



Markus Knigge, Noud van Kruysbergen

Letterdressuur

Richtlijnen en tips voor een betere typografie op internet

Goede typografie is een van de belangrijkste factoren bij een goede vormgeving. Dat geldt ook voor internet, al lijkt het daar vaak een ondergeschoven kindje. Maar er zijn inmiddels weinig belemmeringen meer, technisch is typografie daar ook mogelijk. De grafische regels zijn op detailniveau dan wel wat anders dan voor bijvoorbeeld drukwerk, maar de principes zijn hetzelfde.

Ten tijde van de allereerste websites waren er nog geen bruikbare manieren om HTML-pagina's op te maken. De beginjaren werden dan ook gekenmerkt door een compromis tussen de creatieve en de technische mogelijkheden. Allerlei complexe tabellenstructuren met transparante afbeeldingen dwongen een pagina in het keurslijf van een vormgeving, waarbij zelfgemaakte afbeeldingen voor de typografische

koppen werden gebruikt. Ook vandaag de dag zijn er nog veel websites die met deze technieken zijn opgebouwd. Maar inmiddels is er wel een duidelijke scheiding mogelijk tussen inhoud en vormgeving.

Een vaak voorkomend argument tegen het gebruik van typografische regels is dat die eigenlijk bedoeld zijn voor drukwerk en daarom in die vorm niet bij een 'dynamisch' medium als het internet ge-

bruikt kunnen worden. Sommige richtlijnen zijn inderdaad duidelijk bedoeld voor het starre medium papier en zijn voor internet minder van toepassing. Uit de usability en interactiviteit van internet zijn echter ook weer nieuwe regels voortgekomen.

Kleuren

Ooit wel eens een boek of een langere tekst gelezen die op

rood papier was gedrukt? Daar is een goede reden voor. In de wereld van bladen, boeken en drukkers worden donkere letters op een lichte achtergrond als optimaal beschouwd, omdat deze het minst vermoeiend zijn om te lezen. Bij webdesign geldt hetzelfde, hoewel hier de meningen uiteen lopen. Een goed argument om bij het screendesign ook lichte letters op een donkere achtergrond te gebruiken heeft met de wer-

king van het medium te maken. Op een beeldscherm worden de kleuren met pigmenten gemengd (additieve kleurmenging); een beeldscherm geeft licht, terwijl drukwerk alleen door belichting te lezen is (subtractieve kleurmenging). Daarom vinden veel mensen het aangenamer om bij een beeldscherm lichte tekst op een donkere achtergrond te lezen.

Vergeet niet dat normale displays een resolutie hebben van 72 tot 96 dpi – en moderne laptops en smartphones zelfs tot 160 dpi. Een gedrukt boek haalt daarentegen wel 300 dpi. Los van de toegepaste typografie is lezen van een lange tekst achter een beeldscherm dus sowieso vermoeiender. Grote lappen tekst kunnen daarom beter vermeden worden. Het zinvol linken van korte artikelen werkt veel beter. Dat zal wellicht veranderen met e-bookreaders, maar de meeste webpagina's zullen voorlopig nog op een beeldscherm worden bekeken.

Een vormgever heeft voor internet variabele mogelijkheden om met kleuren en contrasten te werken. Dankzij stylesheets is het zelfs mogelijk om met een druk op de knop van kleuren en elementen te wisselen. Sinds een paar jaar is bij webdesign een trend te zien om wat te spelen met de kleurcontrasten. Vaak worden grijsnuances of

een zachte tint als achtergrond gebruikt – al dan niet met een klein kleurverloop – en zijn de letters ook niet helemaal 100 procent zwart of wit. Dit wat verminderde contrast tussen voor- en achtergrond op het scherm is subjectief prettiger om te lezen.

Tekens

Een van de belangrijkste factoren bij typografie is het kiezen van de juiste letters voor het juiste doel. Daarbij moet je denken aan middelen voor een goede leesbaarheid, die zich de afgelopen eeuwen hebben ontwikkeld. Toch wordt daar in de internetwereld uit gemakzucht vaak geen tot weinig rekening mee gehouden.

