

DIE WICHTIGEN ÄNDERUNGEN UND ERNEUERUNGEN

CASCADING

STYLESHEETS LEVEL 3

CSS 3

GERÄTE UNTERSCHIEDEN

MEDIA ATTRIBUTES

MEDIEN ATTRIBUTE EINSETZEN

```
@media screen and (max-width:480px) { ... }  
@media screen and (min-width:481px) { ... }  
@media screen and (min-width:768px) { ... }  
@media screen and (min-width:1024px) { ... }  
@media screen and (min-width:1280px) { ... }  
@media screen and (min-width:1600px) { ... }  
@media screen and (min-width:1920px) { ... }
```

MOBILE FIRST

PROGRESSIVE ENHANCEMENT

ALL DEVICES

```
@media screen {  
    declarations for all devices,  
    optimized for xtra small mobile devices  
}
```

XTRA SMALL

```
@media screen and (max-width:480px) {  
    changes/specials for extra small  
}  
@media screen and (min-width:481px) { ... }
```

| | | |
|------------|--------------------------------------|---------|
| SMALL | @media screen and (min-width:768px) | { ... } |
| MEDIUM | @media screen and (min-width:1024px) | { ... } |
| LARGE | @media screen and (min-width:1280px) | { ... } |
| XTRA LARGE | @media screen and (min-width:1600px) | { ... } |
| HD | @media screen and (min-width:1920px) | { ... } |

DESKTOP FIRST

GRACEFUL DEGRATION

ALL DEVICES

```
@media screen {  
    declarations for all devices,  
    optimized for medium devices  
}
```

MEDIUM

```
@media screen and (min-width:1024px) and (max-width:1279px) {  
    changes/specials for extra small  
}
```

| | | |
|------------|--------------------------------------|---------|
| SMALL | @media screen and (max-width:767px) | { ... } |
| XTRA SMALL | @media screen and (max-width:479px) | { ... } |
| LARGE | @media screen and (min-width:1280px) | { ... } |
| XTRA LARGE | @media screen and (min-width:1600px) | { ... } |
| HD | @media screen and (min-width:1920px) | { ... } |

ANDERE ATTRIBUTE

```
@media screen and (orientation:portrait) { ... }
```

```
@media screen and (resolution:72dpi) { ... }
```

```
@media screen and (aspect-ratio:16/9) { ... }
```

SELEKTOREN- ERWEITERUNGEN

JAVASCRIPT GRUNDLAGEN

NEUE SELEKTOREN SEIT JS 1.7

Bisher: Elemente nach id ermitteln (DOM API)

```
var el = document.getElementById('section1');  
el.focus();
```

```
var els = document.getElementsByTagName('div');  
els[0].focus();
```

```
var els = document.getElementsByClassName('page-header');  
els[0].focus();
```


JAVASCRIPT GRUNDLAGEN

NEUE SELEKTOREN SEIT JS 1.8

Elemente mit CSS Syntax ermitteln (Selectors API)

```
var el =  
    document.querySelector("form#action");  
  
var els =  
    document.querySelectorAll("table.test > tr > td");
```

JAVASCRIPT: ELEMENT.CLASSLIST

```
<div id="main" class="shadow rounded"></div>

var obj = document.querySelector('#main').classList;

obj.add('highlight');
obj.remove('shadow');
obj.toggle('highlight');

console.log(el.contains('highlight')); // false
console.log(el.contains('shadow')); // false
console.log(el.classList.toString() === el.className); // true

Output:
<div id="main" class="rounded"></div>
```

LEVEL 3 SELEKTOREN

CSS SELEKTOREN CHEAT SHEET

| | |
|---|--------------------------------|
| <code>body * {}</code> | → General Selektor |
| <code>body {}</code> | → Elementeselektor |
| <code>.my-class {}</code> | → Klassenselektor |
| <code>#my-id {}</code> | → ID-Selektor |
| <code>a[href]</code> | → Attributselektor (minified) |
| <code>input[type=email]</code> | → Attributselektor |
| | |
| <code>body [class*=page-]</code> | → Descendantselektor |
| <code>nav > ul > li</code> | → Childselektor |
| <code>[class=page-header] ~ [class*=page-]</code> | → General Following Siblings |
| <code>[class*=column] + [class*=column]</code> | → Immediate Following Sibling |
| | |
| <code>a[href]:hover</code> | → Stateselektor (Pseudoklasse) |
| <code>[class*=page-]::before {}</code> | → Pseudoelement |
| <code>nav li:last-child {}</code> | → Pseudoklasse |

