



DOCTYPE og valid kode - ens udseende i Internet Explorer og FireFox

Langt de fleste artikler her på eksperten omhandler fup eller fakta eller guides til optimering - men den artikel du er ved at læse nu adskiller sig en del på den måde, at fakta allerede er defineret og dette er "kun" et opråb om også at benytte dem.

Skrevet den **08. Jun 2009** af **keysersoze** i kategorien **(D)HTML / Andre** | ★★★★★

Artikel historik

20090609 - oprettet

20090731 - opdatering i forbindelse med DOCTYPE og boxmodellen

Jeg ved ikke helt hvad årsagen er, men den seneste tid er jeg stødt på utrolig mange spørgsmål gående på, hvorfor en hjemmeside ikke ser ens ud når man tester i fx Internet Explorer og Firefox. Grunden til min undren skal nok mest af alt findes i, at det jo som sådan ikke er noget nyt "problem" da det har eksisteret så længe der har været mere end én browser på markedet, men det skyldes måske bare at man - langt om længe - er blevet mere opmærksom på problemet.

Så godt som alle af disse spørgsmål kan besvares med en eller begge af følgende løsninger;

- har du sørget for at skrive valid kode?
- har du husket en DOCTYPE?

Valid kode

HTML, XHTML og for den sags skyld også CSS er ikke bare tilfældige kode-elementer man frit kan sætte sammen som man lyster - der ligger nogle regler til grund for sprogene, og følger man ikke disse regler kan det meget let have konsekvenser fx i form af, at siden ikke helt får det udseende man egentlig forestillede sig eller at hver browser fortolker og ikke mindst forsøger at rette op på fejlene på hver sin måde så udseendet dermed bliver forskellige fra den ene til en anden.

Det er organisationen W3 (World Wide Web Consortium på <http://www.w3.org/>), der styrer retningslinjerne/anbefalingerne for bl.a. HTML, XHTML og CSS - og når jeg siger retningslinjer og ikke standarder så skyldes det, at det teknisk set faktisk ikke er standarder men kun retningslinjer der udstedes, men reelt set skal de nu alligevel betragtes som standarder. Er man derfor i tvivl om hvad man kan og ikke kan inden for hver af retningslinjerne er deres site derfor et fremragende, omend måske knap så brugervenligt, opslagsværk.

Ved at overholde standarderne sikrer man bl.a. at man ikke koder til én specifik browser hvilket man let kunne komme til tidligere da hver browser-producent kunne opfinde og understøtte egne elementer (fx havde Internet Explorer <marquee> og Netscape have <ilayer> samt <blink>) og dette betød, at sider ikke ville virke hensigtsmæssigt i andre browsere. Ved overholdelse af standarderne sikrer man, at man udvikler til alle klienter, der understøtter standarderne og ydermere vil valid kode også sikre imod de mest oplagte visningsfejl - har man fx et start- eller et slut-tag for meget eller for lidt kan det have store konsekvenser for visningen i browseren og det kan virkelig spare tid hvis man i valideringen kan se, at det er en simpel validitetsfejl man skal rette op på og at det ikke en større hovsa-løsning man skal ud i for at få tilpasset siden. Endelig vil det være betydeligt nemmere at få kode-hjælp på diverse fora da det i mange

tilfælde vil være irrelevant at fejlsøge og give løsninger uden valid kode da fejlene oftes skyldes netop den invalide kode.

DOCTYPE

Som nævnt ovenover er (X)HTML ikke bare tilfældig kode sat sammen efter behag, derimod ligger der altså retningslinjer til grund sprogene - og det er for at fortælle browseren hvilken standard ens side, forhåbentlig, opfylder at man benytter en DOCTYPE. Benyttes en forkert eller i værste fald ingen DOCTYPE kan det give lige så uheldige bivirkninger på ens side som invalid kode - det kan fx betyde, at browseren vælger en forkert renderings-mode (læs mere om det længere nede i artiklen), at store dele af ens CSS sættes ud af drift og at siden læses langsommere af browseren. Mere eller mindre alt sammen noget, der også har betydning for, hvor crossbrowser siden fungerer.

