

Införande av CSS i existerande tabellbaserat HTML-dokument

Henrik Larsson

Införande av CSS i existerande tabellbaserat HTML-dokument

Examensrapport inlämnad av Henrik Larsson till Högskolan i Skövde, för Kandidatexamen (B.Sc.) vid Institutionen för kommunikation och information. Arbetet har handletts av Bengt Efring.

2006-06-07

Härmed intygas att allt material i denna rapport, vilket inte är mitt eget, har blivit tydligt identifierat och att inget material är inkluderat som tidigare använts för erhållande av annan examen.

Signerat: _____

Sammanfattning

För att på ett effektivt sätt kunna konstruera komplexa lösningar för webben krävs det tekniker med stora krav på förändringsbarhet. Det gäller att de tekniker som används ger utrymme till att dynamiskt modifiera sidor, och att alltför statiska lösningar undviks då dessa oftast resulterar i att senare uppdateringar försvåras. Andra krav som ställs är kraven på hastighet. För att en teknik skall accepteras räcker det sällan med att den endast bidrar till en dynamisk lösning. Den skall även öka prestanda hos applikationen. En teknik som stämmer väl in på dessa krav är CSS. CSS är ett språk som kompletterar HTML på de punkter där HTML är som svagast, nämligen designdelen.

I det här arbetet används en exempelsida helt konstruerad med HTML. Genom arbetet konverteras denna sida till en lösning innehållande en HTML-del och en CSS-del. Syftet är att studera vilka fördelar en sådan konvertering resulterar i. Rapporten visar att CSS fungerar mycket väl som en komplettering av HTML.

Nyckelord: Förändringsbarhet, CSS, HTML

Innehållsförteckning

1	Introduktion	1
2	Bakgrund.....	2
2.1	HTML	2
2.2	CSS	3
3	Problemområde.....	4
3.1	Problemprecisering	4
3.2	Avgränsning	4
4	Metod.....	6
5	Genomförande.....	7
5.1	Utgångssidan	7
5.2	Rensning av onödiga HTML-element	9
5.3	Återskapa designen med CSS	11
5.3.1	Globala stilar	11
5.3.2	Menyn	12
5.3.3	Recensionsrubrik och Summering.....	13
5.3.4	Informationsbox	14
5.3.5	Recensionstext och citat.....	15
5.3.6	Bildrubrik och subtitel	16
6	Resultat.....	18
7	Slutsats.....	19
7.1	Diskussion	19
7.1.1	Metoden	19
7.1.2	Resultatet.....	19
7.2	Fortsatt arbete.....	20
	Referenser	21
	Bilagor	22
	Bilaga 1 – Kod för det ursprungliga dokumentet	
	Bilaga 2 – Reducerad HTML-kod	
	Bilaga 3 – Slutgiltig implementation	

1 Introduktion

Det finns ett stort antal verktyg som kan användas för att konstruera dagens krävande webbplatser. De krav som främst ställs på dessa verktyg är att kunna arbeta på ett dynamiskt sätt. Där uppdateringar och vidareutveckling av projektet kan genomföras effektivt. Två tekniker som effektivt kompletterar varandra och som används flitigt i dagens webbsidor är HTML och CSS. HTML är en äldre teknik än CSS och kom till för att på ett effektivt sätt kunna publicera forskarrapporter på WWW. Detta användningsområde gör att HTML är starkt begränsat i vilka grafiska förändringar som kan göras. Rapporterna som publicerades var ofta uppbyggda kring fyra typer av element; paragrafer, rubriker, listor samt länkar. Dessa fyra element har under senare tid kommit att bli otillräckliga för att skapa en tillfredställande design på en modern hemsida. För att öka användningsområdet för HTML har nyare versioner släppts som innehåller ett bredare utbud av designmässiga funktioner. Dessa senare funktioner har dock visats vara bristande på ett flertal punkter. En av dessa punkter är att webbläsarna har svårigheter att tolka specifikationerna för språket vilket har resulterat i att samma design kan skilja i presentationen hos olika webbläsare. En annan punkt är att de verktyg som levereras med nyare versioner av HTML skapar en statisk webbsida som är svår att modifiera och uppdatera. En stor del av det CSS bidrar med är att skapa en fristående del i HTML-dokumentet i vilken samtlig information som behandlar hur dokumentet skall presenteras grafiskt samlas.

Det denna rapport syftar till att undersöka är vilka förbättringar som kan uppnås genom att ta en befintlig sida, helt skriven i HTML, och konvertera den till en sida som består av en CSS-del samt en HTML-del. Rapporten kommer främst att undersöka om en separering mellan *vad* som skall visas och *hur* detta visas kan öka en sidas förändringsbarhet samtidigt som laddningstiden för sidan minskar.

Rapporten är riktad till personer med intressen inom webbdesign. Både grafiker och programmerare kommer att finna material som kan användas för att eventuellt förbättra befintliga lösningar. Rapporten är skriven på en nivå som förutsätter att läsaren har en grundläggande förkunskap om de tekniker som används i rapporten.

2 Bakgrund

Under senare delen av 1990-talet var webben fortfarande ett relativt outforskat medium och ingen visste egentligen vad användningsområdet skulle komma att bli. När något skulle konstrueras för webben fanns inga direkta regler för hur detta skulle ske. Detta medförde att skaparna använde sina egna designregler, vilka innebar att de var fria att göra i princip vad som helst. Allteftersom webben växte ökade behovet av mer komplexa lösningar vilka i sin tur ökade kraven på att tillverkarna skapade mer kompetenta webbläsare. Då de olika tillverkarna av webbläsare började konstruera sina egna standarder, som tolkade samma design på olika sätt, blev problemet tydligt för webbutvecklarna. Problemet var att en design som är konstruerad för webben skall se likadan ut oberoende av vilken webbläsare som används eller vilken plattform som kör denna webbläsare på. Så som läget var då, och till stor del även är idag, fanns inga garantier på hur en design kom att visas på olika webbläsare. För att enas om en standard startades en organisation av webbutvecklare som kallades WaSP (Web Standards Project) vilka samarbetade med dåtidens största tillverkare av webbläsare. Det WaSP arbetade för var att övertyga tillverkarna om att bättre följa de regler som blivit specificerade av W3C (World Wide Web Consortium). Genom att övertyga tillverkarna att enas om samma standarder kunde webbutvecklare skapa en mer pålitlig design som krävde mindre kod då det inte längre var nödvändigt att skapa olika design för alla olika webbläsare. (Shea, Holzschlag, 2005).

Två av dessa tekniker som idag utgör en grundläggande del av webbdesign är HTML (HyperText Markup Language) och CSS (Cascading Style Sheets).

2.1 HTML

HTML är ett standardformat för webbsidor som presenteras på WWW (World Wide Web). Den senaste versionen av HTML är 4.01 och blev en godkänd standard i december 1999 (W3C, 2006a). Detta är också den sista versionen som baseras på SGML (Standard Generalized Markup Language) (W3C, 2004). Senare versioner av HTML kallas XHTML (Extensible HyperText Markup Language) (W3C, 2006a) och baseras på XML (eXtensible Markup Language) (W3C, 2006b).

Det ursprungliga användningsområdet för HTML var att formatera dokument, så som forskarrapporter, för att göra dem möjliga att läsa via WWW. Senare versioner av HTML har dock implementerat stöd för grafisk presentation.

