

Für ältere IEs können Sie außerdem auf Filter zurückgreifen. Das letzte Beispiel mit dem Browserfenster füllenden Hintergrundbild funktioniert mit folgender Ergänzung auch im IE7 oder IE8. Allerdings wird dabei das Hintergrundbild verzerrt, d.h., Breite und Höhe werden auf 100% gesetzt.

```
<!--[if lt IE 9]>
<style>
body {
  filter: progid:DXImageTransform.Microsoft.AlphaImageLoader(src='landschaft.jpg', sizingMethod='scale');
  -ms-filter: "progid:DXImageTransform.Microsoft.AlphaImageLoader(src='landschaft.jpg', sizingMethod='scale')";
  zoom: 1;
}
</style>
<![endif]-->
```

Listing 6-22

Mit dieser Ergänzung funktioniert ein Browserfenster füllendes Hintergrundbild auch im IE (bildschirmfuellend_ie.html).

6.8 Angenehme Schatten: box-shadow

Um Schatten um Elemente zu platzieren, musste man früher zu ausgefeilten Tricks greifen, im Zweifelsfall zu zusätzlichen Grafiken, wenn die Schatten halbwegs echt, d.h. leicht verschwommen, dargestellt werden sollen. In CSS3 gibt es eine eigene Eigenschaft, die Schatten bei Elementen definiert.

6.8.1 Schattensteuerung

box-shadow erwartet mindestens drei Angaben:

```
box-shadow: hsla(36, 4%, 25%, 0.8) 10px 20px;
```

Diese drei Angaben bedeuten:

- Die Farbe des Schattens. Hier empfehlen sich häufig halb transparente Farben über `rgba()` oder `hsla()` – das Vorgehen ist auch akzeptabel, da Browser, die box-shadow unterstützen, auch `rgba()` und `hsla()` implementiert haben.
- Der erste Zahlenwert bestimmt die Verschiebung von links.
- Der zweite Zahlenwert legt die Verschiebung von oben fest.

Im Beispiel ist der Schatten also um 10px von links und 20px von oben verschoben.

Grad der Verschwommenheit bei
box-shadow

Häufig werden Sie `box-shadow` noch mit einem dritten Zahlenwert verwenden, der den Grad der Verschwommenheit festlegt, im nächsten Beispiel etwa `4px` – da Safari `box-shadow` erst ab Version 5.1 ohne Präfix implementiert, sollten Sie die entsprechende Angabe für ältere Webkit-Browser ergänzen:

```
#eins {
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 10px 10px 4px;
  box-shadow: hsla(36, 4%, 25%, 0.8) 10px 20px 4px;
}
```

Ein höherer Wert bewirkt eine größere Verschwommenheit des Schattens.

```
box-shadow: hsla(36, 4%, 25%, 0.8) 10px 10px 20px;
```

Abb. 6-28

Drei Schattenbeispiele: links ohne
Angabe einer Verschwommenheit,
Mitte mit `4px`, links mit `20px`.



Allerdings funktioniert `box-shadow` bei Android nur, wenn Sie einen Wert für die Verschwommenheit angeben, bei `0px` wird kein Schatten angezeigt.⁸

Wenn Sie vier Angaben verwenden, so funktionieren diese genauso wie die vier Angaben bei `text-shadow` (siehe Kapitel 5, »Mehr Typografie – Webfonts, Textschatten und mehr«). Die Reihenfolge der drei Zahlenwerte muss wie angegeben sein, die Farbangabe kann hingegen am Anfang oder am Ende stehen.