Det er ikke kun vigtigt, at du har en DOCTYPE på din side - placeringen er også vigtigt; er DOCTYPE ikke det absolut første i dit dokument vil din side heller ikke blive tolket korrekt. Det vil altså sige, at hvis du placerer fx javascript-kode før din DOCTYPE eller, som nogle tror de skal når de laver XHTML, placerer en XML-deklaration som det første vælger browseren den forkert renderings-mode.

Et vigtig ting i forbindelse med valg DOCTYPE er hvorvidt man skal vælge at køre Frameset, Transitional eller Strict. Frameset burde give sig selv som valget til sider der indeholder frames, men dem holder vi os selvfølgelig langt fra - men hvornår skal man så vælge Transitional og hvornår skal man vælge Strict? Jeg vil til enhver en tid anbefale at man går efter at benytte Strict - dette dels fordi det er hvad sprogene som standard er konstrueret til og dels, at man ved at benytte Strict udelukker brugen af præsentationsattributter og -elementer hvilket tvinger en til at benytte CSS, altså opnår man en logisk opsplnitning mellem indhold og præsentation hvilket igen gør koden mere overskuelig og gør præsentationsændringer lettere. Transitional indeholder de samme muligheder som Strict - men åbner altså op for en lettere overgang fra ældre versioner, bl.a. i form af, at der kan benyttes attributter til at styre udseende.

Nederst i artiklen finder du link til hvilke DOCTYPEs du kan benytte - og det er vigtigt at de kopieres direkte og ikke tilpasses efterfølgende. For eksempel er XHTML case-sensitiv og jeg har set eksempler hvor "EN" var udskiftet med "DA" med tanke på at det jo var en side på dansk man skrev, men værdien hentyder ikke til sproget på sitet men derimod til det sprog definitionen er skrevet i, og de er skrevet på engelsk - og ændringer kan altså betyde, at den indsatte DOCTYPE bliver lige så forkert som var der slet ingen DOCTYPE.

Generelt er DOCTYPE en lidt sjov størrelse i og med at et korrekt udseende (altså i henhold til at undgå at gå i quirksmode) kan varierer meget. Eksempelvis vil en DOCTYPE uden URL være valid i XHTML og HTML4 Strict men ikke valid i HTML4 Transitional, så anbefalingen må være at gå ud fra de anbefalede du finder i linket nederst i artiklen.

En par *rigtige* DOCTYPEs:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN">
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

en *forkert* DOCTYPE:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

Det er vigtigt at lægge mærke til, at DOCTYPE altså *ikke* et HTML-tag, det er en instruktion til browseren omkring hvilken version siden er skrevet i.

Renderings-mode - !DOCTYPE "Switch"

Da browserne skal understøtte mange forskellige standarder, og specielt pga de helt gamle, kan der køres med 2 forskellige renderings-modes, nemlig standard- samt quirks-mode. Dette skift mellem standard- og quirks-mode bliver der oftest refereret til som *The !Doctype Switch* da det netop er vores DOCTYPE der er med til at bestemme renderingsmoden.

Den renderingsmode vi er interesseret i er uden tvivl standard-mode, da det er denne der håndterer W3-standarderne bedst muligt - kører siden i quirks-mode vil browserens fortolkning svare til helt gamle browseres fortolkninger og det er selvsagt meget uhensigtsmæssigt. Hvis du vil se hvilken renderingsmode dit site kører under kan du bl.a. se dette i Firefox ved vælge "Vis sideoplysninger" i højrekliksmenuen.