NTH-CHILD

```
h2:first-child { ... }  
h2:last-child { ... }  
  
.row:nth-child(even) { /*Gerade Zeilen */  
    background: #dde;  
}  
.row:nth-child(odd) { /* Ungeraden Zeilen */  
    background: white;  
}  
.row:nth-child(5n) { /* Jede fünfte Zeile */  
    background: #dde;  
}  
.row:nth-child(5n+1) { /* Jede fünfte Zeile */  
    background: #dde;  
}
```

NTH-OF-TYPE()

```
div#test p:nth-of-type(4) { ... }
```

```
div#test p:first-of-type { ... }
```

```
div#test p:last-of-type { ... }
```

```
div#test p:only-of-type { ... }
```

NTH-CHILD() VS. NTH-OF-TYPE

```
<style>
footer a:nth-of-type(odd) { color: green; }
footer a:nth-child(odd) { color : lightblue; }
</style>
```

```
<footer>
  <a href="">link</a>
  <span>Span</span>
  <a href="">link</a>
  <a href="">link</a>
  <span>Span</span>
  <span>Span</span>
  <a href="">link</a>
  <a href="">link</a>
  <a href="">link</a>
  <a href="">link</a>
</footer>
```

```
<footer>
  <a href="">link</a>
  <span>Span</span>
  <a href="">link</a>
  <a href="">link</a>
  <span>Span</span>
  <span>Span</span>
  <a href="">link</a>
  <a href="">link</a>
  <a href="">link</a>
  <a href="">link</a>
</footer>
```

VERNEINUNG VON ATTRIBUT SELEKTIEREN

```
input[type=text] {  
    background: #eee;  
}  
input[type!=text] {  
  
a[href] { ... }  
a[!href]
```


NEGATION VON SELEKTOREN

```
:not(.box) {  
  color: #00c;  
}  
:not(span) {  
  display: block;  
}
```

```
h2:not(.teaser)  
statt h2.default, h2.special  
<h2 class="teaser"></h2> -> wird nicht ausgewählt!  
<h2 class="default"></h2>  
<h2 class="special"></h2>
```

:: FÜR PSEUDOELEMENTE

```
span[data-descr]:hover::after {  
    content : attr(data-descr);  
}
```

TOOLTIPP PER PSEUDOELEMENT (NEUE SCHREIBWEISE)

```
span[data-descr] {  
    position : relative;  
    color    : #00F;  
    cursor   : help;  
    text-decoration : underline;  
}  
span[data-descr]:hover::after {  
    content      : attr(data-descr);  
    position     : absolute;  
    left        : 0;  
    top         : 24px;  
    min-width   : 200px;  
    background-color : #ffffcc;  
    padding     : 12px;  
    color       : #000000;  
    font-size   : 14px;  
    z-index     : 1;  
    border      : 1px #aaaaaa solid;  
    border-radius : 10px;  
}
```

DISPLAY - APPEARANCE - TEXT-OVERFLOW - BOX-SIZING

DIE ANZEIGE VON ELEMENTEN STEUERN

INLINE-BLOCK

```
div {  
  display: inline-block;  
}
```

Stattet das Inlineelement mit width/height aus,
auf ein float Attribut kann hier teilweise verzichtet werden.

```
div {  
  display: inline-block;  
  width : 12em;  
  height :12em;  
  margin : 1em;  
  padding : 2em;  
}
```

ELEMENT MIT TABELLENVERHALTEN

```
div {  
  display: table;  
}  
div {  
  display: table-cell;  
}
```

Ein Blockelement erhält eine Höhe und verdrängt damit Elemente, auch wenn nichts drin steht.
Lässt eine vertikale Zentrierung zu!

MICRO CLEARFIX VON GALLAGHER

```
.cf:before, .cf:after {  
    content: " ";  
    display: table; /* 2 */  
}  
.cf:after {  
    clear: both;  
}  
.cf {  
    *zoom: 1; /* for IE 6/7 only */  
}  
  
/* seit CSS3: element::after { ... } */
```

BOX MODEL ADDITION

`box-sizing: content-box`

Dimensionsverhalten wie in CSS 2 – Höhe, Breite werden definiert, padding, border, margin und outline werden dazu gerechnet.