Schattenpositionen

Über die beiden anderen Parameter bestimmen Sie die Position des Schattens. Bei zwei positiven Werten ist der Schatten rechts und unten:

```
#drei {
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 10px 20px 20px;
  box-shadow: hsla(36, 4%, 25%, 0.8) 10px 20px 20px;
}
```

Wenn der erste Wert negativ und der zweite positiv ist, ist der Schatten links und unten:

```
#vier {
```

⁸ <http://code.google.com/p/android/issues/detail?id=7531>

```
-webkit-box-shadow: hsla(36, 4%, 25%, 0.8) -10px 20px 20px;
box-shadow: hsla(36, 4%, 25%, 0.8) -10px 20px 20px;
}
```

Der erste Wert ist positiv, der zweite negativ: Der Schatten ist oben und rechts:

```
#fuenf {
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 10px -20px 20px;
  box-shadow: hsla(36, 4%, 25%, 0.8) 10px -20px 20px;
}
```

Zwei negative Werte: Der Schatten ist oben und links angeordnet.

```
#sechs {
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) -10px -20px 20px;
  box-shadow: hsla(36, 4%, 25%, 0.8) -10px -20px 20px;
}
```

Listing 6-23

Ausschnitt aus box-shadow.html



Abb. 6-29

Die unterschiedlichen Positionen für die Schatten

Durch 0px 10px würden Sie beispielsweise bewirken, dass ein Schatten nur von oben verschoben wird.

Standardmäßig wird der Schatten außerhalb der Box platziert (aber bei der Anordnung der weiteren Elemente bleibt er unberücksichtigt). Über das Schlüsselwort `inset` können Sie einen Schatten nach innen zeichnen.

Nach innen gehende Schatten

```
#sieben {
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) -10px -20px 20px inset;
  box-shadow: hsla(36, 4%, 25%, 0.8) -10px -20px 20px inset;
}
```



Abb. 6-30

Mit `inset` wird der Schatten innerhalb des Elements angezeigt, nicht außerhalb.

Beim folgenden Beispiel wird kein Versatz angegeben, aber einmal die Standarddarstellung (#acht) und einmal inset (#neun) gewählt.

Listing 6-24
Ausschnitt aus box-shadow.html

```
#acht {
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 20px;
  box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 20px ;
}
#neun {
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 20px inset;
  box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 20px inset;
}
```

Abb. 6-31

Dieses Mal ist der Schatten nicht versetzt: Links ist er außerdem normal, rechts wurde inset angegeben.



Mehrfache Schatten

Aber auch damit sind noch nicht alle Möglichkeiten erschöpft – Sie können nämlich auch mehrfache Schatten definieren. Diese geben Sie einfach durch Komma getrennt hintereinander an. Zur Verdeutlichung werden im folgenden Beispiel zwei unterschiedlich eingefärbte Schatten zugewiesen.

Listing 6-25
Schattenvariationen
(box-shadow.html)

```
#zehn {
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 10px 10px 20px, hsla(40, 81%, 65%, 0.8)-10px -10px 20px;
  box-shadow: hsla(36, 4%, 25%, 0.8) 10px 10px 20px, hsla(40, 81%, 65%, 0.8) -10px -10px 20px; ;
}
```

Abb. 6-32

Ein Element mit zwei Schatten



Spread Distance

Es ist noch eine vierte Längenangabe möglich, die die Ausbreitung des Schattens festlegt: die Spread Distance. Durch einen positiven Wert wird die Ausbreitung des Schattens um diesen Wert vergrößert, ein negativer Wert verringert entsprechend die Ausbreitung des Schattens um den angegebenen Wert.

Die folgenden Zeilen führen die unterschiedlichen Spielarten vor: Der Basisschatten ist immer derselbe: Er ist nicht verschoben und hat einen Verschwommenheitsgrad von 15px. Beim ersten Element ist kein Wert für Spread Distance angegeben:

```
div:nth-of-type(1) {  
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 15px;  
  box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 15px;  
}
```

Beim zweiten Element ist für Spread Distance 0 angegeben:

```
div:nth-of-type(2) {  
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 15px 0px;  
  box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 15px 0px;  
}
```

Bei den nächsten beiden Elementen wird die Spread Distance erhöht – 6px und dann 15px.

```
div:nth-of-type(3) {  
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 15px 6px;  
  box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 15px 6px;  
}  
div:nth-of-type(4) {  
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 15px 15px;  
  box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 15px 15px;  
}
```

