

Filstruktur og CVS-struktur, organisering av databasen

by Sjur N. Moshagen

Table of contents

1 Filstruktur og CVS-struktur.....	2
1.1 Filstruktur i Forrest.....	2
1.2 CVS-struktur.....	2
1.3 Installera portalen.....	4
2 Databasestructur.....	4

1. Filstruktur og CVS-struktur

Filene i ordboksdatabasen er lagra to stader:

1. i to webapp-katalogar i Tomcat - det er desse som brukarane ser
2. i eit CVS-repositorium, for å sikra versjonskontroll, og gje fleire brukarar tilgang til tekstene samtidig

Nedanfor er dei to filstrukturane, og samanhengen mellom dei, skissert.

1.1. Filstruktur i Forrest

Filstrukturen til eit Forrest-prosjekt ser slik ut:

```

/                               # $PROJECT_ROOT =
/Volumes/Data/Users/sjur/Documents/samsandok
/status.xml                     # List of project developers, todo list and
change log
/forrest.properties            # Optional file describing your site layout
/src/documentation/            # Doc-specific files
/src/documentation/skinconf.xml # Info about your project used by the
skin
/src/documentation/content/     # Site content.
/src/documentation/content/     # Static content files
/src/documentation/content/xdocs # XML content.
/src/documentation/content/xdocs/index.xml # Home page
/src/documentation/content/xdocs/site.xml # Navigation file for site
structure
/src/documentation/content/xdocs/tabs.xml # Skin-specific 'tabs' file.
/src/documentation/content/xdocs/images # Project images (logos, etc)

```

Meir info om Forrest finn du på [heimesida til Forrest-prosjektet](#).

1.2. CVS-struktur

Note:

Per 1. februar 2005 er CVS-repositoriet omorganisert. Dokumentasjonen nedanfor gjeld den nye strukturen. Dei som evt. har sjekka ut ein lokal kopi før denne datoen, må sletta han og sjekka ut på nytt (ta kopi av ev. lokale endringar fyrst!).

I versjonar av Forrest frå 0.6 og nyare, er det slutt på å kopiera filer frå strukturen over til instansar i ein webapp (som når ein skriv `forrest run`). I og med omorganiseringa av CVS-en har eg derfor lagt Forrest-strukturen over til grunn, med nokre tillegg som vist nedanfor.

CVS-strukturen speglar den logiske oppdelinga i tre hovudkomponentar: konvertering frå gammalt til nytt format, databasegrensesnittet (søking og redigering), og den kringliggjande dokumentasjonen. Dokumentasjonsdelen er delt i to: ein del som inneheld den eigentlege dokumentasjonen organisert etter standardhierarkiet til Forrest, og ein del som inneheld lokale tilpassingar og endringar av Forrest. Resultatet er:

```
$CVSROOT/termdb/src/db-conv/      ! konvertering frå SQL/8-bits til
```

```
XML/utf-8
db-app/      ! søkjegrensesnitt / interaksjon med
XML-basen
documentation/ ! portal og dokumentasjon, alt
utanfor XML-basen
site-app/    ! tilpassingar av portalen,
modifikasjonar av Forrest
forrest.properties ! grunnleggjande innstillingar for
Forrest
status.xml   ! inneheld status for prosjektet:
endringar og gjereliste
```

Desse fire delane er strukturert som vist nedanfor.

1.2.1. Konverteringsmodulen

Katalogstruktur for konverteringa frå ein 8-bits SQL-database til ein utf-8-koda XML-database:

```
db-conv/bin/      ! Ferdige konverteringsverktø
db-files/         ! databasefiler
db-files/doc/     ! dokumentasjon for databasefiler
doc/              ! overordna dokumentasjon for konverteringa
src/              ! kjeldefiler for konverteringsverktøya
src/doc/          ! dokumentasjon for konverteringsverktøya
```

1.2.2. Databasemodulen

Katalogstruktur for databasen:

```
db-app/admin/    ! passordbeskytta område for administrasjon
query/           ! søkjeskript
edit/            ! redigeringskript
support/         ! Additional supporting files
```

1.2.3. Forrest-basert dokumentasjon

Katalogstruktur for portalen:

```
documentation/classes/ ! intern Forrest-mappe
content/              ! det eigentlege innhaldet i portalen;
! statiske filer ligg her.
content/xdocs/       ! XML-dokument her, detaljstruktur finst
nedanfor
resources/           ! ressursar for å prosessera eigne format,
eller
sitemap.xml         ! annan tilpassing
! Sitemap for å overkøyra
standard-forrest;   kan brukast til å handtera eigne
XML-format
skinconf.xml        ! tilpassing av Forrest-portalen, inkl.
! eigne stilar, søkjemotor, m.m.
translations/       ! omsetjingar for flikar og menyar
```

Innhaldet er organisert slik:

```
xdocs/              ! det eigentlege innhaldet i portalen
```

```

index.xml      ! Framsida
site.xml       ! "katalog" over alle filer som skal publiserast,
               ! med visingsnamn - grunnlaget for menyane
tabs.xml      ! fil for flikane i portalen
gram/         ! grammatikkar finst her
  sme/        ! nordsamisk grammatikk
  smj/        ! lulesamisk grammatikk
  sma/        ! sørsamisk grammatikk
help/         ! hjelp for portalen
hist/         ! det som ligg under Historikk i den gamle portalen
images/       ! bilete og grafikk som heng i hop med dokumenta,
               ! ordna i ein parallell katalogstruktur:
  hist/       ! grafikk for hist-dokumenta
  gram/sme/   ! grafikk for nordsamisk grammatikk
  specs/      ! grafikk for spesifikasjonane
specs/        ! prosjektspesifikasjonar