HTML byggs upp med hjälp av så kallade taggar där varje tagg representerar en viss del av dokumentet. I taggen kan sedan attribut anges. Dessa attribut definierar hur taggen kommer att presentera sin del av dokumentet. Varje tagg som skrivs i HTML måste inneslutas inom `<>` och sedan avslutas med `</>`. Som ett exempel kan taggen `` användas, vilken anger att texten mellan klamrarna kommer presenteras på ett speciellt sätt. Hur texten kommer presenteras avgörs av attributen. Lägg till exempel attributet "color" till så att taggen ser ut som i exemplet nedan, kommer texten "HTML" att få en rödaktig färg samt ha teckenstorleken som är definierad för `<h2>`-taggen.

```
<html>
<head>
</head>
<body>
```

```
<h2><font color="#600">HTML</font></h2>
</body>
</html>
```

2.2 CSS

Som nämns i avsnitt 2.1 tillkom det i senare versioner av HTML möjligheter att manipulera utseendet av dokumentet såsom textfärger, textfonter, marginalbredd och så vidare. Detta är dock områden som inte hör hemma i HTML-språket, utan är något som är bättre lämpat för CSS.

CSS är ett språk som definierar hur stilmallar skrivs, det vill säga dokument som beskriver hur olika taggar på en webbsida skall återges. Detta kan avse rubriker som skall visas i fetstil eller brödtext som skall använda någon speciell font. CSS finns i två versioner (CSS1, CSS2). CSS2 bygger på CSS1. CSS2 är bakåtkompatibelt i största utsträckning. Detta betyder att CSS1 kod som redan är skrivet också är godkänd CSS2 kod. Stora fördelar med CSS2 är att den stödjer mediaspecifik användning. Detta innebär att stilmallar kopplas beroende av vilket media som skall presentera data. En stilmall kan användas när data skrivs ut på en skrivare och en annan stilmall kan användas då sidan visas i webbläsaren. En stilmall som är skapad för ett HTML-dokument kan användas på två sätt. Det första är som visas i exemplet nedan och det är även denna metod som använts för exemplet i bilaga 3. Den andra metoden är att använda en extern CSS-fil där samtliga stilmallar placeras. Sedan länkas denna fil till HTML-dokumentet. Den stora fördelen med den senast nämnda metoden är att en ändring i den externa filen uppdaterar samtliga HTML-dokument som har filen länkad till sig. I den först metoden krävs ett manuellt arbete med att gå igenom samtliga HTML-dokument och ändra stilmallarna för varje om en uppdatering skall ske på flera sidor. (W3C, 2006c)

Exemplen i denna rapport kommer uteslutande behandla CSS1. I exemplet nedan visas en variant som kan användas för att återskapa exemplet i avsnitt 2.1. I detta exempel används CSS för att definiera färgen på texten.

```
<html>
<head>
<style type="text/css">
h2 (color: #600;)
</style>
</head>
<body>
<h2>HTML</h2>
</body>
</html>
```

3 Problemområde

Internet har under det senaste årtiondet vuxit sig så starkt att det idag har en självklar plats i många människors liv. Folk använder idag Internet för att maila, läsa dagstidningar, handla mat, se på film och utnyttjar ett nästan oändligt utbud av tjänster. En stor del av dessa tjänster erbjuds via WWW, vilket är ett nätverk av servrar som stödjer speciellt formaterade dokument. Dessa dokument är oftast formaterade med språket HTML. Med HTML har utvecklaren möjlighet att konstruera sidor som stödjer ett strukturerat sätt att presentera information på. Det finns även ett begränsat stöd för hur informationen visuellt skall presenteras med färger, bilder och övrig grafik. Denna del av presentationen kommer genom rapporten att refereras till som enbart design. När HTML används för att skapa designen får oftast utvecklaren använda sig av metoder som inte är menade att användas för design vilka ofta får konsekvenser enligt nedanstående punkter.

- Ostrukturerad och svårläst kod
- Längre kod samt fler referenser till objekt vilka bidrar till onödigt stora filer och längre nedladdningstider av dokumentet
- Oklarheter kring kompatibiliteten hos webbläsare vilket medför att utvecklare inte kan vara säkra på hur designen kommer att se ut på olika tillverkares webbläsare
- Minskad förändringsbarhet

(Myer, 2002)

För att lösa ovanstående problem skulle ett nytt språk behöva konstrueras vilket kompletterar HTML genom att separera innehållet från designen. Ett sådant språk är CSS. Med CSS används stilmallar (se avsnitt 2.2) för att designa innehållet och kan på så sätt förbättra flera aspekter genom att skilja information från presentation.

3.1 Problemprecisering

Då det ligger mycket arbete bakom att konvertera en befintlig HTML-sida till en sida som både använder HTML och CSS, är det av intresse att undersöka fördelarna som uppnås genom en konvertering.

Arbetet bakom denna rapport syftar till att undersöka hur följande egenskaper påverkas av en konvertering:

- Antal bildobjekt på sidan som inte bidrar med någon visuell information
- Antal nästlade tabellstrukturer som används för ändamål andra än vad som är syftet med tabeller
- Storlek på filen

3.2 Avgränsning

Exemplet i rapporten visar bara på ett specifikt användningsområde av CSS. Tillämpningarna är betydligt fler än vad som beskrivs i rapporten.

Rapporten syftar inte till att ge läsaren en fullständig insikt i hur webbstandarder kan användas för att skapa en pålitlig design. Den är heller inte menad som någon manual

Införande av CSS i existerande tabellbaserat HTML-dokument

över vilka olika standarder som finns framtagna av W3C och hur dessa är konstruerade, då utbudet är enormt. För detta hänvisas läsaren till W3C:s hemsida <http://www.w3.org>.

4 Metod

Som metod har proof of concept valts. Genom ett antal välkända böcker inom området webbdesign (Myer, 2002), (Myer, 2004), (Cederholm, 2004), (Shea, Holzschlang, 2005) har kända exempel hämtats av sidor som är helt baserade på HTML. I dessa sidor har sedan svagheter i implementationen identifierats. Dessa svagheter har sedan undersökts för att avgöra vilka fördelar en konvertering till CSS skulle ge. Det var främst en egenskap som var av extra intresse för en konvertering till CSS vid denna undersökning. Denna egenskap var den statiska tabellstrukturen. Genom rapporten kommer HTML-sidan att konverteras till en lösning innehållande en HTML-del för innehåll och en CSS-del för presentation. Konverteringen kommer att starta med att skapa en ny version av original HTML-dokumentet där samtliga koder som inte är önskvärd efter konverteringen tas bort. Detta ger en starkt reducerad kod och en sida med minimal presentation. Efter att detta minimala dokument har skapats delas det in i åtta stycken delar vilka ligger i grunden vid konverteringen till CSS. Huruvida denna konvertering blir lyckad eller inte avgörs genom värdering av de angivna punkterna i problempreciseringen (3.1). De två första av dessa punkter kommer att bedömas genom att studera den resulterande koden och manuellt räkna antal förekommande instanser. Den fjärde punkten kommer att bedömas genom att jämföra hur många bytes de olika kodfilerna upptar.

5 Genomförande

I detta avsnitt beskrivs hur exempelsidan skapades. De första punkterna i avsnittet beskriver hur sidan är konstruerad med HTML. Sedan beskrivs i detalj hur konverteringen till en sida med CSS har genomförts.