Et af de vigtigste punkter under *The !Doctype Switch* er den såkaldte *boxmodel*. Meget kort fortalt går boxmodellen ud på, hvordan højde og bredde af elementer sammensættes til det endelige visuelle resultat (ud fra *height*, *width*, *padding*, *border* samt *margin*), og da alle elementer i HTML i bund og grund er en firkantede byggeklodser kan man let gå galt i byen hvis ikke man har gennemskuet boxmodellen. Nederst i artiklen kan du finde links til længere forklaringer om boxmodellen.

Det kan nok være lidt svært at forestille sig hvilken betydning det reelt set har om man kører under den ene eller den anden renderingsmode, så betragt billeder på <http://web-dev.dk/post/2009/06/DOCTYPE-og-valid-kode---ens-udseende-i-Internet-Explorer-og-FireFox.aspx>, hvor den eneste forskel i den bagvedliggende kode er brugen af DOCTYPE eller ej så de dermed kører i hver sin renderings-mode.

Hverdageksempel

En nok desværre for udbredt holdning er, at man kan bruge tid på at gøre sin kode valid så snart man har fået siden til at se ud som den skal - men ofte er den manglende validitet netop årsagen til, at siden ikke vises som ønsket og så er det jo som at angribe mod sit eget mål i fodbold; nemlig nærmest håbløst, i hvert fald hvis man vil vinde. For ganske kort tid siden sad jeg og fulgte en debat med netop denne problematik - udviklerens side blev på flere punkter ikke vist som ønsket og trods flere kommentarer om det irrelevante i at forsøge at rette op på visningen før end siden var valid blev dette ikke accepteret som værende en løsning, for selvfølgelig måtte der findes en mulighed uden at der skulle bruges tid på at lære standarderne at kende og ikke mindst følge dem... Det viste sig bare, at årsagen til hele problemet selvfølgelig var den invalide kode; der var - blandt flere andre fejl - brugt to <body> tags hvor kun ét er tilladt. Så snart det overskydende <body> var fjernet blev alle fejl i visningen løst - koden var stadig ikke helt valid (og blev det formentlig desværre aldrig), men det viser bare hvor lidt det nytter at gøre tingene i omvendt rækkefølge.

Jamen, det virker stadig ikke?

Nu gik vi lige og troede at brugen af DOCTYPE og valid kode var svaret på alle vore bønner - men sådan er det desværre ikke helt. Det skal nemlig også nævnes, at bare fordi koden er valid kan man desværre ikke altid forvente at siden vises helt ens i forskellige browsere - der kan let stadig være små eller i sjældne tilfælde måske endda store forskelle og i de tilfælde er der ikke andet for end at lære hvad forskellene går ud på så man ved det til en anden gang. Ofte er det en simpel ændring/tilføjelse der skal til for at løse problemet og det er bestemt også værd at nævne at det da heller ikke nødvendigvis skyldes dårlig kode men bare den enkelte browsers måde at fortolke den samme ting på. Valideringen af ens side kan lidt tolkes som stavekontrollen i Word - den sikrer imod stavefejl, men kan af naturlige årsager ikke validere på betydningen/visningen af dokumentet.

Husk også - god kode er altid valid, men valid kode er ikke altid god kode.

Gode link

W3C Markup Validation Service; <http://validator.w3.org/> - brug denne service til at validere din (X)HTML

kode, du kan validere både via en URL, filupload samt direkte kodeinput.

W3C CSS Validation Service; <http://jigsaw.w3.org/css-validator/> - samme som ovenstående, bare til din CSS.

Anbefalede DOCTYPEs; <http://www.w3.org/QA/2002/04/valid-dtd-list.html> - er du i tvivl om hvordan din DOCTYPE skal se ud i specifikke tilfælde indeholder denne side de relevante svar.

Boxmodellen; <http://www.w3.org/TR/css3-box/> samt <http://www.w3.org/TR/CSS2/box.html>

Lidt om både boxmodel og !DOCTYPE Switch; [http://msdn.microsoft.com/en-us/library/bb250395\(VS.85\).aspx#cssenhancements_topic3](http://msdn.microsoft.com/en-us/library/bb250395(VS.85).aspx#cssenhancements_topic3)