

[Drupal]

[utseende]



Brendan Johan Lee

Department of Informatics
University of Oslo, Norway

`brendajl@simula.no`

March 9, 2011

- Til Oblig 4 - utseendemaler anbefaler jeg på det sterkeste å bruke Drupal installert på egen maskin eller i vmware-bildet dere har fått utdelt
- Enormt merarbeid på kursnettstedene (stadig pakking og opplasting av filer)
- Man bør ha mulighet til å endre filer direkte
- Derfor bør hver gruppe sørge for tilgang til en maskin hvor man kjører Drupal selv
- Hverken forelesning eller tutorial gir noen komplett oversikt over det å lage utseendemaler i Drupal, vi skummer bare toppen. Pensumboka anbefales som supplement.
- Last opp utseendemalen på kursnettstedet først når den er ferdig

- For å lage utseendemaler skal vi benytte oss av *devel* og *Theme developer*-modulene, samt drupal for firebug
- Man må sørge for at disse er installert og aktivert på nettstedet man skal jobbe på

- Dagens forelesning er tilgjengelig både som slides, og en mer grundig tutorial
- Vi går raskt gjennom ting nå, deretter får dere utdelt en tutorial som dere nå kan begynne å følge
- Gå gjennom tutorialen på egenhånd hjemme, slik at dere er klar for neste lab
- Neste time tar vi og jobber i fellesskap på en lab med mulighet for hjelp med utvikling av utseendemal

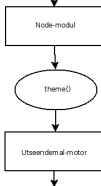
- Drupal har klart og tydelig skille mellom utseende og innhold
- Vi har til nå konsentrert oss om innhold, men ser nå på utseende
- En god utseendemal er det som skiller et godt Drupal-nettsted fra et nettsted som ser “typisk Drupal” ut.
- Utseendemaler består av bilder, CSS og vanligvis HTML/PHP
- Mulig å lage rene utseendemaler i CSS, men det er ikke vanlig

- Standard og ferdiginstallert i Drupal er PHPTemplate
- Flere andre tilgjengelig (se <http://drupal.org/project/Theme+Engines>)
- Vi går ikke gjennom annet enn PHPTemplate
- Bytte av utseendemalmotor anses som avansert materiale for viderekomne

- Om man “hacker” kjernen vil man få problemer
- Drupal er helt klart inndelt i “ting som bruker kan endre” og “ting man skal holde seg unna”
- Endrer man “ting man skal holde seg unna” vil man miste alle endringer ved oppdatering av Drupal (og oppdateringer gjøres tilgjengelig ofte)
- Om man endrer kjernen eller andre slike ting lager man i utgangspunktet en egen proprietær versjon av Drupal
- Ved slik “hacking” er man garantert at nettstedet man lager vil havarere og havne i bakevja

- Alt innhold genereres av moduler
- Hvilken modul som genererer hva avhenger av innholdstype.
For eksempel vil
 - *Noder* - genereres av *Node*-modulen
 - *Administratorsider* - genereres av *System*-modulen
 - *Views* - genereres av *Views*-module
- Felles for alle moduler: Etter at de har generert innhold, kaller de spesialfunksjonen *theme()*
- *theme()*-funksjonen tar seg deretter av all formatering av innholdet (vi vil se hvordan senere)

http://dwi.ifi.uio.no/node/1



N30

no, 02/02/2010 - 11:28 -- admin
Produktid: N30
Pris: 100000
SKU: N30000

Detta er en fiktiv beskrivelse på minimum 10 ord for å bli sett av en interessebegreper.

Average:

Four
Five
Six
Seven

Average

0.00 (0)

Your rating helps Average: 1 (0 votes)

[Logg på](#)

- [Logg på](#) eller [registrer en konto](#) for å gjøre kommentarer



N30

no, 02/02/2010 - 11:28 -- admin

Produktid: N30
Pris: 100000

Detta er en fiktiv beskrivelse på minimum 10 ord for å bli sett av en interessebegreper.