Bij markup-talen is de unicode-tekenset zo'n beetje de standaard geworden. Op internet staat die bekend als UTF-8. Met unicode is het mogelijk om niet-Latijnse tekens te gebruiken, waardoor interessante typografische effecten mogelijk zijn. Om deze tekens te kunnen gebruiken, moet je in de header van de webpagina's de volgende regel zetten:

```
<meta http-equiv="content-type"
content="text/html; charset=UTF-8"/>
```

Je moet er dan wel op letten dat de editor die je voor je HTML-broncode gebruikt met unicode

overweg kan, anders worden de moeizaam ingevoerde bijzondere tekens vervangen door kleine vierkantjes of rare tekens. Als je het niet helemaal vertrouwt, kun je de speciale tekens in de broncode ook aangeven met hun unicode-nummer (decimaal of hexadecimaal met een '0x' tussen &# en ;. Om dit te laten werken moeten de tekens dan natuurlijk ook wel in het gebruikte lettertype zitten. Voor tekens van exotische talen is dat niet altijd het geval.

Een probleem zijn aanhalingstekens. Daar zijn traditiegetrouw een aantal oplossingen voor. Ook in gangbare tekstverwerkers kun je werken met rechte en schuine aanhalingstekens, onder of boven aan de regel. Voor websites worden meestal de rechte enkele of dubbele aanhalingstekens gebruikt, maar andere landen of talen gebruiken standaard vaak schuine.

Op het toetsenbord zit een toets voor een streepje (-), en die wordt dan ook voor alle mogelijke streepjes gebruikt. Maar eigenlijk is dat alleen een verbindingstreepje voor samenstellingen of om aan het einde van een regel een woord af te breken. In de typografie heet dat een divisie; het unicode-nummer is 0x2010. Het aanzienlijk langere gedachtestreepje (-) staat meestal tussen spaties en heet in de typografie een kastlijntje; al wordt in de praktijk meestal een half kastlijntje gebruikt omdat het streepje anders te lang wordt. Ook daar is in unicode voor voorzien: het korte gedachtestreepje heet 'en dash' (–) en heeft nummer 0x2013, het lange gedachtestreepje heet 'em dash' (—) met code 0x2014. Dit is weer een ander teken dan het wiskundige minteken. De code daarvan is 0x2212 en heeft dus net als alle wiskundige symbolen een eigen nummer.

Afbreken

Het afbreken van woorden aan het einde van een zin is bij webdesign een ander verhaal dan bij drukwerk. Niemand kan honderd procent voorspellen waar een browser een tekst gaat afbreken. Er zijn daarom meerdere stuurtekens om te beïnvloeden waar de browser tekst mag afbreken en waar niet.

Speciale tekens

Teken	HTML-Entity	Unicode
Dubbele aanhalingstekens		
"	„	„
"	“	“
Enkele aanhalingstekens		
'	‘	‘
'	’	’
Vishaken		
"	»	»
"	«	«
Amperсанд		
&	&	&
Euroteken		
€	€	€
Promille		
‰	‰	‰
Paragraaf		
§	§	§
Copyright		
©	©	©
Trademark		
™	™	™
Eenvoudige machten		
²	²	²
³	³	³
Eenvoudige breuken		
½	½	½
¼	¼	¼
¾	¾	¾
Wiskunde		
–	−	−
×	×	×
√	√	√
≈	≈	≈
Pijlen		
←	←	←
↑	↑	↑
→	→	→
↓	↓	↓
↔	↔	↔
Dubbele pijl		
⇒	⇒	⇒
Kruizen		
†	†	†
‡	‡	‡
Speelkaarten		
♥	♥	♥
♠	♠	♠
♣	♣	♣
♦	&diamonds;	♦
Afbreken/tussenruimte		
Spatie zonder afbreken	 	
Kleine spatie	 	 
Kleinere spatie, niet afbreken	–	 
Koppeltken zonder afbreken	–	‑
Zacht afbreekteken	­	­
Puntjes (...)	…	…

De belangrijkste speciale tekens kun je met HTML-entiteiten aangeven. Voor andere is alleen een unicode-nummer beschikbaar.



Een visueel aansprekende vormgeving kan ook met een simpel gehouden ontwerp.