5.1 Utgångssidan

Som utgångssida används en helt HTML-baserad sida. Fullständig kod för sidan finns under bilaga 1. I fig. 5.1 visas hur sidan ursprungligen ser ut.

The screenshot shows a web page with a header containing a logo and the title 'SMARTARE CSS NU'. Below the title is a subtitle 'Så bemästras CSS' and a navigation menu with links for 'info', 'datalogi', 'profil', 'sök', and 'kontakt'. On the left side, there is a sidebar box with contact information: 'Ämne: Datalogi', 'Teknik CSS', 'Skola: HIS', 'Högskolan i Skövde', '0500-44 80 00', 'Info | Karta', and 'Henrik Larsson Hemsida'. The main content area features a red heading 'CSS' followed by a summary: 'Summering: Snyggare, snabbare, roligare'. Below this is a paragraph explaining CSS as a W3C recommendation. To the right of this paragraph is a red text box with the quote: 'Med den rätta tekniken kan du förändra ditt sätt att designa hemsidor drastiskt.' At the bottom, there is a source attribution: 'Källa: Susning.nu'.

Fig. 5.1 Originalsidans utseende utan några stilmallar.

För att lättare kunna referera till de olika delarna av sidan delas den upp i åtta delar enligt nedan:

1. I översta vänstra hörnet på sidan finns en bild med innehållet "SMARTARE CSS NU". Denna kommer att refereras till som bildrubrik.
2. Strax under bildrubriken finns en annan bild "Så bemästras CSS". Denna refereras till som subtitel.
3. Till höger om sidrubriken finns fem stycken ord vilka tillsammans refereras till som meny.
4. På sidans vänstra del, nedanför sidrubriken, finns en box innehållande information om ämne, teknik och så vidare. Denna refereras till som informationsbox.
5. Till höger om informationsboxen finns en röd text "CSS". Denna refereras till som recensionsrubrik.

Införande av CSS i existerande tabellbaserat HTML-dokument

6. Under recensionsrubriken finns en summering som säger ”Snyggare, snabbare, roligare”. Denna refereras till som summering.
7. Det stora textstycket på sidan kallas recensionstext.
8. Inuti recensionstexten finns ett stycke text som markeras med större och grövre teckensnitt. Denna text refereras till som citat.

För att kunna konvertera sidan måste först den nuvarande designen identifieras. Det som kan identifieras genom att studera koden i bilaga 1 är att sidan är uppbyggd kring <table>-element. För att identifiera hur dessa <table>-element är konstruerade skapas en temporär stilmall till dokumentet för att markera tabellernas kanter. (Myer, 2004)

```
table {border: 2px solid red; margin: 2px;}  
td {border: 1px dotted purple; padding: 1px}
```

Den första av de två reglerna ramar in samtliga <table>-element med en två pixlars bred röd ram samt ger en två pixels bred marginal mellan <table>-elementen. Den andra regeln ger alla <td>-element en ram bestående av prickar (en pixel bred) samt en spaltfyllnad om en pixel. I Fig. 5.2 visas resultatet av stilmallen.

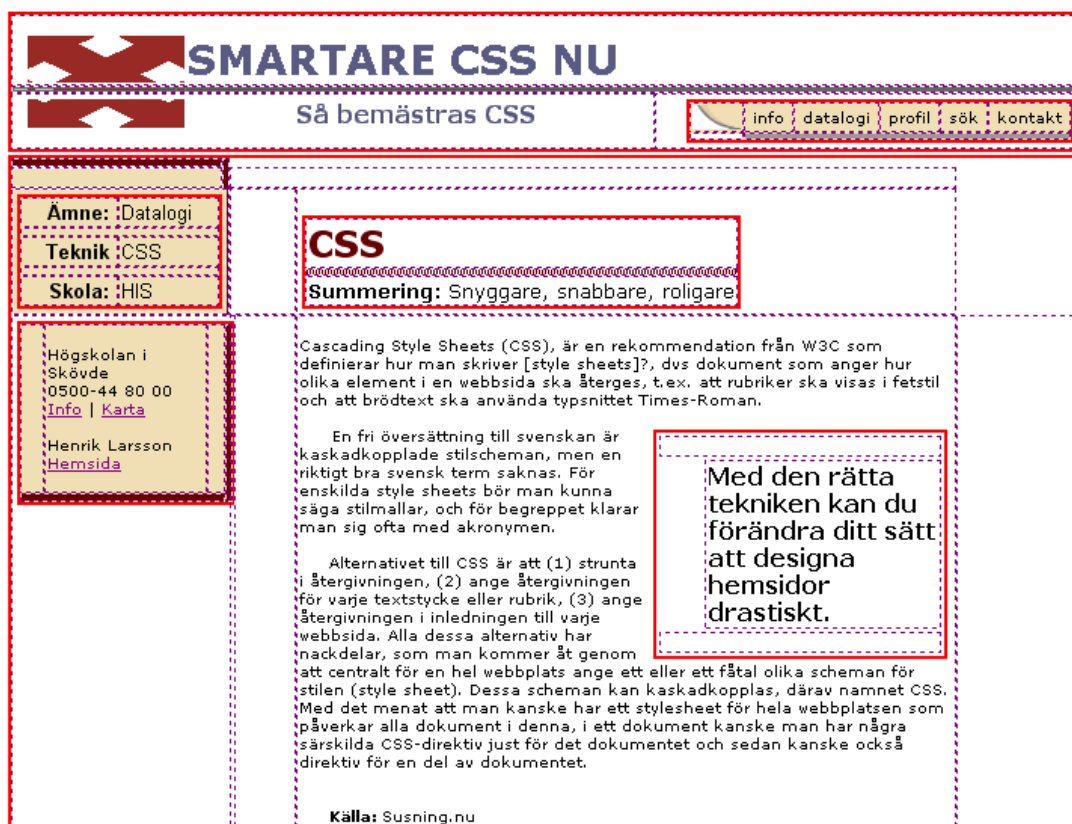


Fig. 5.2 Originalsidan med stilmall för att markera tabellerna i designen

Genom att studera Fig. 2 fås en överblick över hur sidan är konstruerad. Sidrubriken är innesluten i ett `<table>`-element medan resten av sidan befinner sig i ett andra `<table>`-element. Inuti detta andra element finns vidare nästlingar av `<table>`-element.

5.2 Rensning av onödiga HTML-element

För att på ett effektivt sätt konvertera en sida från enbart HTML-kod till HTML-kod och CSS-kod krävs att två saker görs. Först måste originalkoden rensas från samtlig HTML-kod som kommer att bli överflödigt i CSS-designen. I rapporten beskrivs en arbetsmetod som startar med att ta bort all "onödig" HTML-kod från dokumentet. Efter detta kommer varje del av de åtta punkterna som listats under rubrik 5.1 att behandlas separat.

Att implementera CSS på en HTML-sida innebär att så mycket som möjligt av den HTML-kod som används för att påverka sidans utseende skall tas bort. Som första steg för att göra detta kommer elementen i nedanstående lista att tas bort ut HTML-dokumentet.

- ``
- `
`
- ` `
- Samtliga attribut på `<body>`-element.