Average

★★★★☆

0.00 (0 votes)

Your rating helps Average: 1 (0 votes)

[Logg på](#)

- [Logg på](#) eller [registrer en konto](#) for å gjøre kommentarer

- Alfa og omega for en utseendemal er .info filen
- navnpåmal.info
- To funksjoner:
 - Forteller Drupal at mappen inneholder en utseendemal
 - Gir Drupal informasjon om utseendemalen

```
; $Id: garland.info,v 1.5 2007/07/01 23:27:32 goba Exp $
name = Garland
description = Tableless, recolorable, multi-column, fluid width theme (default).
version = VERSION
core = 6.x
engine = phptemplate
stylesheets[all][] = style.css
stylesheets[print][] = print.css

regions[header] = Header
regions[footer] = Footer
regions[left] = Left sidebar
regions[right] = Right sidebar
regions[content] = Content
```

- Oversikt over det viktigste innholdet i .info-filen
 - *name* - menneskelesbart navn på utseendemal (maskinlesbart navn er navnet på .info filen)
 - *description* - menneskelesbar beskrivelse av utseendemal (som vises i listen over utseendemaler)
 - *version* - Utseendemalens versjon
 - *core* - Drupalkjerneversjonen utseendemalen er lagd for (f. eks. 6.x)
 - *engine* - (frivillig) utseendemal-motor
 - *stylesheet[type][] = xxxx.css* - inkluder eller overstyr en .css-fil
 - *scripts[] = xxxx.js* - inkluder eller overstyr en javascriptfil
 - *regions[navn] - menneskelesbartnavn* - definerer en region - altså et sted man kan plassere blokker
 - *features[] - navn* - definerer en tilgjengelig *Feature* (logo, søkeboks, etc)

- Seks (sju) ting man må holde styr på:
 - .css-filer
 - (javascriptfiler)
 - templates
 - variabler
 - hooks
 - regioner

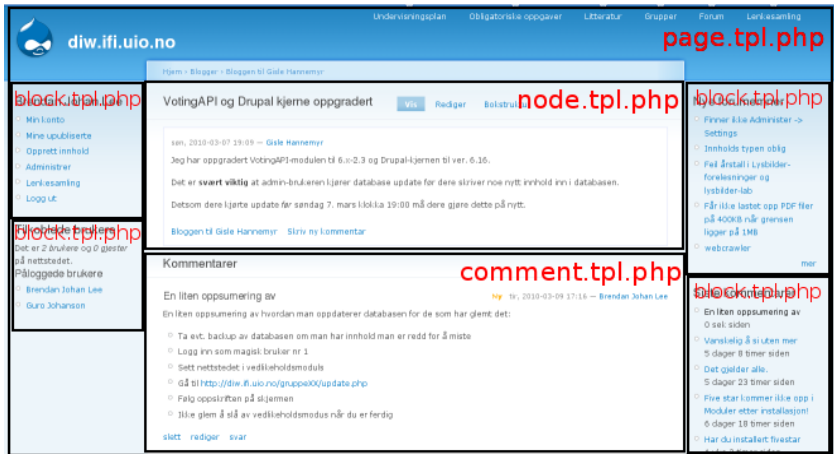
- .css-filer brukes for å bestemme utseende på elementer på nettstedet
- går ikke gjennom .css, mye bra introduksjoner på nett, om noen ønsker en forelesning om det si fra
- man kan inkludere egne .css-filer i utseendemal
- eller man kan overstyre eksisterende .css-filer (moduler f. eks. har ofte egne .css-filer)
 - ikke rediger eksisterende .css-filer i Drupal men i stedet:
 - kopier .css-filen til utseendemalmappa
 - gjør endringer
 - inkluder kopien vha .info-filen

- Når man jobber med .css-filer i Drupal må man de-aktivere og re-aktivere utseendemal hver gang man ønsker å se en endring slik at utseendemalmellomlageret tømmes
- *Men* siden vi har installert *devel*-modulen kan vi få den til å gjøre dette for oss hver gang en side rendres:
 - Administrer... Innstillinger... Devel settings... Rebuild the theme registry on every page load
 - Bare husk å slå av når man er ferdig

- Firefox
- Firebug-plugin til Firefox
- Drupal for Firebug modul og plugin
- I Firefox: Tools... Firebug... Inspect elements (eller Ctrl+Shift+c)

