struktur:

dato author, titel på fremlæggelsen (.ps) (.pdf) (.gz)

dato author, titel på fremlæggelsen (.ps) (.pdf) (.gz)

..

Sorteret og grupperet efter dato.

4.2 Regler for navngivning/struktur

Der er 2 mulige formater.

4.2.1 Præsentationer skrevet i L^AT_EX:

Samme regler gør sig gældende til navngivning af filer (.tex .pdf .ps .gz) som under rapporter. Forskellen ligger i at datoen tages fra filnavngivningen i stedet for i \dates taget, og \titlehead og \authorhead ikke eksistere.

Eksemel fra en .tex fil:

```
\author{ dit navn }
\title{ title på projekt }
```

Filnavnet skal indeholde **dd.mm.yyyy** - herfra datoen på fremlæggelsen. Eks. jyrki-08.12.2001 .

Dokumentklassen \[a4paper]{slides} forventes benyttet. Da der kun læses første line fra taget, kan det være nødvendigt at placere forkortet versioner af \author og \title, øverst, og udkommenteret i .tex filen.

Eventuelle N/A vil blive sat ind, hvis et tag ikke findes. Samme regler gør sig gældende til navngivningen, som under rapporter (se afsnit 3.2).

4.2.2 Præsentationer der ikke er skrevet i LaTeX

Eksempel på navngivelse:

Navn på fremlægger, Title på præsentation, dd.mm.yyyy.(ps pdf gz)

Bemærk! kommasepareret.

Hvis formatet (kommasepareret) ikke er opfyldt vises filen ikke på siden. Der vises en N/A hvis dato formatet ikke er korrekt. Der kan ligge flere filer i samme katalog. Der scannes 2 subdir dybde.

4.3 Implementeringsstrategi

Der benyttes overordnet sammen strategi og datastruktur som under rapporter. Der er flere muligheder for at lede efter de relevante informationer, alt efter om der findes en .tex fil eller ej. Regulære udtryk benyttes til at filtrere informationerne efter ovenstående regler. Generering af siden sker ved at iterere gennem datastrukturen, sorteret efter aftagende dato.

5. Programmell og Setup

Jeg har valgt at bruge Perl som script sprog, der bl.a. er meget velegnet til tekstmanipulation og filhåndtering, det forventes at eksekveres i UNIX omgivelser, men kan sagtens tilpasses andre operativsystemer, ved blot at ændre stinavnene. Perl er platform uafhængigt hvis man undgår at bruge de platform afhængige operationer, som fx. signaler og lav niveau filoperationer.

5.1 Krav til værtsmaskinen

Følgende programmell og services forventes tilgængeligt på værtsmaskinen:

webservers, php, Perl, CVS, cron.

Webserveren skal kunne fortolke php, og man skal have mulighed for at sætte cron job op, men det er ikke et krav.

5.2 Setup

I filerne reports.pl og presentations.pl skal data indtastes direkte i scriptet omkring webserversens dokumentrod og udfilerne placering.

wwwdir:

Dette er webservers dokumentrod, "/"sti.

reportdir/presdir:

Reports/presentation kataloget, relativt fra webserversens dokumentrod

outfile:

.php filen, der henvises til, fra menuen (se udskrift fra websiden)

outfileidx (kun repports.pl):

Indeks, bliver inkluderet (php include()) fra fra *outfile*, hop til et indeks markering <a name> HTML tag.

outfileconf (kun repports.pl):

Scientific articles, bliver inkluderet fra *outfile*.

excludedir:

Kataloger der ikke ønskes behandlet (fx CVS kataloget)

Cron setup:

Fx,

```
5 6 * * * root cd /var/www && cvs -d /usr/local/CPHSTL -Q update -d
```

Report

```
5 6 * * * root cd /var/www && cvs -d /usr/local/CPHSTL -Q update -d
```

Presentations

```
10 6 * * * root /usr/home/hubertus/cron/scan.sh
15 6 * * * cd /var/www && cvs -d /usr/local/CPHSTL -Q update -d WWW
```

Hver dag kl 6.05 opdateres rapporter og præsentationer, scan.sh må nødvendigvis køres efter opdateringen fx 5min efter, WWW repositoret opdateres sidst 6.15.

scan.sh:

```
/var/www/WWW/Script/reports.pl 2>> errorlog &&
/var/www/WWW/presentations.pl 2>> errorlog
```

6. Konklusion - forbedringer

Opgaven har været at betragte som en en mindre webudviklingsopgave, hvor jeg sammen med opgavestiller, Jyrki Katajainen, iterativt er kommet frem til løsningen. Jyrki har kommet med ønsker og forventninger til websidens struktur, samt været ansvarlig for cron job, CVS mm. Jeg har implementeret løsningen og dokumenteret et regelsæt der garanterer det forventede output på websiden. Jeg har dog set en række forbedringer der kunne gøre funktionaliteten/vedligeholdelsen endnu bedre. (se afsnit). Opgaven har givet mig en god indsigt i sproget Perl og anvendelse af regulære udtryk, samt ikke mindst \LaTeX .

Programkoden er ikke inkluderet, men kan fremsendes på opfordring.

6.1 Fremtidige forbedringer

- Fuld konvertering mellem \LaTeX kommandoer og HTML.
- Fuld parsning af \backslash author \backslash title tags.
- Bedre adskildelse af programkode og HTML, for at gøre det lettere at rette i designet.
- Software til automatisk generering af regelsæt for filnavne, indhold af tags, placering på filsystemet mm.

www.cphstl.dk