Nästa steg blir att rensa upp i tabellstrukturen tills en struktur som den nedan erhålls.

```
<TABLE>
  [...bildrubrik...]
  [...subtitel...]
</TABLE>
<DIV id="meny">
  [...menyn...]
</DIV>
<DIV id="recension">
  [...recensionsrubrik...]
  [...summering...]
<DIV id="infobox">[... informationsboxen...]</DIV>
  [...recensionstext...]
  [...citat]
</DIV>
```

Som ses i strukturen ovan lämnas bildrubriken och subtiteln kvar i en tabell. Detta är en temporär lösning. Målet med konverteringen var ju att eliminera denna typ av tabell. Därför kommer tabellen att senare i rapporten tas bort men får för tillfället vara kvar för att underlätta konverteringen.

Införande av CSS i existerande tabellbaserat HTML-dokument

Efter att ha minimerat HTML-koden görs vissa förändringar i dokumentet för att göra det resulterande dokumentet enklare att konvertera. Menyn har fått ett lite annat utseende då tabellen som inneslöt varje element har tagits bort. Detta innebär att samtliga länkar i menyn har flutit samman och är mycket svåra att särskilja. För att skilja länkarna infogas vertikala linjer (|) inneslutna i ett ``-element mellan varje länk. Koden för den nya menyn visas nedan.

```
<div id="meny">
  <a href="">info</a><b>|</b>
  <a href="">datalogi</a><b>|</b>
  <a href="">profil</a><b>|</b>
  <a href="">sök</a><b>|</b>
  <a href="">kontakt</a>
</div>
```

Resterande delen av dokumentet är efter konverteringen inneslutet i ett `<div>`-element som har ersatt den tidigare nästlade `<table>`-designen.

```
<div id="recension">
  <h2>CSS</h2>
  <p id="summering"><strong>Summering:</strong>
    Snyggare, snabbare, roligare</p>
  <div id="infobox">[... informationsboxen ...]</div>
  [...recensionstext...]
  [...citat...]
</div>
```

Den sista förändringen som har tillförts i den nya minimala versionen av HTML-sidan är förändringen av citatet. Detta är nu inneslutet i ett så kallat `<blockquote>`-element.

```
<blockquote class="citat">
```

Med den rätta tekniken kan du förändra ditt sätt att designa hemsidor drastiskt

```
</blockquote>
```

I Fig. 3 visas hur sidan ser ut efter ovanstående reduceringar i koden. I detta exempel framgår det tydligt att HTML har använts för vad det är menat att användas till, nämligen innehåll. Innehållet har strukturerats väl men presenteras på ett tråkigt sätt. Det är där CSS kommer att hjälpa genom att tillföra verktyg som behövs för att skapa en sida som är mer attraktiv att besöka ur ett designmässigt synsätt. Koden för sidan i Fig. 5.3 finns i bilaga 2.



[info](#) | [datalogi](#) | [profil](#) | [sök](#) | [kontakt](#)

CSS

Summering: Snyggare, snabbare, roligare

Ämne: Datalogi

Teknik: CSS

Skola: HIS

- Högskolan i Skövde
- 0500-44 80 00
- [Info](#) | [Karta](#)

- Henrik Larsson
- [Hemsida](#)

Cascading Style Sheets (CSS), är en rekommendation från W3C som definierar hur man skriver [style sheets]?, dvs dokument som anger hur olika element i en webbsida ska återges, t.ex. att rubriker ska visas i fetstil och att brödtext ska använda typsnittet Times-Roman.

Med den rätta tekniken kan du förändra ditt sätt att designa hemsidor drastiskt.

En fri översättning till svenskan är kaskadkopplade stilscheman, men en riktigt bra svensk term saknas. För enskilda style sheets bör man kunna säga stilmallar, och för begreppet klarar man sig ofta med akronymen.

Alternativet till CSS är att (1) strunta i återgivningen, (2) ange återgivningen för varje textstycke eller rubrik, (3) ange återgivningen i inledningen till varje webbsida. Alla dessa alternativ har nackdelar, som man kommer åt genom att centralt för en hel webbplats ange ett eller ett fåtal olika scheman för stilen (style sheet). Dessa scheman kan kaskadkopplas, därav namnet CSS. Med det menat att man kanske har ett stylesheet för hela webbplatsen som påverkar alla dokument i denna, i ett dokument kanske man har några särskilda CSS-direktiv just för det dokumentet och sedan kanske också direktiv för en del av dokumentet.

Källa: Susning.nu

Fig. 5.3 Originalsidan efter upprensning i koden

5.3 Återskapa designen med CSS

Det som nu återstår att arbeta med är ett dokument som endast använder HTML för designen av dokumentet. Under denna rubrik kommer originalsidan att återskapas men istället för att använda HTML för designen kommer CSS att användas.

Under arbetets gång kommer varje del av dokumentet att diskuteras separat och referenser till HTML-versionen av originalsidan kommer att göras.

5.3.1 Globala stilar

Innan sidan börjar designas kommer några globala stilar (stilar som gäller för hela dokumentet) att sättas. Den första är en regel som skall återskapa textlayouten som i HTML versionen motsvaras av ``. Den närmast ekvivalenta storleken i CSS är *x-small*.

```
body {font: x-small Verdana, Arial, Helvetica, sans-serif;}
```

```
a.navlink {text-decoration: none;}
```

Som en del i konvertering togs alla attribut bort på body-elementet. Bland dessa attribut fanns marginheight och topmargin. Dessa attribut finns för att ta bort det tomma utrymme som blir runt om dokumentet. För att få samma effekt med CSS krävs att både margin och padding sätts till noll. Anledningen till att båda attributen sätts till noll är för att olika webbläsare väljer att tolka den ena men inte den andra.

```
body {font: x-small Verdana, Arial, Helvetica, sans-serif; margin: 0; padding: 0}
```

```
a.navlink {text-decoration: none;}
```


Införande av CSS i existerande tabellbaserat HTML-dokument

```
#meny {position: absolute; top: 44px; right: 0; padding: 2px 0 2px 32px; white-space: nowrap; background: #F0DFB4 url(leftcurveimg.gif) bottom left no-repeat;}
```

Som beskrivs under rubrik 5.2.1 skapades en vertikal linje för att särskilja länkarna. Dessa är dock inte önskvärda i den nya version men är bra att ha kvar om någon besökare använder en webbrowser som inte stödjer CSS. Därför används -elementet för att göra strecken osynliga för webbläsare som stödjer CSS.

För att inte länkarna skall vara för ihopklämda i menyn skapas även utrymme runt dem.

```
#meny b {display: none;}
```

```
#meny a {text-decoration: none; color: #000; padding 0 1em 0 0;}
```

Det enda som återstår för att återskapa menyn är att rama in den genom att skapa en svart linje i nederkant. Detta görs enklast genom att ge varje länk i menyn ett svart streck i nederkanten. För att göra detta streck parallellt med strecket i bilden som finns i menyns vänstra hörn behövs även en justering av bottenmarginalen med 1 pixel.

```
#meny a {text-decoration: none; color: #000; border-bottom: 1px solid gray; padding 0 1em 1px 0;}
```

Med denna justering har ett resultat uppnåtts som mycket väl stämmer överens med originalsidas design (fig. 5.5). Fördelen med den nya designen är att den är mycket enkel att flytta runt då den är helt oberoende av övriga sidan. I originalsidan var menyn konstruerad i en tabell och var därför mycket svår att flytta.