De non-breaking-space (unicode 0xc2a0, in HTML) wordt vaak gebruikt, maar niet altijd voor het juiste doel. Vaak wordt hij toegepast om de afstand tussen woorden of tekens kunstmatig te vergroten bij bijvoorbeeld getalaanduidingen (Hoofdstuk 15). Voor koppelteken is er iets vergelijkbaars met de non-breaking-hyphen (0x2011, ‑), bijvoorbeeld bij 'e-book'.

Maar wat als een lang woord de lay-out lijkt te gaan ondermijnen? Dan is het mogelijk om binnen dat woord mogelijke afbreekplekken aan te geven met zachte afbreektekens (soft-hyphen, ­), bijvoorbeeld bij gitaar­versterker­simulatie­programma. Sommige oude browsers negeren die stuurtekens, maar geven ook geen foutmeldingen.

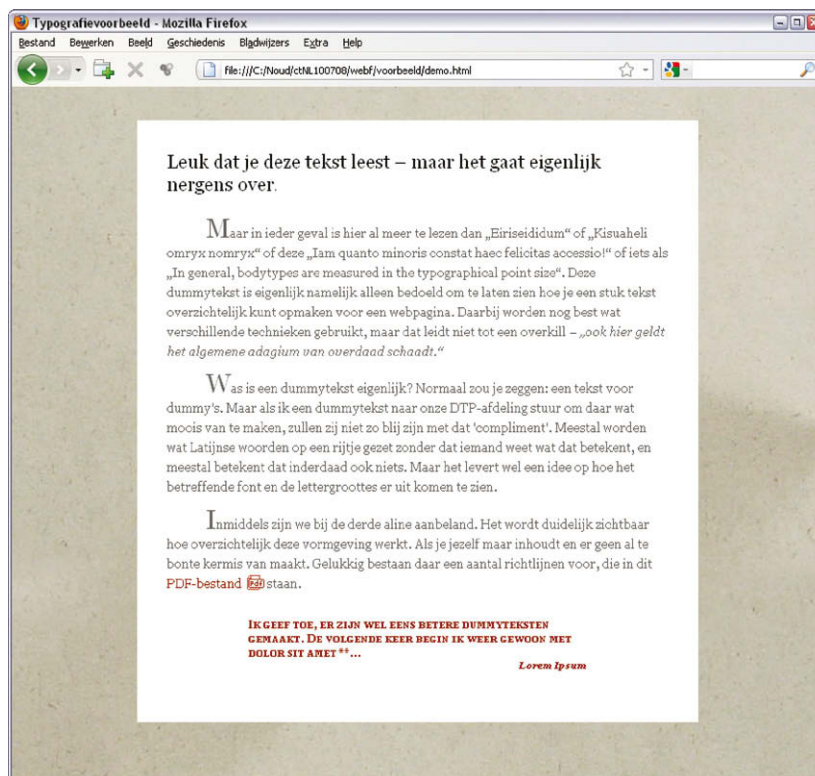
Afstand houden

Bij typografie is de juiste afstand tussen tekens essentieel. Afstanden tussen woorden zijn een kleiner probleem. Die worden alleen in uitzonderingsgevallen aangepast. In bepaalde gevallen is het echter toegestaan om afstanden tussen letters te verkleinen. Een goede typografie uit zich in de juiste afstand bij bijvoorbeeld de lay-out van getalblokken of afkortingen die optisch een wat grotere ruimte tussen de letters krijgen. Zo staat R.U.N. wat dicht op elkaar, maar R. U. N. weer te ver. Dat wordt opgelost door een halve spatiëring te gebruiken: R. U. N.

Dit soort afstanden zijn ook via CSS te realiseren, maar het kan makkelijker met het daarvoor bedoelde speciale teken  . En als er op die plek niet afgebroken mag worden, wat de voorkeur heeft bij een afkorting, dan kun je   gebruiken (unicode 0x202f, narrow no-break space). De vuistregel: tot en met vier cijfers doe je zonder spatiëring, bijvoorbeeld postcodes, telefoonnummers groepeer je in getallen van 2 of 3 cijfers, bankrekeningnummers in groepen van 3. Grote getallen krijgen een punt als scheidingsteken (1.000.000) en kleine cijfers worden uitgeschreven.

Dan is er nog het beletseltekens (...). Die drie puntjes worden ook niet met een standaard spatiëring weergegeven, maar

Beginkapitalen, harmonische afstanden, symbolen voor links en andere details zorgen voor een fraai en rustig beeld.



dichter bij elkaar. Voor dat teken gebruik je ….