`box-sizing: border-box`

In die Boxbreiten- und Höhenangabe werden padding und border eingerechnet.

```
div.row-fluid {  
    box-sizing : border-box;  
}
```

`box-sizing: padding-box`

Padding mit in die width Dimension

APPEARANCE

Mit appearance kann man beliebige HTML-Elemente so aussehen lassen, dass sie wie die diversen Bedienelemente (Buttons, Textfelder) des Betriebssystems/Browsers aussehen. Um einen Link wie eine Checkbox aussehen zu lassen, macht man etwa folgendes:

```
appearance:
normal|icon|window|button|menu|field;
```

```
section a {
    appearance:checkbox;
    -webkit-appearance:checkbox;
    -moz-appearance:checkbox;
    text-indent: 1.2em;
}
```

WEBKIT VALUES

checkbox
radio
push-button
square-button
button
button-bevel
listbox
listitem
menulist
menulist-button
menulist-text
menulist-textfield
scrollbarbutton-up
scrollbarbutton-down
scrollbarbutton-left
scrollbarbutton-right
scrollbartrack-horizontal
scrollbartrack-vertical

scrollbarthumb-horizontal
scrollbarthumb-vertical
scrollbargripper-horizontal
scrollbargripper-vertical
slider-horizontal
slider-vertical
sliderthumb-horizontal
sliderthumb-vertical
caret
searchfield
searchfield-decoration
searchfield-results-decoration
searchfield-results-button
searchfield-cancel-button
textfield
textarea

MOZILLA VALUES

| | |
|---------------------------|---------------------------------|
| none | separator |
| button | statusbar |
| checkbox | tab |
| checkbox-container | tab-left-edge Obsolete |
| checkbox-small | tabpanel |
| dialog | textfield |
| listbox | textfield-multiline |
| menuitem | toolbar |
| menulist | toolbarbutton |
| menulist-button | toolbox |
| menulist-textfield | -moz-mac-unified-toolbar |
| menupopup | -moz-win-borderless-glass |
| progressbar | -moz-win-browser-tabbar-toolbox |
| radio | -moz-win-communications-toolbox |
| radio-container | -moz-win-glass |
| radio-small | -moz-win-media-toolbox |
| resizer | tooltip |
| scrollbar | treeheadercell |
| scrollbarbutton-down | treeheadersortarrow |
| scrollbarbutton-left | treeitem |
| scrollbarbutton-right | treetwisty |
| scrollbarbutton-up | treetwistyopen |
| scrollbartrack-horizontal | treeview |
| scrollbartrack-vertical | window |

TEXTE ABSCHNEIDEN

- ```
div {
 text-overflow: ellipsis;
}
```

- 

TITEL EINER SE...

- Leider nur bei einzeiligen Texten!

# ANIMATIONEN UND ÜBERGÄNGE

# CSS TRANSITION ANIMATIONEN UND EFFEKTE

- Eine CSS Transition ändert den Wert einer CSS-Eigenschaft über eine festgelegte Zeit.
- Dabei entstehen Animationen wie das Ein- und Ausblenden und Bewegen von HTML-Elementen ohne Javascript oder Flash.

# AUFBAU EINER TRANSITION

```
#box.left { margin-left: 0; }
#box.right { margin-left: 1000px; }

#box {
 transition : margin-left 1s ease-in-out;
}

document.getElementById('box').className = 'left';
document.getElementById('box').className = 'right';
```

# TRANSITION PROPERTY, DURATION, TIMING

## **transition-property**

Die Eigenschaft, die während des Übergangs geändert wird.

## **transition-duration**

Die Zeit, in der ein Übergang stattfinden soll.

## **transition-delay**

Zeit bis zum Start des Übergangs.

## **transition-timing-function**

Zeitlicher Verlauf des Übergangs.



# EIN AUFKLAPPENDES MENU

```
.content-nav {
 display : block;
 height : 0;
 overflow : hidden;

 transition-property : height;
 transition-duration : 0.25s;
 transition-timing-function : ease-in-out;
}

.menu-show {
 display : block;
 height : 5em;
}

$('button#menu-toggle').on('click', function () {
 $('.content-nav').toggleClass('menu-show');
});
```

# ANIMATIONEN

```
@-keyframes my-animation {
 0% { background-color:black; }
 25% { background-color:red; }
 50% { background-color:green; }
 75% { background-color:yellow; }
 100% { background-color:black; }
}

.background {
 animation: my-animation 4s infinite;
}
```