```

1.2.4. Tilpassingar av Forrest og den ferdige portalen

Visse lokale tilpassingar har eg gjort

```

site-app/httpd/      ! Mappe for introsider som automatisk
opnar portalen
  schema/            ! Alternative eller modifiserte DTD-ar
ligg her
  site/              ! Innhaldet her er forelda:
Makefile              ! make-fil for å "installera"
filene i mappa
javabasert           ! make-system, build-fila er ei
XML-fil)
  skin/              ! "huda", utsjånaden til portalen (delvis
forelda):
Makefile              ! make-fil for å "installera" filene i
underkatalogane
  css/               ! CSS-filer for huda
  images/            ! grafikk som huda bruker
  xslt/              ! hud-spesifikke XML-transformasjonar

```

1.3. Installera portalen

For å installera portalen på servaren til Sametinget, må ein laga eit "web-arkiv" av web-applikasjonen:

```
> forrest war#
```

Dette arkivet (war-fila) må ein så kopiera og installera i Tomcat (web-applikasjonsservaren) til servaren. Hugs å fjerna den gamle applikasjonen fyrst!

2. Databasestruktur

Databasen (eXist) organiserer xml-filene i eit hierarki likt eit vanleg filsystem, men i staden for mapper kallar eXist det for *collections* - ei *samling* med xml-dokument. Kvar slik samling kan ha andre samlingar i seg, og dermed får ein ein hierarkisk struktur. Kvar database har ei rot-samling, /db. Nedanfor fylgjer ei skisse til korleis databasen kan

organiserast.

```
/db/ordbase/          ! Felles rot for alle termsamlingar og
ordboksfiler;       !
andre formål        ! gjer det mogleg å ha andre samlingar for
                    ! om det trengst seinare
                    ! rot for terminologisamlingar
terms/              !
class/             ! klassifikasjonssystem: det er mogleg å ha
fleire paral-      !
                    ! lelle klassifikasjonssystem
                    ! $NAME/ ! ei samling per terminologisamling, $NAME er
eit stutt          !
                    ! namn på samlinga
                    ! dict/ ! rot for ordbøker
                    ! $NAME/ ! ei samling per ordbok, $NAME er ei stuttform
av ordboka        !
                    ! som eignar seg for teknisk bruk (blir ikkje
vist brukaren)   !
                    ! common/ ! felles ressursar for alle ordbøker og
terminologisamlingar,
                    ! typisk ordklasser og bøyingsklasser (dei er
språk-           !
                    ! spesifikke, men ikkje spesifikke for kvar
ordbok el.)
```

Skillet mellom termar og ordbøker er reint teknisk, og speglar ingen lingvistiske realitetar. (Betydingar av) ord som det er hensiktsmessig å organisera på tvers av fleire språk (t.d. faguttrykk/-termar, medisinske termar, flora og fauna) blir termar i kraft av måten dei er organisert på i databasen/xml-filene, og alle andre ord (som fylgjer ein tradisjonell tospråkleg ordboksstruktur) blir ordbøker.

Kvar ordbok og kvar termbase ligg i si eiga samling, og vil vera bygd opp av fleire filer. Ein typisk termbase vil ha ein struktur som skissert her (for detaljar om xml-strukturen, sjå [spesifikasjonen for ny termstruktur](#)):

```
/terms/$NAME/       ! termbase med kortnamnet $NAME (sjå
nedanfor)
meta.xml            ! tekniske metadata
front-sme.xml       ! innleiing, forord, forkortingar, samisk
versjon
front-nor.xml       ! innleiing, forord, forkortingar, norsk
versjon
...                 ! ... fleire språkspesifikke innleiingar m.m.
termcenter.xml     ! navet for termbasen, ordna etter omgrep
terms-sme.xml       ! språkspesifikke termpostar, nordsamisk
terms-nor.xml       ! språkspesifikke termpostar, norsk
...                 ! ... tilsvarande ei fil pr språk
exmpl-sme.xml       ! språkspesifikke døme i ei eiga xml-fil
...                 ! ... fleire språkspesifikke dømefiler
```

Tilsvarande vil ordbøker vera organiserte i separate samlingar, ei samling pr. ordbok, og med ein fast filstruktur (som eit døme kan vi tenkja oss ei ordbok mellom nordsamisk og norsk, baa vegar):

```
/dicts/$NAME/      ! ordbok med kortnamnet $NAME
meta.xml            ! tekniske metadata
front-sme.xml       ! innleiing, forord, forkortingar, samisk
```

```
versjon      front-nor.xml      ! innleiing, forord, forkortingar, norsk
versjon      dict-nor-sme.xml   ! norsk-samisk ordbok
              dict-sme-nor.xml   ! samisk-norsk ordbok
              exmpl-nor-sme.xml ! norske eksempel med samisk omsetjing
              exmpl-sme-nor.xml ! samiske døme med norsk omsetjing
```

Dei ordboksbasane som er definert til no (dvs innhaldet av \$NAME i opplistingane over) er:

```
SD-terms      ! Terminologibasen til Språkavdelinga, Sametinget
              ! Inneheld alle oppslag med ein fagkode i den gamle
databasen
SD-dict-sme   ! Dei ikkje-terminologiske oppslaga i den gamle databasen
```

Note:

SD-dict-sme er ikkje i bruk enno, og står der berre som ein planlagd modul. Namnet kan bli endra i framtida.