5.3.3 Recensionsrubrik och Summering

För att introducera recensionen används en mycket enkel presentation av rubriken samt summeringen. Dessa skiljer sig mycket från den formatering de hade på originalsidan. Nedan visas den HTML-kod som användes för recensionsrubriken och summeringen på originalsidan.

```
<h2>CSS</h2>
```

```
<p id="summering"><strong>Summering: </strong> Snyggare, snabbare, roligare</p>
```

För att återskapa utseende på rubriken krävs inte stora förändringar. Det som behövs göras är att ändra storlek samt färg på texten. För att kompensera för de utfyllningsbilder som togs bort vid konverteringen skapas även en marginal runt titel så att utrymme lämnas för summeringstexten.

Införande av CSS i existerande tabellbaserat HTML-dokument

```
#recension h2 {color: #600; font-size: x-large; margin: 1em 0 0;}
```

De förändringar som skall göras för summeringen är att ändra storlek på texten. Det behövs även en horisontell linje över summeringen som skiljer titeln från summeringen.

```
#recension #summering {font-size: small; border-top: 1px dotted #600; display: inline;}
```

5.3.4 Informationsbox

Informationsboxen har blivit rejält förändrad efter konverteringen och det är svårt att föreställa sig hur den en gång såg ut. Till att börja med krävs en omplacering av den vilken kan göras enligt samma princip som användes för menyn. Informationsboxen skall placeras på sidans vänstra del, precis under bildrubriken. Bildrubriken är 72 pixlar hög. Alltså sätts informationsboxen 73 pixlar nedanför övre kanten av dokumentet. Även en justering av bakgrundsfärg och bredd görs.

```
#infobox {position: absolute; top: 73px; left: 0; width 140px; background: #F0DFB4;}
```

Nästa steg blir att skapa en ram runt informationsboxen samt att skapa en marginal mellan boxens kant och innehållet i den.

```
#infobox {position: absolute; top: 73px; left: 0; width 140px; background: #F0DFB4; padding: 0.75em 0; border: 1px solid #600; border-width: 2px 1px 2px 0;}
```

Punktlistan inuti boxen skiljer sig fortfarande från originalet. Skillnaden är att i originalet fanns inga punkter för markerade alternativen. Dessa punkter behöver därför tas bort samtidigt som indenteringen bör försvinna.

```
#infobox ul {list-style: none; margin: 1em; padding: 0;}
```

Det sista som kvarstår är att strukturera den tabell som utgör den övre delen av boxen. De förändringar som skall genomföras är att högerjustera texten, öka utrymmet mellan texten och att ändra teckensnitt.

```
#infobox th {text-align: right;}
```

```
#infobox td {padding: 0.125em;}
```

```
#infobox table {font: small Arial, Verdana, Helvetica, sans-serif;}
```

I fig. 5.4 visas hur sidan ser ut efter de beskrivna förändringarna.

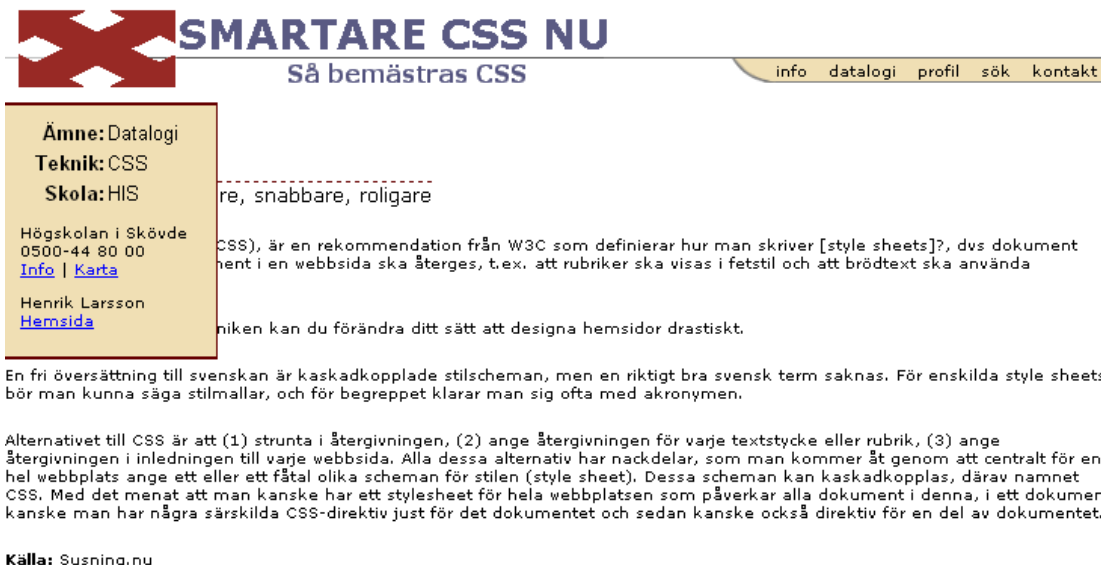


Fig. 5.4 Problem med överlappning

Som syns i figur 5.4 överlappar informationsboxen recensionstexten vilket inte kan tillåtas. I nästa rubrik kommer detta problem att behandlas.

5.3.5 Recensionstext och citat

För att skapa utrymme kring texten samt för att skapa en marginal tillräckligt stor för att informationsboxen inte skall skymma texten ges 180 pixlars vänstermarginal på texten. På höger sida sätts 80 pixlars marginal och i nederkant 2em (em motsvarar den använda fontens höjd).

```
#recension {margin: 0 80px 2em 180px;}
```

På originalsidan var varje paragraf utom den första indenterad. Som syns i fig. 5.4 finns inte denna indentering kvar. För att återskapa denna effekt krävs följande regel:

```
#review p {text-indent: 2em;}
```

Problemet med denna regel är att den skapar en indentering även på den första paragrafen i texten samt summeringen. För att skapa en individuell regel för just det <p>-element som innesluter den första paragrafen i texten läggs en class till för detta element.

```
<p class="paragraf1">
```

Nu kan en regel skrivas som är specifik för detta element.

Införande av CSS i existerande tabellbaserat HTML-dokument

```
#recension .paragraf1 {text-indent: 0;}
```

För att få bort indenteringen på summeringen skrivs dess regel om.

```
#recension #summering {font-size: small; border-top: 1px dotted #600; display: inline; text-indent: 0;}
```

Citatet skapades på originalsidan genom en avancerad hopsättning av tabeller vilket gjorde resultatet mycket statiskt och svårt att flytta. För att återskapa detta med CSS krävs endast en regel och resultatet blir en betydligt mer dynamisk lösning som kan flyttas utan att påverka övriga delar på sidan.

```
blockquote .citatt {float: right; width: 40%; padding: 1em 0 1em 5%; margin 0; font-size: medium; font-weight: bold;}
```

5.3.6 Bildrubrik och subtitel

Anledning till att bildrubriken är innesluten i tre separata ramar är att den genomgående linjen som sträcker sig tvärs igenom bildrubriken skall visas korrekt. Som syns i koden i bilaga 1 är denna linje inget annat än en bakgrundsbild som är innesluten i den mittersta tabellen. För att skapa denna effekt har huvudbilden blivit uppdelad i tre delar vilket gör den onödigt komplicerad att hantera. I CSS-versionen delas bilden upp i två delar. Dels kommer huvudrubriken, ”SMARTARE CSS NU”, att hamna i en bild och sedan underrubriken, ”Så bemästras CSS”, i en egen. Anledningen till att placera underrubriken i en egen bild är att göra den enkel att positionera på sidan. Nedan visas den nya implementationen av huvudrubriken då tabellstrukturen har blivit borttagen.

```
<div id="bildrubrik">  
  
  
</div>
```