- Live demo

- Bestemmer plassering av elementer på nettstedet
- Blanding av (X)HTML og litt PHP
- Navngis etter hva som genereres: f. eks. page.tpl.php og node.tpl.php
- Hierarki som styrer hvilke elementer som lar seg påvirke f. eks. vil story-node.tpl.php
- Mer om hierarkiet i tutorial
- For å bruke templates, kopier dem til utseendemalmappa (evt. endre navn med tanke på hierarkiet) og endre dem
- Mer om templates i tutorial



The screenshot shows a Drupal blog page for 'diw.ifi.uio.no'. The page layout is annotated with red text labels for various template blocks:

- page.tpl.php**: Located at the top right of the page.
- block.tpl.php**: Located on the left sidebar (navigation menu), the top left of the main content area, and the bottom right of the main content area.
- node.tpl.php**: Located in the top right of the main content area, overlapping the article title.
- comment.tpl.php**: Located in the top right of the main content area, overlapping the article text.
- block.tpl.php**: Located in the bottom right of the main content area, overlapping the comment section.

The main content area displays a blog post titled "VotingAPI og Drupal kjerne oppgradert" by Gisle Hannemyr, dated 2010-03-07. The post text states: "Jeg har oppgradert VotingAPI-modulen til 6.0-2.3 og Drupal-kjernen til ver. 6.16. Det er svært viktig at admin-brukeren kjører database update før dere skriver noe nytt inn i databasen. Detsom dere kjørte update før søndag 7. mars klokka 19:00 må dere gjøre dette på nytt." Below the post is a comment section with one comment by "Brendan Johan Lee" dated 2010-03-09 17:16, which discusses database backup and update procedures.

```
<html>  
<head>  
<title><?php print $head_title; ?></title>  
</head>  
<body><?php print $content;?></body>  
</html>
```

- Ved hjelp av *Theme developer*-modulen
- Live demo

- Bestemmer hvor det er lov å plassere blokker
- 5 stk definert som standard
- Om man ønsker å definere nye, må man også definere de som er standard
- Defineres i .info-filen og inkluderes i page.tpl.php
- Mer i tutorial

- I Drupal er *hooks* en stor greie
- Dere kjenner dem kanskje som *callback*-metoder
- Altså metoder med kjente navn som Drupal kaller før eller etter den “gjør noe”
- Ved hjelp av *hooks* kan vi kontrollere nesten alt
- Mer om *hooks* i tutorial

- Alt innhold Drupal genereres plasseres i variabler
- Variablene skrives ut i diverse templates
- Ved hjelp av preprocess *hooks* kan vi huke tak i variablene og endre dem etter innholdet er plassert i variabelen men før den inkluderes i template
- Brukes for å endre innhold, f. eks. hvilken tekst som skal vises i en nodetittel
- Navn på variabel som skal endres finner man i template-fil. Altså finn ut hvilken template-fil som behandler elementet, åpne template-filen, og finn navnet på variabelen
- Mer om overstyring av variabler i tutorial
- Husk at man bare overstyrer variabler når man ønsker å endre innhold, ikke når man ønsker å endre utseende

- Noen ganger er det vanskelig eller umulig å bare endre verdien av en variabel
- Da kan vi i stedet overstyre selve *hooken* som genererer innholdet
- Lignende metode som å overstyre variabler

- Ved hjelp av *Themer devel modulen*
- Og ved behov *nix kommandolinjeverktøy
- Live demo
- Mer om hooks i tutorial

- Vi har bare gått gjennom ting overflatisk
- *Les gjennom og utfør tutorial før neste gruppetime*
- Forsøk å forstå det du kan, det du ikke forstår, kan vi ta på gruppetimen
- På neste gruppetime er det STOR fordel å ha med egen bærbar maskin med Drupal-installert, i hvertfall en for hver gruppe.

- Det er bare en ordinær gruppetime igjen. Om 14 dager går vi gjennom WAI
- Resten av gruppetimene er “frie”
- Send meg epost med problemer eller temaer dere ønsker forelesning om
- Ellers vil jeg være tilgjengelig i gruppetimene for å hjelpe dere med Drupal
- Begynn NÅ å lag nettstedet deres, ikke forvent at det er mulig å gjøre alt siste uken
- Jo fortere dere finner ut av hva dere trenger hjelp/forelesning om, jo større sjanse for at vi faktisk får tid.