Als je een duidelijk accent wilt leggen en een deel van een regel MET HOOFDLETTERS WILT WEERGEVEN, dan is het bij veel fonts aan te raden om de tekenafstand (kerning) wat te verhogen en het lettertype te verkleinen. Veel fonts hebben speciaal voor dat doel een extra tekenset met kleinkapitalen of 'small-caps', die deze aanpassingen standaard al hebben. De CSS-eigenschap font-variant: small-caps doet dat min of meer automatisch.

Ook de regelafstand is een belangrijk stijlmiddel. Ten tijde van de oude drukpersen werd die afstand bepaald door de hoogte van de metalen letters. Een grotere interlinie werd gecreëerd door vulmateriaal tussen de regels te gebruiken. Voor een website is het veel eenvoudiger om de afstanden te regelen, omdat dat gewoon via CSS kan. De minimale afstand wordt normaal op 120 procent van de letterhoogte ingesteld, dus een regelafstand van 12 punten bij een lettergrootte van 10 punten. Doordat er vaak met relatief korte teksten wordt gewerkt, wordt deze afstand vaak verhoogd, maar zelden tot boven de 150 procent. Een grotere regelafstand ziet er wat chi-

quer uit, maar bij een te grote regelafstand verliest een tekstblok zijn samenhang.

Alinea's krijgen een extra witregel en kunnen inspringen. Als je een hele alinea laat inspringen kun je een mooi accent leggen op een citaat of een belangrijke conclusie. Als je het advies volgt om met relatieve groottes in procenten of met 'em' te werken, kun je makkelijk een uitgebalanceerde tekst maken voor alle verschillende browserverstergroottes. Voor de regellengte geldt het typografische principe om niet meer dan 70 tekens op een regel te zetten. Met CSS3 zal het uiteindelijk zonder trucs mogelijk zijn om meerkolomstext in een browser te laten zien, waardoor de breedte van monitoren nog beter te benutten is.

Accenten

Cursieve letters zijn op een beeldscherm eigenlijk niet done omdat ze moeilijker te lezen zijn. Eigenlijk zijn ze alleen geschikt om korte passages te accentueren. In een lopende tekst moet je ook vette en onderstreepte woorden zoveel mogelijk vermijden. Onderstrepen is een relikwie uit de tijd van oude schrijfmachines en wordt in de webwereld boven-

dien gebruikt voor het vormgeven van hyperlinks. Voetnoten kun je met de CSS-eigenschap font-variant optisch benadrukken, bijvoorbeeld als kleinkapitalen. Met text-transform: capitalize kun je instellen dat ieder woord met een hoofdletter moet beginnen. Een andere mooie manier is het via CSS automatisch toevoegen van kleine afbeeldingen aan een link, in dit geval bij een PDF-bestand:

```
a.pdf { background: url(pdf1.gif) right
no-repeat; }
a.pdf:visited { background: url(pdf2.gif)
right no-repeat; }
```

Het consequent gebruik van de juiste tekens op de juiste plek verhoogt niet alleen de leesbaarheid, maar geeft je website ook een personal touch. Het gebruik van inspringingen, hoofdletters en andere typografische technieken draagt bij aan de stijl. Met CSS kun je deze opmaakelementen flexibel en makkelijk aansturen, zie de voorbeeldcode bij de softlink.

Door groottes in procenten aan te geven, werk je met een relatieve lay-out. Daardoor loopt de tekst nooit hetzelfde en kun je niet tot op de millimeter nauwkeurig aangeven waar wat moet komen, zoals bij drukwerk wel kan. Maar dat is inhe-

SUPERMICRO[®]

It's a New Day

Intel[®] Xeon[®] Processor 5500 Series (Tylersburg/Nehalem) Based Solutions are Here.