# TRANSFORMATIONEN

```
transform : rotateY(45deg);
transform : scaleX(25deg);
transform : translate3d(0, 0, 90deg);
transform : perspective(500px)

#threed-example {
 transform : rotateZ(5deg);
 transition : transform 2s ease-in-out;
}

#threed-example:hover {
 transform : rotateZ(-5deg);
}
```

# ANIMIEREN VON STATE-KLASSEN

Eine CSS3 Transition wird immer dem Ausgangszustand eines Elementes zugewiesen. So können alle Zustände des Elements ebenfalls animiert werden:

- :hover
- :target
- :focus
- :valid
- :checked
- :active
- :disabled
- :invalid

# ÜBERGÄNGE MIT STATES

```
<div class="button">Mit Transition</div>
```

```
...
```

```
.button a {
 background-color: #9FC0D0;
 transition-property: background-color;
 transition-duration: 1s; }

.button a:hover { background-color: #759DB2; }
```

# DIE EASING EFFEKTE

## **ease**

Ohne weitere Angaben starten Transitions mit CSS langsam, beschleunigen in der Mitte und werden am Ende wieder langsamer.

## **linear**

ist eine gleichförmiger Übergang,

## **ease-in**

beginnt langsam und wird schneller,

## **ease-out**

beginnt schnell und läuft langsamer aus

## **ease-in-out**

kombiniert das langsame Anlaufen mit einem langsamen Auslaufen, aber einem deutlich schnelleren Teil in der Mitte.

## **cubic-bezier(x1,y1,x2,y2)**

ist Fine-Tuning für den Verlauf der Animation

# SCHRIFTEN

# MEHRSPALTENSATZ



# MEHRSPALTENSATZ - DAS MULTICOLUMN MODEL

```
p { column-count: 3 }
p { column-width: 12em }
p { columns: 3 12em }

p {
 column-gap: 1em;
 column-rule-width: 1px;
 column-rule-style: solid;
 column-rule-color: black;
}
```

In March 1936, an unusual confluence of forces occurred in Santa Clara County.

A long cold winter delayed the blossoming of the millions of cherry, apricot, peach, and prune plum trees covering hundreds of square miles of the Valley floor. Then, unlike many years, the rains that followed were light and too early to knock the blossoms from

overnight, the ocean of green that was the Valley turned into a low, soft, dizzyingly perfumed cloud of pink and white. Uncounted bees and yellow jackets, newly born, raced out of their hives and holes, overwhelmed by this impossible banquet.

Then came the wind.

It roared off the Pacific Ocean, through the nearly uninhabited

air, it tore along the columns of trees, ripped the blossoms apart and carried them off in a fluttering flood of petals like foam rolling up a beach.

This perfumed blizzard hit Stevens Creek Boulevard, a two-lane road with a streetcar line down its center, that was the main road in the West Valley. It froze traffic, as drivers found themselves lost in a soft, muted whiteout. Only the

# UMBRUCHSTEUERUNG BEI MEHRSPALTENSATZ

```
h1 {
 break-before: column;
 break-inside: avoid-column;
 break-after: avoid-column;
}

div {
 width: 100px;
 column-width: 45px;
 column-gap: 0;
 column-rule: none;
}
```

# SILBENTRENNUNG

```
p {
 hyphens: none | manual | auto;
}
```

manual -> &shy;

none -> keine Trennungen

auto -> trennt immer dort, wo getrennt werden muss.

Ggf. Vendor-Präfixe, also „-moz-hyphens“, „-webkit-hyphens“

lang-Attribut verwenden, damit die Trennung nach Sprachregeln funktioniert!

# ERZWUNGENER WORTUMBRUCH

Für Wörter mit Überlänge und ohne Trennregel:

```
p {
 word-wrap: normal | break-word;
}
```

# SATZREGELN

```
p {
 text-align : justify;
 hyphens : auto;
 widows : 3; /* Schusterjungen-Regel */
 orphans : 3; /* Hurenkinder-Regel */
 word-spacing : normal;
 letter-spacing : normal;
 word-break : normal;
}
```

# SCHRIFTEINBINDUNG

# EINBINDUNG VON ZEICHENSÄTZEN

```
@font-face {
 font-family: 'LeagueGothic';
 src: url(LeagueGothic.otf);
}

header {
 font-family: 'LeagueGothic', Arial;
}

@font-face {
 font-family: 'Droid Sans';
 src: url(Droid_Sans.ttf);
}

@font-face {
 font-family: 'Droid Sans';
 src: url(Droid_Sans.woff);
}
```