Genom att räkna antalet pixlar i originaldesignen kan positionen för subtiteln bestämmas.

```
#subtitel {position: absolute; top 46px; left: 151px;}
```

Det som nu återstår för att fullständigt ha konverterat sidan till CSS-design är att återskapa det vertikala streck, som löper genom sidan högst upp. För att göra detta används en svart bild med storleken 1x1 pixel som sedan itereras genom bilden i bildrubriken.

```
#bildrubrik {background: white url(graypixmap.gif) 0 43px repeat-x;}
```

SMARTARE CSS NU
Så bemästras CSS

info datalogi profil sök kontakt

Ämne: Datalogi
Teknik: CSS
Skola: HIS

Högskolan i Skövde
0500-44 80 00
[Info](#) | [Karta](#)
Henrik Larsson
[Hemsida](#)

CSS

Summering: Snyggare, snabbare, roligare

Cascading Style Sheets (CSS), är en rekommendation från W3C som definierar hur man skriver [style sheets]?, dvs dokument som anger hur olika element i en webbsida ska återges, t.ex. att rubriker ska visas i fetstil och att brödtext ska använda typsnittet Times-Roman.

En fri översättning till svenskan är kaskadkopplade stilscheman, men en riktigt bra svensk term saknas. För enskilda style sheets bör man kunna säga stilmallar, och för begreppet klarar man sig ofta med akronymen.

Alternativet till CSS är att (1) strunta i återgivningen, (2) ange återgivningen för varje textstycke eller rubrik, (3) ange återgivningen i inledningen till varje webbsida. Alla dessa alternativ har nackdelar, som man kommer åt genom att centralt för en hel webbplats ange ett eller ett fåtal olika scheman för stilen (style sheet). Dessa scheman kan kaskadkopplas, därav namnet CSS. Med det menat att man kanske har ett stylesheet för hela webbplatsen som påverkar alla dokument i denna, i ett dokument kanske man har några särskilda CSS-direktiv just för det dokumentet och sedan kanske också direktiv för en del av dokumentet.

Med den rätta tekniken kan du förändra ditt sätt att designa hemsidor drastiskt.

Källa: Susning.nu

Fig. 5.5 Slutgiltigt resultat efter konvertering till CSS

6 Resultat

Efter genomförd konvertering kan erhållna resultat ses i tabell 6.1. Nedan följer ett förtydligande av innehållet i tabellen.

På rad ett i Tabell 6.1 syns originalsidan. Rad två visar den version av sidan där HTML och CSS är skrivet i samma dokument. Rad tre visar en teknik som kallas för länkad CSS. Med länkad CSS menas att CSS och HTML separeras i två olika filer. Sedan länkas CSS filen in i HTML dokumentet genom nedanstående kod:

```
<link rel="stylesheet" type="text/css" href="externfil.css">
```

Denna teknik medför två förbättringar:

1. Genom att länka in CSS filen i HTML-dokumentet kan samtliga HTML-sidor som använder det länkade CSS-dokumentet uppdateras samtidigt. Det räcker att ändra reglerna i CSS-filen och på så sätt kommer samtliga HTML-dokument som använder den regeln att påverkas.
2. Flertalet webbläsare kan konfigureras så att de cachar externa CSS-filer. Detta medför att antalet serverförfrågningar minskar efter att CSS-filen blivit laddad första gången. Detta är anledningen till att antal serverförfrågningar på rad tre, kolumn 3 kan variera beroende av om CSS-filen är cachad eller ej.

Metod	ASCII tecken	Server förfrågningar	Tabeller	Bilder
HTML	5986	6-23	7	22
HTML + CSS	3742	5	1	4
HTML + länkad CSS	2405 + 1356	5-6	1	4

Tabell 6.1: Jämförelse mellan två olika CSS implementationer och den ursprungliga HTML implementationen

7 Slutsats

Efter genomfört arbete kan följande slutsatser dras:

- En stark reducering av antalet bildreferenser är gjord vilket leder till snabbare laddningstider
- Borttagning av statiska tabellstrukturer som har ersätts med en mer dynamisk <div>-element struktur medför en sida som enklare kan uppdateras och underhållas
- Storleken på HTML-filen har blivit reducerad med upp till 60 % (Tabell 6.1)
- Den visuella överensstämmelsen mellan originalsidan och den konverterade sidan är mycket överensstämmande. Endast några små ändringar i pixelplaceringarna har gjorts

En konvertering från ett tabellbaserat HTML-dokument till ett dokument med CSS har gett många fördelar vilket ger slutsatsen att en konvertering är värd arbetet och pengarna. Denna slutsats baseras på resultatet i tabell 6.1.

7.1 Diskussion

Under diskussion kommer metoden och resultatet för arbetet att diskuteras.

7.1.1 Metoden

Alla sidor på WWW är mer eller mindre unikt konstruerade. Vissa konstruktioner tjänar mer på att implementera CSS än vad andra gör. Den konstruktion som används i rapporten som exempel har en ganska grundläggande konstruktion. Den visar dock på flera övergripande fördelar som de flesta konstruktioner kommer att dra nytta av genom en konvertering till CSS, såsom positionering av objekt, texthantering, bildhantering. Kanske skulle en konstruktion ha använts som visade på ett bredare användningsområde för CSS. Detta skulle dock troligen inte tillföra så mycket i rapporten, då stora delar av exemplet inte skulle bidra med någon inverkan hos de punkter som specificerats under problemprecisering (avsnitt 3.1).

7.1.2 Resultatet

Resultatet ger en bra bild över hur de aspekter som ytterst påverkar en sidas laddningstid har förbättrats genom konverteringen. Det som dock inte framgår i resultatdelen är hur väl den konverterade sidan visuellt sett överensstämmer med originalsidan. Detta har valts att lämnas utanför rapporten eftersom det är mycket svårt att mäta visuell överensstämmelse.

Genom att konvertera originalsidan till en sidan innehållande både HTML och CSS erhålls en strukturerad kod. Den resulterande koden är betydligt enklare att söka igenom för att finna felaktigheter samt att uppdatera med ny information.

Andra fördelar som den nya designen ger är att sidan nu är mer dynamisk i designen. I den gamla versionen där sidan var uppbyggd kring en nästlad tabelldesign gavs inte mycket utrymme till förändring i designen. För att kunna flytta ett objekt som var instängt i en tabell krävdes det mycket arbete eftersom det inte räckte att uppdatera endast den delen av tabellen som innehöll det egentliga objektet. Vidare krävde den gamla designen att samtliga HTML-dokument som var baserade på originalsidans design var tvungna att uppdateras om någon förändring gjordes. Den konverterade

sidan lider inte av dessa brister eftersom objekten på sidan är placerade inom positionerbara <div>-element. Varje sådant element kan fritt positioneras på sidan utan att det påverkar något av de övriga objekten. Om sidans design skall uppdateras räcker det med att uppdatera CSS-delen och samtliga dokument som innehåller samma CSS-mall kommer då också att uppdateras.