375 GFLOPS/kW, Highest Performance per Watt per Dollar

Most advanced video processing with up to 4x **GPU-ready PCI-E Gen. 2.0 x16 slots**

Greatest I/O Bandwidth **Dual IOH** for **72 PCI-E Gen 2 Lanes**

Widest variety of I/O options with **UIO** (Universal I/O) technology

Most advanced storage options with onboard **SAS 2** (6 Gbps)



SuperBlade[®]
14 Blade



2U Twin²
4 Nodes in 2U



2U SuperServer
6026T-3RF



4U Workstation
(Up To 4 GPU Supported)

Most flexible networking with onBoard **IPMI** / dedicated third LAN

Most efficient Gold Level (**93%+**) power supplies

Quietest operation as low as **28dB**

Supermicro Computer BV
Tel : +31-73-6400390
E-mail : Sales @supermicro.nl
www.supermicro.nl

SMBE Belgium
+32-495 533 245
www.smbe.be
sales@smbe.be

TWP Computers
+31-20-6383057
www.twp.nl
info@twp.nl

NCS International
+31-544-470000
www.ncs.nl
info@ncs.nl

Triple S Belgium
+32-9-261 5310
www.s3s.be
sales@s3s.be



**Powerful.
Intelligent.**

rent aan het medium. In ieder geval staan je bestandsdelen in de browser altijd in de goede verhouding bij elkaar.

Inspringen van alinea's doe je bij CSS met `text-indent`. Met `p:first-letter {font-size: 2em;}` leg je de nadruk op de eerste letter.

Maar in principe geldt: overdaad schaadt. Als je overdrijft, vallen de accenten niet meer op en wordt de tekst onrustig en rommelig, wat een onprofessionele indruk achterlaat.

Lettertypes

De keuze van het juiste lettertype is natuurlijk een van de belangrijkste lay-outaspecten. De basisregel daarbij is dat je twee of hooguit drie verschillende lettertypes gebruikt.

Meestal wordt het verschil tussen koppen en de lopende tekst aangegeven door verschillende fonts. Voor de lopende tekst moet je een font kiezen dat een optimale leesbaarheid voor het betreffende medium garandeert. Bij drukwerk wordt vaak gekozen voor lettertypes met schreven (Eng: Serif). Deze uitsteeksels aan letters, zoals in het font Times New Roman, helpen om het regelbeeld te verbeteren en om je ogen te begeleiden naar de volgende regel. Op internet worden de schreefloze Antiqua-fonts het meest gebruikt, omdat de schreven door de relatief lage resolutie van een display tot een veel te onrustig beeld bij kleinere lettergroottes leiden. Deze regel verbiedt natuurlijk niet dat je niet met het uitgekozen font mag spelen. Van ieder lettertype bestaan normaal minimaal cursieve en vette varianten. Professionele fonts worden vaak in meerdere

varianten aangeboden, bijvoorbeeld met verschillende dikte en breedte of met of zonder schreef. Vooral bij koppen heb je wat meer ruimte om wat te spelen met de lay-out.

In 1996 publiceerde Microsoft het door Matthew Carter gemaakt lettertype Verdana, dat geoptimaliseerd is voor de leesbaarheid van kleine letters op een beeldscherm. Dit lettertype wordt door de systeem-onafhankelijke beschikbaarheid dan ook heel vaak gebruikt voor de lopende tekst bij websites. Je kunt je dan ook van de massa onderscheiden door net een ander font dan Verdana te gebruiken.

Aparte fonts

Er zijn maar een handvol fonts zo wijd verspreid dat je die voor internet kunt gebruiken zonder daar bij na te hoeven denken. Daarom worden fonts vandaag de dag vaak als afbeeldingen in een website ingebouwd. Dat is wel handig wat de browser-onafhankelijke compatibiliteit betreft, maar qua toegankelijkheid, zoekmachineoptimalisatie (SEO) en beheer is het een ramp.

Na veel gesteggel komt er in CSS3 eindelijk een oplossing voor dit probleem met de `@font-face`-declaratie. Alle moderne browsers kunnen hiermee overweg [1]. Het gebruik van commerciële fonts met CSS3 leidt echter tot auteursrechtelijke problemen. Bovendien zijn er wat kleine incompatibiliteiten die het gebruik lastig maken en sommige oudere browsers moeten simpelweg passen. Daarom hebben alternatieve technieken als sIFR en Cufón voorlopig nog bestaansrecht.