# EINBINDEN VERSCHIEDENER FORMATE

```
@import url(http://fonts.googleapis.com/css?
family=Open+Sans:300italic,400italic,600italic,
700italic,800italic,400,300,600,700,800);
@font-face {
 font-family : OpenSans;
 src : url("../..//libraries/fonts/
Open_Sans/OpenSans-Light.ttf") format("ttf"),
 url("../..//libraries/fonts/Open_Sans/
OpenSans-Light.eot") format("eot");
 font-weight : 300;
 src : url("../..//libraries/fonts/
Open_Sans/OpenSans-Regular.ttf");
 font-weight : 400;
 src : url("../..//libraries/fonts/
Open_Sans/OpenSans-Semibold.ttf");
 font-weight : 600;
}
```



# DIE GRÖÖE VON ZEICHENSÄTZEN JUSTIEREN



```
p.beispiel {
 font-family:
 Verdana, Arial, sans-serif;
 font-size:12px;
 font-size-adjust:0.58;
}
```

Mit `font-size-adjust` kann man eine bestimmte x-Höhe erzwingen, ganz egal welche Schrift aus `font-family` eingesetzt wird. Im Beispiel wird die Schriftliste auf den Adjustwert der Verdana eingestellt.

Siehe auch

<http://www.webspaceworks.com/resources/fonts-web-typography/43/>

# TEXTKONTUREN

```
div {
 outline:
 text-fill-color: black;
 text-stroke-color: red;
 text-stroke-width: 4.00px;
}
```

# FARBEN UND FARBVERLÄUFE, IMAGE PROCESSING

# NEUE FARBMODELLE MIT TRANSPARENZ

Für Farben kann nun eine Deckkraft eingestellt werden. Der vierte Wert steht für einen Alpha-Kanal. Im Gegensatz zu Opacity, welches jeweils für das ganze Element auswirkt, können mit den Alpha Farben Fläche, Rand oder Text einzeln eingestellt werden.

```
color:
rgba(255, 0, 0, 0.75);
color:
hsla(360, 255, 128, 0.75);
```

# FARBVERLÄUFE - LINEAR UND RADIAL

```
background:
linear-gradient(
 to bottom,
 #1e5799 0%,
 #2989d8 50%,
 #207cca 51%,
 transparent 100%
);
```

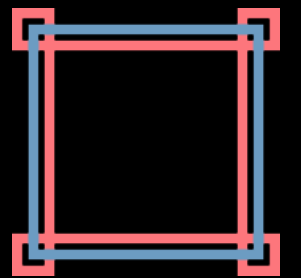
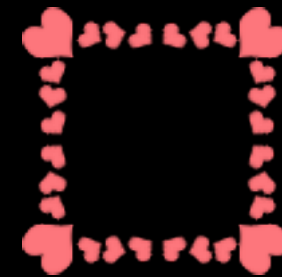
```
background:
radial-gradient(
 ellipse at center,
 #1e5799 0%,
 #2989d8 50%,
 #207cca 51%,
 #7db9e8 100%
);
```