I exemplet konverteras en tabellbaserad sida till en sida som byggs med <div>-element. Dessa element har inget med CSS att göra. Detta kommer dock ha den inverkan på resultatdelen att kodlängden minskar då ett <div>-element skapas med färre tecken än ett <table>-element. Den filminskning som detta innebär är inte CSS förtjänst vilket inte framgår i resultatdelen.

7.2 Fortsatta arbeten

CSS har ett mycket brett användningsområde och som nämnts tidigare i rapporten visas bara en liten del av användningen. Därför finns det mycket utrymme för fortsatta studier. I vidare studier kan det vara intressant att studera andra exempel med CSS som kan vara fördelaktiga att använda framför de traditionella HTML-teknikerna. Vidare arbeten skulle även kunna behandla i vilken utsträckning skillnaderna mellan CSS1 och CSS2 påverkar resultatet vid en konvertering som den i rapporten.

För att CSS skall fungera krävs kompatibilitet med webbläsaren. Denna kompatibilitet är något som skulle vara intressant att testa för att se vilka webbläsare som bäst följer standarderna för CSS1 och CSS2.

Att konvertera en befintlig tabellbaserad HTML-sida till en sida som använder <div>-element samt CSS stilmallar är en process som kräver mycket arbete. Det kan i många fall vara lönsamt att helt bygga om sidan i stället för att konvertera den. Företag lägger enorma pengar på att konstruera sina webbplatser och det är inte alltid önskvärt, från företaget sett, att konvertera en fullt fungerande sida och betala pengar för det. Därför skulle det vara mycket intressant att undersöka i vilken utsträckning en konvertering som den i rapporten skulle gå att automatgenerera. Som input skulle en statisk tabellbaserad sida läsas in, denna sida skulle sedan analyseras och konvertera den statistiska tabelldesignen till en mer dynamisk sida med <div>-element. Originalsidan och den konverterade sidan skulle sträva efter ett gemensamt visuellt gränssnitt.

I vidare arbeten skulle det vara intressant att studera vilka andra tekniker som finns för att komplettera HTML. Studien kan undersöka vilka tekniker som har motsvarande användningsområde som CSS, samt vilka andra alternativ det finns för att ersätta en statisk tabellstruktur. Detta skulle vara av stort intresse då nya tekniker för att skapa hemsidor ständigt utvecklas. Därför skulle det vara intressant att jämföra hur äldre tekniker som till exempel CSS klarar konkurrensen från de nya teknikerna. Det är inte alltid som nya tekniker endast tillför positiva aspekter. Ofta kräver nya tekniker en tid för att fullt stöd skall hinna implementeras i mukvaror som webbläsare. Det finns alltså ett stort intresse i att undersöka hurvida stöd för nya tekniker bör utvecklas hos webbläsarna eller om en nyare teknik kan ersättas genom smart användning av redan befintliga tekniker.

Referenser

Shea, D., Holzschlag, M., (2005) *The Zen of CSS Design: Visual Enlightenment for the Web* Peachpit Press

Cederholm, D., (2004) *Web Standards Solutions: The Markup and Style Handbook* Springer-Verlag

Myer, E., (2002) *Eric Meyer on CSS: Mastering the language of web design* New Riders

Meyer, E., (2004) *More Eric Meyer on CSS* New Riders

W3C, (2006a), *HyperText Markup Language (HTML) Home Page*, <http://www.w3.org/MarkUp/>, hämtad 2006-06-01

W3C, (2004), *Overview of SGML Resources*, <http://www.w3.org/MarkUp/SGML/>, hämtad 2006-06-01

W3C, (2006b), *Extensible Markup Language (XML)*, <http://www.w3.org/XML/>, hämtad 2006-06-01

W3C, (2006c), *Cascading Style Sheets Home Page*, <http://www.w3.org/Style/CSS/>, hämtad 2006-06-01

Håkon Wium Lie, Janne Saarela, (1999), *Multi-purpose publishing using HTML, XML, and CSS*, <http://www.w3.org/People/Janne/project/paper.html>, hämtad 2006-06-01

Bilagor

Bilaga 1 – Kod för det ursprungliga dokumentet

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title>CSS: 10p Examensarbete Datalogi</title>
</head>
<body marginwidth="0" marginheight="0" marginleft="0" margintop="0">
<table cellspacing="0" cellpadding="0" width="100%">
<tr>
<td colspan="2"></td>
</tr>
<tr>
<td colspan="2" bgcolor="#666666"></td>
</tr>
<tr>
<td></td>
<td valign="top">
<table cellspacing="0" cellpadding="0" align="right" bgcolor="#F0DFB4">
<tr>
<td valign="bottom" align="left" width="10"></td>
<td nowrap><font size="-2">&nbsp;<a href="" class="meny">
<font color="#000000">info</font></a>&nbsp;</font></td>
<td nowrap><font size="-2">&nbsp;<a href="" class="meny">
<font color="black">datalogi</font></a>&nbsp;</font></td>
<td nowrap><font size="-2">&nbsp;<a href="" class="meny">
<font color="black">profil</font></a>&nbsp;</font></td>
<td nowrap><font size="-2">&nbsp;<a href="" class="meny">
<font color="black">sök</font></a>&nbsp;</font></td>
<td nowrap><font size="-2">&nbsp;<a href="" class="meny">
<font color="black">kontakt</font></a>&nbsp;</font></td>
</tr>
<tr>
<td bgcolor="#FFFFFF"></td>
<td bgcolor="#999999" height="1" colspan="5"></td>
</tr>
</table>
</td>
</tr>
</table>
<table cellspacing="0" cellpadding="0" width="100%">
<tr>
<td height="2" bgcolor="#660000"></td>
<td width="1" bgcolor="#660000" rowspan="3"></td>
```

Införande av CSS i existerande tabellbaserat HTML-dokument

```
</tr>
<tr>
<td valign="bottom" bgcolor="#F0DFB4"></td>
<td colspan="2"></td>
</tr>
<tr>
<td valign="bottom" bgcolor="#F0DFB4">
<table cellspacing="0" cellpadding="0" width="100%">
<tr>
<td align="right"><font size="-1" face="Arial, Helvetica, sans-serif"><b>Ämne:&nbsp;&nbsp;&nbsp;</b></font></td>
<td><font size="-1" face="Arial, Helvetica, sans-serif">Datalogi</font></td>
</tr>
<tr>
<td colspan="3" height="2"></td>
</tr>
<tr>
<td align="right"><font size="-1" face="Arial, Helvetica, sans-serif"><b>Teknik&nbsp;&nbsp;&nbsp;</b></font></td>
<td nowrap><font size="-1" face="Arial, Helvetica, sans-serif">CSS</font></td>
</tr>
<tr>
<td colspan="3" height="2"></td>
</tr>
<tr>
<td align="right"><font size="-1" face="Arial, Helvetica, sans-serif"><b>Skola:&nbsp;&nbsp;&nbsp;</b></font></td>
<td><font size="-1" face="Arial, Helvetica, sans-serif">HIS</font></td>
</tr>
</table>
</td>
<td width="40"></td>
<td valign="bottom">
<table cellspacing="0" cellpadding="0" >
<tr>
<td><font size="+2" color="#660000"
face="Verdana, Arial, Helvetica, sans-serif"><b>CSS</b></font></td>
</tr>
<tr>
<td height="1" background="lineimg.gif"></td>
</tr>
<tr>
<td><font size="-1"><strong>Summering:</strong> Snyggare, snabbare, roligare</font></td>
</tr>
</table>
</td>
</tr>
<tr>
<td valign="top" colspan="2">
```


Bilaga 2 – Reducerad HTML-kod

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title>CSS: 10p Examensarbete Datalogi</title>
</head>
<body>
<table cellspacing="0" id="bildrubrik">
<tr>
<td></td>
</tr>
<tr>
<td bgcolor="#666666"></td>
</tr>
<tr>
<td></td>
</tr>
</table>
<div id="meny">
<a href="">info</a><b>|</b>
<a href="">datalogi</a><b>|</b>
<a href="">profil</a><b>|</b>
<a href="">sök</a><b>|</b>
<a href="">kontakt</a>
</div>
<div id="recension">
<h2>CSS</h2>
<p id="summering"><strong>Summering:</strong> Snyggare, snabbare, roligare</p>
<div id="infobox">
<table cellspacing="0">
<tr>
<th>Ämne:</th><td>Datalogi</td>
</tr>
<tr>
<th>Teknik:</th><td>CSS</td>
</tr>
<tr>
<th>Skola:</th><td>HIS</td>
</tr>
</table>
<ul>
<li>Högskolan i Skövde</li>
<li>0500-44 80 00</li>
<li><a href="">Info</a> | <a href="">Karta</a></li>
</ul>
</div>
```