sIFR

Scalable Inman Flash Replace (afgekort sIFR) is een populaire opensourcetoel om via standaard webtechnieken en Flash toch andere fonts in een website te kunnen gebruiken (zie softlink). sIFR is licentievrij en werkt door de populariteit van Flash systeem- en platformonafhankelijk. De gebruikte fonts worden samen met een van tevoren gemaakt FLA-bestand met Flash Professional in een gecompileerde SWF-container gezet, wat licentieperikelen minimaliseert. Als je geen Flash Professional hebt, kun je ook uitwijken naar verschillende gratis oplossingen, zoals OpensIFR. Een webpagina laadt eerst de JavaScript- en CSS-componenten van sIFR:

```
<script src="sifr/sifr.js"
  type="text/javascript"></script>
<link href="css/sifr_screen.css"
  rel="stylesheet" type="text/css"
  media="screen"/>
```

In de body van de HTML verschijnen de te formatteren delen als normale tekst, wat gunstig is voor de toegankelijkheid en zoekmachineoptimalisatie. Als JavaScript en Flash in de browser geactiveerd zijn, wordt de tekst aan de SWF doorgegeven, daar met de ingesloten tekens gerenderd en op de plek van de oorspronkelijke tekst weer ingevoegd. De volgende aanroep vervangt alle teksten met het attribuut `class="headlines"`:

```
<script type="text/javascript">
if(typeof sIFR == "function") {
  sIFR.replaceElement(".headlines",
    named({FlashSrc:".sifr/script_1.swf",
      sColor: "#FFFFFF",
      sWmode: "transparent"}));
}
</script>
```

Met verschillende variabelen kun je een aantal details regelen, zoals hier een transparante achtergrond. De tekst is te selecteren en kopiëren. De grootte van de SWF-video met de tekst kan wel een probleem worden, maar daar hebben alle webfonttechnieken mee te maken. Er is veel ruimte te winnen door in het SWF-bestand alleen de gebruikte tekens op te nemen.

Cufón

Cufón is een nog relatief nieuwe, licentievrije techniek die een "worthy alternative" voor sIFR wil zijn (zie softlink). Het principe is vergelijkbaar, alleen werkt het zonder plug-in. Om de fontfiles te maken, moet je de fonts in het formaat TTF of OTF via het webfront-end <http://cufon.shoqolate.com/> genereren en uploaden. Vanuit licentie-oogpunt zit dat toch niet helemaal lekker. De makers van Cufón wijzen er nadrukkelijk op dat degene die de fonts uploadt, daar ook de rechten voor moet hebben. Er zijn ook nog eens verschillen tussen de verschillende fontproducenten.

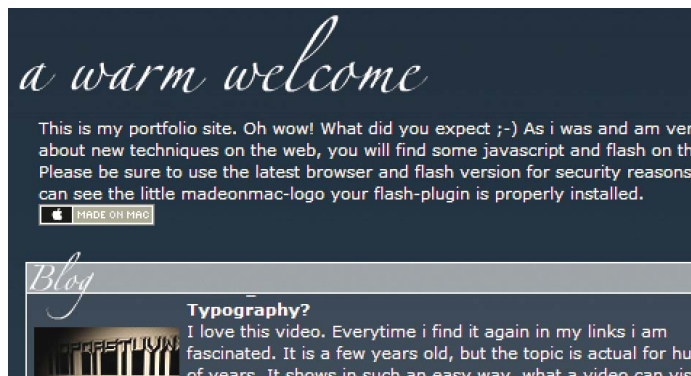
Het is mogelijk het weergeven van de webfonts te beperken tot bepaalde domeinen. Cufón converteert de geüploade fonts met de software FontForge op de server naar vectorgraphics, voegt daar wat stuurinformatie aan toe en zet het resultaat in een JavaScript-bestand. De HTML-pagina's includen de JavaScript voor het font en de rendering en geven aan welke tekst vervangen moet worden. In dit voorbeeld gebeurt dat met alle `<h1>`-koppen:

```
<script src="cufon-yui.js"
  type="text/javascript"></script>
<script src="mijnfont.js"
  type="text/javascript"></script>
<script type="text/javascript">
  Cufon.replace('h1');</script>
```

Internet Explorer heeft nog een extra zetje nodig, net voor de afsluitende `</body>`-tag, om een vertraging in de opbouw te voorkomen:

```
<script type="text/javascript">
  Cufon.now();</script>
```

Cufón werkt goed omdat het gebruik maakt van browserspecifieke kenmerken. De ontwikkelaars hebben daar een aardige spagaat voor moeten maken: FontForge converteert de fonts eerst naar SVG, dan naar VML en geeft dat dan door aan Internet Explorer, die dat vectorformaat als enige ondersteunt. Voor alle andere gangbare browsers converteert Cufón de SVG-code naar HTML5-canvas, omdat er bij SVG problemen zijn met de compatibiliteit en de snelheid. Door de browserspecifieke



Om mooie letters op een site te krijgen, hoeft je niet meer terug te vallen op afbeeldingen.

functies kan er weinig verkeerd gaan. De performance is dan ook opmerkelijk.