# LINEARER FARBVERLAUF FÜR ALLE BROWSERVERSIONEN

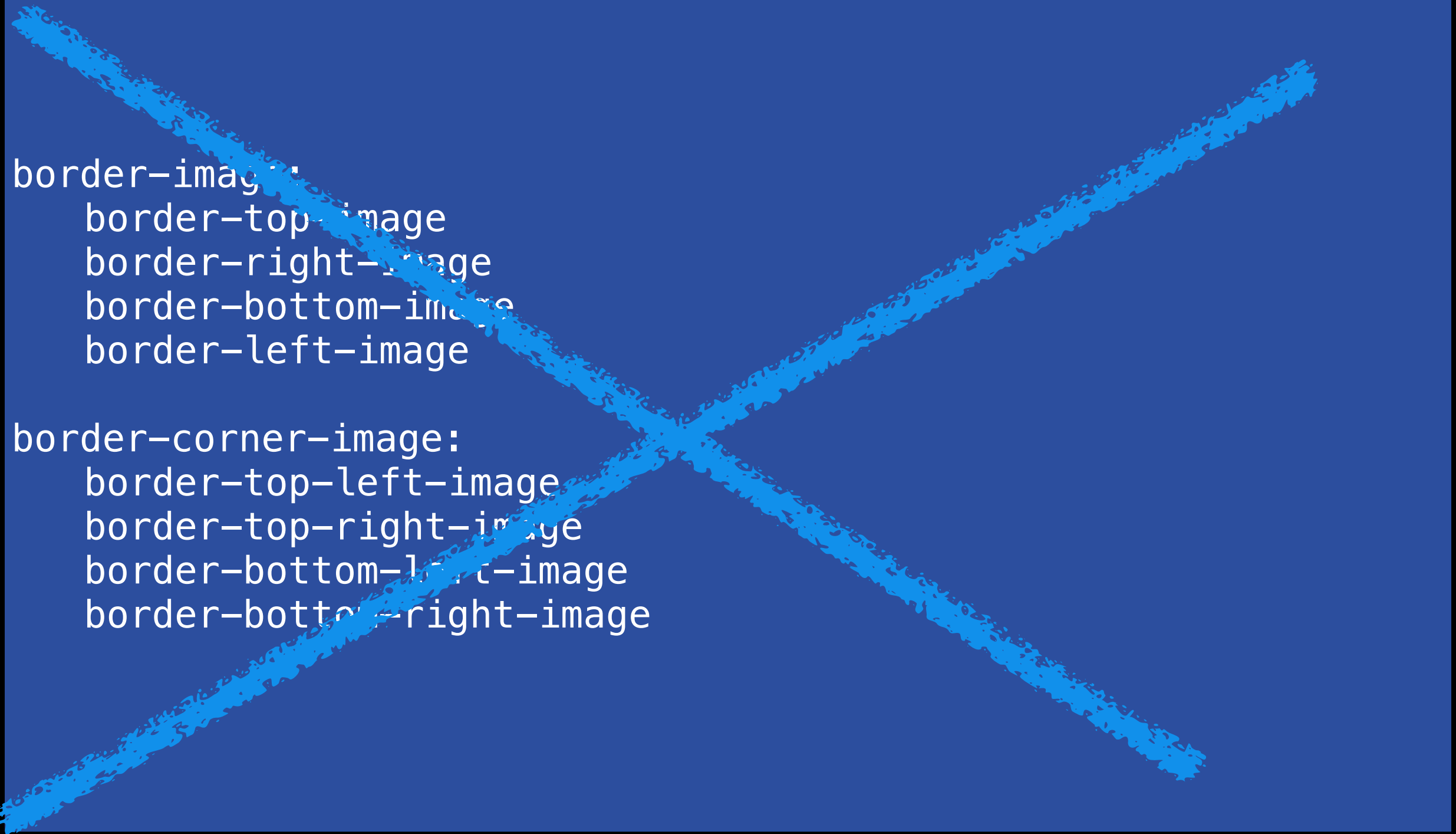
```
background: rgb(59,103,158); /* Old browsers */
background: -moz-linear-gradient(top, rgba(59,103,158,1) 0%,
 rgba(43,136,217,1) 50%, rgba(32,124,202,1) 51%, rgba(125,185,232,1) 100%); /
/* FF3.6+ */
background: -webkit-gradient(linear, left top, left bottom, color-
 stop(0%,rgba(59,103,158,1)), color-stop(50%,rgba(43,136,217,1)), color-
 stop(51%,rgba(32,124,202,1)), color-stop(100%,rgba(125,185,232,1))); /*
Chrome,Safari4+ */
background: -webkit-linear-gradient(top, rgba(59,103,158,1)
 0%,rgba(43,136,217,1) 50%,rgba(32,124,202,1) 51%,rgba(125,185,232,1)
 100%); /* Chrome10+,Safari5.1+ */
background: -o-linear-gradient(top, rgba(59,103,158,1)
 0%,rgba(43,136,217,1) 50%,rgba(32,124,202,1) 51%,rgba(125,185,232,1)
 100%); /* Opera 11.10+ */
background: -ms-linear-gradient(top, rgba(59,103,158,1)
 0%,rgba(43,136,217,1) 50%,rgba(32,124,202,1) 51%,rgba(125,185,232,1)
 100%); /* IE10+ */
background: linear-gradient(to bottom, rgba(59,103,158,1)
 0%,rgba(43,136,217,1) 50%,rgba(32,124,202,1) 51%,rgba(125,185,232,1)
 100%); /* W3C */
filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#3b679e',
 endColorstr='#7db9e8',GradientType=0); /* IE6-9 */
```