Införande av CSS i existerande tabellbaserat HTML-dokument

```
<li>Henrik Larsson</li>
<li><a href="">Hemsida</a></li>
</ul>
</div>
```

```
<p class="paragraf1">
```

Cascading Style Sheets (CSS), är en rekommendation från W3C som definierar hur man skriver [style sheets]?, dvs dokument som anger hur olika element i en webbsida ska återges, t.ex. att rubriker ska visas i fetstil och att brödtext ska använda typsnittet Times-Roman.

```
</p>
```

```
<blockquote class="citatt">
```

Med den rätta tekniken kan du förändra ditt sätt att designa hemsidor drastiskt.

```
</blockquote>
```

```
<p>
```

En fri översättning till svenskan är kaskadkopplade stilscheman, men en riktigt bra svensk term saknas. För enskilda style sheets bör man kunna säga stilmallar, och för begreppet klarar man sig ofta med akronymen.

```
</p>
```

```
<p>
```

Alternativet till CSS är att (1) strunta i återgivningen, (2) ange återgivningen för varje textstycke eller rubrik, (3) ange återgivningen i inledningen till varje webbsida. Alla dessa alternativ har nackdelar, som man kommer åt genom att centralt för en hel webbplats ange ett eller ett fåtal olika scheman för stilen (style sheet). Dessa scheman kan kaskadkopplas, därav namnet CSS. Med det menat att man kanske har ett stylesheet för hela webbplatsen som påverkar alla dokument i denna, i ett dokument kanske man har några särskilda CSS-direktiv just för det dokumentet och sedan kanske också direktiv för en del av dokumentet.

```
</p>
```

```
<p>
```

```
<strong>Källa:</strong> Susning.nu
```

```
</p>
```

```
</div>
```

```
</body>
```

```
</html>
```

Bilaga 3 – Slutgiltig implementation

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>CSS: 10p Examensarbete Datalogi</title>
<style type="text/css">
body {font: x-small Verdana, Arial, Helvetica, sans-serif; margin: 0; padding: 0;}
table {width: 100%;}
th, td {padding: 0;}
#bildrubrik {background: white url(graypiximg.gif) 0 43px repeat-x;}
#bildrubrik img {display: block;}
#subtitel {position: absolute; top: 46px; left: 151px;}
#meny {position: absolute; top: 44px; right: 0; padding: 2px 0 2px 32px; white-space: nowrap; background: #F0DFB4
url(leftcurveimg.gif) bottom left no-repeat;}
#meny b {display: none;}
#meny a {text-decoration: none; color: #000; border-bottom: 1px solid gray; padding: 0 1em 1px 0;}
#infobox {position: absolute; top: 73px; left: 0; width: 140px; background: #F0DFB4; padding: 0.75em 0; border: 1px
solid #600; border-width: 2px 1px 2px 0;}
#infobox table {font: small Arial, Verdana, Helvetica, sans-serif;}
#infobox th {text-align: right;}
#infobox td {padding: 0.125em;}
#infobox ul {list-style: none; margin: 1em; padding: 0;}
#recension {margin: 0 80px 2em 180px;}
#recension h2 {color: #600; font-size: x-large; margin: 0.85em 0 0; line-height: 1.2em;}
#recension #summering {font-size: small; border-top: 1px dotted #600; display: inline; text-indent: 0;}
#recension p {text-indent: 2em;}
#recension .paragraf1 {text-indent: 0;}
blockquote.citat {float: right; width: 40%; padding: 1em 0 1em 5%; margin: 0; font-size: medium; font-weight: bold;}
</style>
</head>
<body>
<div id="bildrubrik">


</div>
<div id="meny">
<a href="">info</a><b>|</b>
<a href="">datalogi</a><b>|</b>
<a href="">profil</a><b>|</b>
<a href="">sök</a><b>|</b>
<a href="">kontakt</a>
</div>
<div id="infobox">
<table cellspacing="0">
<tr>
<th>Ämne:</th><td>Datalogi</td>
```


Införande av CSS i existerande tabellbaserat HTML-dokument

```
</tr>
<tr>
<th>Teknik:</th><td>CSS</td>
</tr>
<tr>
<th>Skola:</th><td>HIS</td>
</tr>
</table>
<ul>
<li>Högskolan i Skövde</li>
<li>0500-44 80 00</li>
<li><a href="">Info</a> | <a href="">Karta</a></li>
</ul>
<ul>
<li>Henrik Larsson</li>
<li><a href="">Hemsida</a></li>
</ul>
</div>
<div id="recension">
<h2>CSS</h2>
<p id="summering"><strong>Summering:</strong> Snyggare, snabbare, roligare</p>
<p class="paragraf1">
Cascading Style Sheets (CSS), är en rekommendation från W3C som definierar hur man skriver [style sheets]?, dvs dokument som anger hur olika element i en webbsida ska återges, t.ex. att rubriker ska visas i fetstil och att brödtext ska använda typsnittet Times-Roman.
</p>
<blockquote class="citatt">
Med den rätta tekniken kan du förändra ditt sätt att designa hemsidor drastiskt.
</blockquote>
<p>
En fri översättning till svenskan är kaskadkopplade stilscheman, men en riktigt bra svensk term saknas. För enskilda style sheets bör man kunna säga stilmallar, och för begreppet klarar man sig ofta med akronymen.
</p>
<p>
Alternativet till CSS är att (1) strunta i återgivningen, (2) ange återgivningen för varje textstycke eller rubrik, (3) ange återgivningen i inledningen till varje webbsida. Alla dessa alternativ har nackdelar, som man kommer åt genom att centralt för en hel webbplats ange ett eller ett fåtal olika scheman för stilen (style sheet). Dessa scheman kan kaskadkopplas, därav namnet CSS. Med det menat att man kanske har ett stylesheet för hela webbplatsen som påverkar alla dokument i denna, i ett dokument kanske man har några särskilda CSS-direktiv just för det dokumentet och sedan kanske också direktiv för en del av dokumentet.
</p>
<p>
<strong>Källa:</strong> Susning.nu
</p>
</div>
</body>
</html>
```