Beide technieken laten zien dat de wens om makkelijk eigen fonts bij webdesign te kunnen gebruiken de ontwikkelaars aanspoort tot fascinerende oplossingen. Als een website sowieso Flash nodig heeft, is er niets tegen sIFR. Cufón heeft geen plug-in nodig, maar heeft wat fontlicenties betreft nog een heikel punt. Je kunt er ook geen PostScript-fonts mee gebruiken en in tegenstelling tot sIFR kun je teksten ook niet kopiëren en plakken.

Voor koppen en korte teksten werken deze oplossingen prima – maar eigenlijk zijn het hacks.

Fonts met CSS3

In CSS 2.0 uit 1997 zit al een zui-vere oplossing om eigen fonts via CSS te embedden:

```
@font-face {
  font-family: MyHelvetica;
  src: local("Helvetica New"),
       local("HelveticaNew"),
       url(MyHelvDownload.ttf);
}
```

Deze declaratie definieert de fontfamilie MyHelvetica. De beide local-opdrachten controleren of een font met de naam Helvetica New of HelveticaNew op de client geïnstalleerd is. Als dat niet het geval is, wordt een TTF-bestand gedownload en als basis voor de definitie gebruikt. Dat werk in Safari (vanaf 3.1), Firefox (vanaf 3.5), Opera (vanaf 10.0) en Chrome (vanaf 4.0) met TrueType- en OpenType-fonts. Bij versie 3.5 van Firefox komt daar nog het nieuwe Web Open Font Format (WOFF) bij.

Internet Explorer kent deze truc al sinds versie 4.0, maar dan alleen met Embedded OpenType (EOT) fonts. Maar met weinig moeite is gelukkig een browseronafhankelijke declaratie te maken:

```
@font-face {
  font-family: MyHelvetica;
  src: url(MyHelvDownload.eot),
       url(MyHelvDownload.otf);
}
```

Internet Explorer kijkt naar het EOT-bestand, alle anderen gebruiken OpenType. Hierna is het

lettertype gewoon te gebruiken. Om compatibel te zijn met oudere browsers moet je wel nog fonts declareren die zoveel mogelijk lijken op de fonts die je wilt gebruiken en op praktisch iedere computer te vinden zijn:

```
#text1 {
  font-family: MyHelvetica, Helvetica,
               Arial, Verdana;
}
```

Hoe mooi deze techniek ook is, in de praktijk zijn er toch nog wel wat problemen. Zo zijn de meeste fontbestanden gewoon te groot om in zijn geheel als webfont te integreren: meestal zitten er namelijk meerdere varianten en een groot aantal speciale tekens in zo'n bestand. Het liefst zou je alleen die varianten willen die ook echt gebruikt worden, maar daarvoor moet je het fontbestand uit elkaar gaan halen. Dat kan met opensource-programma's als FontForge of met een online font-optimizer (zie softlink), maar kost wel wat tijd.

Dit probleem treedt bij sIFR en Cufón natuurlijk ook op, maar de vraag naar het formaat is een specifiek CSS3-probleem. Omdat Internet Explorer alleen het EOT-formaat ondersteunt, moeten de fonts niet alleen uitgedund worden, maar ook geconverteerd. Microsoft heeft daar de gratis tool WEFT voor (zie softlink). Voor andere besturingssystemen wordt het wat lastiger om software te vinden waarmee dat kan, maar onmogelijk is het niet (zie softlink).