# CSS BORDER IMAGE

```
-webkit-border-image: url(border.png)
 30 30 round; /* Safari 5 */
-o-border-image :
 url(border.png)
 30 30 round; /* Opera */
border-image:
 url(border.png)
 30 30 round;
```



# BORDER-IMAGE UND BORDER-CORNER-IMAGE



```
border-image:
 border-top-image
 border-right-image
 border-bottom-image
 border-left-image

border-corner-image:
 border-top-left-image
 border-top-right-image
 border-bottom-left-image
 border-bottom-right-image
```



# ABGERUNDETE ECKEN FÜR ALLE BROWSER

```
border-radius: 5px;
border-left-top-radius: 10px;
```

```
-moz-border-radius : 5px;
-o-border-radius : 5px;
-webkit-border-radius : 5px;
-khtml-border-radius : 5px;
-ms-border-radius : 5px;
```

Für alte IE's  
`border-radius.htc`  
(Google Coderepository)

# SCHATTEN

```
text-shadow:
 10px /* X Versatz */
 10px /* Y Versatz */
 10px /* Blurradius*/
 rgba(64, 64, 64, 0.5)
;

box-shadow:
 10px /* X Versatz */
 10px /* Y Versatz */
 10px /* Blurradius*/
 rgba(0, 0, 128, 0.25);
```



Das ist Text  
Schlagschat

# REFLEKTION

```
box-reflect:
below | above | left | right
5px;


```



# FILTER FUNKTIONEN

```
filter : blur(5px)
 brightness(0.2)
 saturate(50%)
 hue-rotate(30deg)
 contrast(125%)
 invert(100%)
 greyscale(100%)
 sepia(100%)
```

```
;
```

```
-moz-filter
-webkit-filter
-o-filter
-ms-filter
```

ARBEITEN MIT BILDERN

NEUE BACKGROUND ATTRIBUTE -  
WALLPAPERS - SPRITES - BASE64  
ENCODING VERWENDEN

# RESPONSIVES LOGO

```
<div id="logo"><p class="sr-only">Company Name</p></div>
```

```
#logo {
 width : 100%;
 height : 100%;
 background : url(logo.gif) center center no-repeat;
 background-size : contain;
}
.sr-only {
 display : block;
 position : absolute;
 left : -10000px
}
```

# RESPONSIVER VOLLBILD HINTERGRUND

```
body {
 background : url(background.jpg) center center no-repeat;
 background-size : cover;
}
```

# BACKGROUND STACKING

In CSS 3 können mehrere Hintergrundeigenschaften kombiniert werden. Damit können auch komplexe grafische Designs mit cachebaren URLs (oder dem Application Cache) realisiert werden.

```
div {
 background :
 rgba(200,0,0,0.25),
 linear-gradient(
 to bottom,
 #1e5799 0%,
 #2989d8 50%,
 #207cca 51%,
 transparent 100%
);




 url(src/overlay.png) 10px center no-repeat,
 url(src/background.png) 10px center repeat-x;
}
```