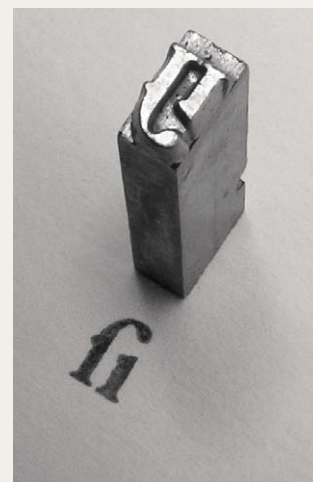
Het grootste probleem zijn echter de licenties. De meeste grote aanbieders staan gebruik van hun fonts op deze manier niet toe of willen daar een extra licentievergoeding voor hebben. Een font dat via CSS op een website wordt gebruikt, is dan namelijk voor iedereen toegankelijk. Je moet als webdesigner of exploitant dus zeker weten dat je de juiste licenties hebt om de auteursrechten niet te schenden. Dat kan namelijk een duur grapje worden. Je kunt ook licentievrije fonts gebruiken, al zijn die voor professionele vormgevers vaak niet goed genoeg.

Als laatste alternatief voor eigen fonts op een website zijn dan nog online aanbieders als typekit.com te noemen. Die

Ligaturen

Ligatuur komt van het Latijnse *ligatura*, verbinding. Het zijn speciale tekens die ten tijde van de boekdrukkunst zijn ontstaan om minder mooie gaten tussen twee letters met overhang te vermijden, zoals bij de f, i, t en l. Er werden speciale lettercombinaties gegoten die uit de twee betreffende letters bestonden. Bij kranten werd daar meestal geen gebruik van gemaakt, maar in de professionele boekdrukkunst worden ligaturen vaak gebruikt.

Bij opmaken via de computer past de software de afstand tussen de letters automatisch aan, maar dat leidt niet helemaal tot hetzelfde resultaat als met ligaturen. Op internet kun je ook ligaturen gebruiken, maar dan gaan wel de toegankelijkheid en de indexeerbaarheid door zoekmachines verloren. Bij moderne OpenType-fonts wordt de afstand tussen twee letters, de kerning, automatisch geregeld en lijken de resultaten aardig op echte ligaturen. Bij gotische letters



Ligaturen verfriaaien het letterbeeld, maar zijn bij webdesign moeilijk toe te passen.

zijn ligaturen echter onvermijdbaar.

Op een scherm ziet een finale floret er anders uit dan wanneer dat met ligaturen wordt gedaan: finale floret. In HTML wordt dat dan `ﬁnale` `ﬂoret`.

hebben van de nood een deugd gemaakt en een businessmodel opgesteld waarbij commerciële fonts geprepareerd worden voor gebruik op een eigen website. Je betaalt vervolgens afhankelijk van het maandelijks dataverkeer en het aantal websites en fonts een bedrag dat ergens tussen de 25 en 250 dollar per jaar ligt.

Stijlen in de toekomst

Met de invoering van CSS3 is in browsers typografisch meer mogelijk dan alleen het gebruik van webfonts. CSS3 is formeel nog niet definitief, maar veel features van de specificatie zitten nu al in de diverse browsers. Je moet per situatie bekijken of je daar nu al gebruik van wilt maken. Je moet er dan namelijk op rekenen dat je meer tijd kwijt bent met testen, zeker ook omdat de details van de implementatie nog kunnen veranderen.

Met de eigenschap `test-wrap/word-wrap` kun je redelijk precies instellen hoe de browser teksten afbreekt. De waarde `unre-`

stricted staat de browser zelfs toe om op lettergreepniveau af te breken. Bij de waarde `suppress` moet er daarentegen naar zinnvolle afbreekpunten als interpuncties worden gezocht. Als die niet worden gevonden, mag er volgens de standaardmethoden worden afgebroken. Met de `word-wrap`-waarde `break-word` mag een browser een woord zelfs op ongebruikelijke plekken afbreken als er verder geen acceptabele mogelijkheid wordt gevonden.

Een aantal andere interessante declaraties staan nog in de kinderschoenen. Zo moeten bijvoorbeeld decoraties als tekstoutlines en tekstschaduw in de toekomst mogelijk zijn om webdesigners meer mogelijkheden te geven dan waar Tim Berners-Lee in het begin van de jaren 90 van had durven dromen. (nkr)

Literatuur

[1] Herbert Braun, *Stijlvast*, Heeft CSS3 nu al iets te bieden voor webdesigners?, c't 5/2010, 116