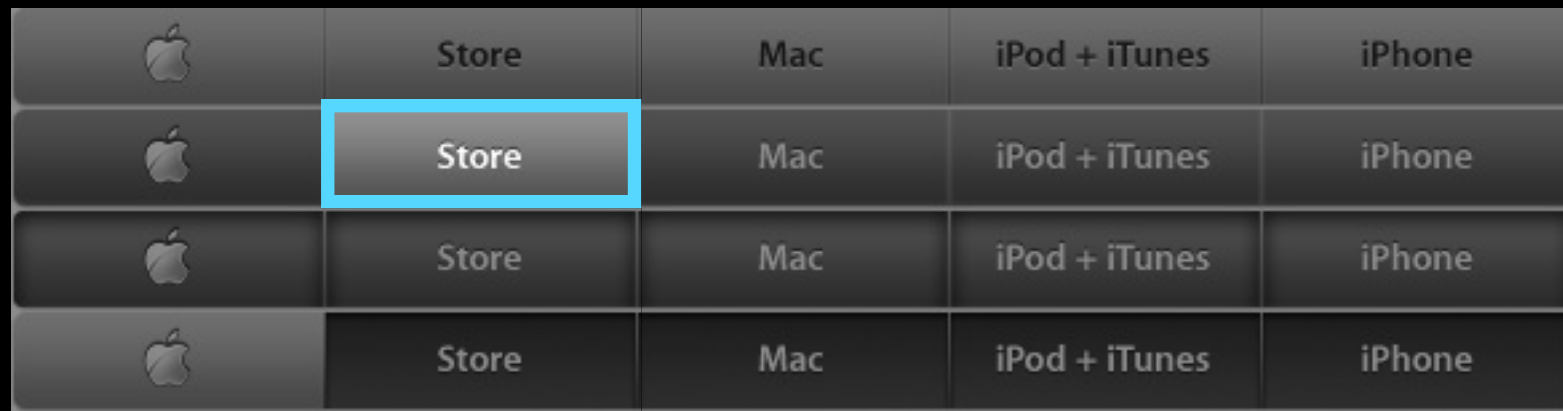
# BILDERTECHNIKEN - SPRITES

- Ein Sprite ist eine Grafik, die mit Hilfe eines Ausschnitts aus einem größeren Bild dargestellt wird.
- Bei einem CSS Sprite wird die Koordinate der Spritegesamtgrafik im Verhältnis zum kleineren darstellenden Element gewechselt.
- Der Begriff erinnert an die C64 Zeit, als mit Hilfe von Sprites Bewegungssequenzen von Spielerfiguren durch einen Wechsel der Einsprungsadresse im Grafikspeicher realisiert wurden.



|                                                                                    |       |     |               |        |
|------------------------------------------------------------------------------------|-------|-----|---------------|--------|
|   | Store | Mac | iPod + iTunes | iPhone |
|   | Store | Mac | iPod + iTunes | iPhone |
|   | Store | Mac | iPod + iTunes | iPhone |
|  |       | Mac | iPod + iTunes | iPhone |





```
.btn-store {
 display : block;
 width : 115px;
 height : 35px;
 background-image : url(sprite-image.png);
 background-position : -116px 0px;
 background-repeat : no-repeat;
}
.btn-store:hover {
 background-position: -116px -36px;
}
.btn-store:active {
 background-position: -116px -72px;
}
```

# SPRITES SIND SCHNELL

- 60 Grafiken benötigen 60 Requests.
- Ein Sprite benötigt nur einen Request.
- Mittlere Requestzeit: 50ms -  
Sprite -> Einsparung von  $59 \times 50\text{ms} = 2.941\text{ms}$
- Nur eine Datei zu bearbeiten und zu speichern.

NEUES LAYOUTEN

FLEX BOX

# WHAT IS FLEX BOX?

- The Flexbox Layout provides a more efficient way to lay out, align and distribute space among items in a container, even when their size is unknown and/or dynamic (thus the word "flex").
- The main idea behind the flex layout is to give the container the ability to alter its items' width/height (and order) to best fill the available space (mostly to accomodate to all kind of display devices and screen sizes). A flex container expands items to fill available free space, or shrinks them to prevent overflow.
- <http://css-tricks.com/snippets/css/a-guide-to-flexbox/>

# DAS HTML GERÜST

```
<div class="flex-header">
 <div class="flex-header-nav">[header nav]</div>
 <div class="flex-header-title">[header title]</div>
 <div class="flex-header-settings">[header settings]</
div>
</div>

<div class="flex-main">
 <div class="flex-main-nav">
 <p class="btn btn-primary">Lorem ipsum dolor
sit ...</p>
 </div>
 <div class="flex-main-content">
 <p>Lorem ipsum dolor sit ...</p>

 </div>
</div>
```

# DER FLEXBOX-CONTAINER

```
.flex-header {
 /* Webkit Syntax */
 display : -webkit-box;
 -webkit-box-orient : vertical;
 -webkit-box-pack : start; /* horizontale Anpassung */
 -webkit-box-align : stretch; /* vertikale Anpassung */

 /* W3 Syntax */
 display : box;
 box-align : stretch; /* vertikale Anpassung */
 box-pack : start; /* horizontale Anpassung */
 box-orient : vertical;
 box-sizing : border-box; /* all browser */
}
```



# FLEX-ITEMS

```
.flex-header-nav {
 -webkit-box-flex : 1;
 box-flex : 1;
 box-sizing : border-box;
}

.flex-header-title {
 -webkit-box-flex : 2;
 box-flex : 2;
 box-sizing : border-box;
}

.flex-header-settings {
 -webkit-box-flex : 0;
 box-flex : 0;
 box-sizing : border-box;
}
```