Und im letzten Fall ist die Spread Distance negativ:

```
div:nth-of-type(5) {  
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 15px -7px;  
  box-shadow: hsla(36, 4%, 25%, 0.8) 0px 0px 15px -7px;  
}
```

Listing 6-26
Ausschnitt aus
box-shadow_spread.html

Abb. 6-33

Die Auswirkung von *Spread Distance* auf die Ausbreitung des Schattens. Sie sehen anhand von Element 1 und 2, dass kein Unterschied besteht zwischen einer *Spread Distance* von *opx* und keiner *Spread Distance*. Die Elemente 3 und 4 zeigen den Effekt von einem größeren Wert für *Spread Distance*, und Element 5 mit einer negativen *Spread Distance* ist recht elegant.



Die *Spread Distance* kann man für verschiedene Effekte nutzen. Beispielsweise lassen sich so beliebig viele Rahmen um ein Element ergänzen.

Abb. 6-34

Mehrfache Rahmen nur über die *Spread Distance*



Der innerste gelbe Rahmen kommt von *border*, die anderen Rahmen sind über mehrfache Schatten erstellt, die mit 0 Grad Verschwommenheit, aber einen deutlichen Wert bei *Spread Distance* besitzen.

```
div:nth-of-type(6) {
  border: 4px solid yellow;
  -webkit-box-shadow: 0 0 0 4px orange, 0 0 0 8px red, 0 0 0 12px blue ;
  box-shadow: 0 0 0 4px orange, 0 0 0 8px red, 0 0 0 12px blue ;
}
```

Da sich Schatten normalerweise überlagern und der zuerst genannte Schatten der oberste ist, müssen die anderen entsprechend größere Werte für *Spread Distance* haben, damit sie darunter zu sehen sind. Im Beispiel sind alle »Rahmen-Elemente« 4px breit.

Schatten nur an einer Seite

Mit *Spread Distance* können Sie einen Schatten auch nur an einer Seite anzeigen lassen. Dafür müssen Sie eine negative *Spread Distance* wählen, die genauso groß ist wie der Grad an Verschwommenheit.

Abb. 6-35

Ein Schatten jeweils nur an einer Seite



Für diese Beispiele ist der Verschwommenheitsgrad mit 4px angegeben, die Spread Distance jeweils mit -4px. Die unterschiedliche Position des einseitigen Schattens ergibt sich durch die ersten beiden Werte.

```
div:nth-of-type(7) {
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 0px 5px 4px -4px;
  box-shadow: hsla(36, 4%, 25%, 0.8) 0px 5px 4px -4px;
}
div:nth-of-type(8) {
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 0px -5px 4px -4px;
  box-shadow: hsla(36, 4%, 25%, 0.8) 0px -5px 4px -4px;
}
div:nth-of-type(9) {
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) -5px 0px 4px -4px;
  box-shadow: hsla(36, 4%, 25%, 0.8) -5px 0px 4px -4px;
}
div:nth-of-type(10) {
  -webkit-box-shadow: hsla(36, 4%, 25%, 0.8) 5px 0px 4px -4px;
  box-shadow: hsla(36, 4%, 25%, 0.8) 5px 0px 4px -4px;
}
```

Listing 6-27

Ausschnitt aus

box-shadow_spread.html

Die Idee für diese beiden zuletzt vorgestellten Einsatzbereiche der Spread Distance stammt übrigens von W3Conversions.⁹

6.8.2 Buttons

Für Buttons werden gerne Schatten eingesetzt und eventuell auch gleich mehrere wie im folgenden Beispiel:

```
.button {
  background-color: #333;
  background-color: rgba(0, 0, 0, 0.25);
  border: 1px solid #111;
  border: 1px solid rgba(0, 0, 0, 0.5);
  border-radius: 0.5em;
  width: 100px;
  height: 1.2em;
  padding: 0.5em 1.5em;
```

Listing 6-28

Button definieren (mehrfache_
schatten_button.html)

⁹ <http://blog.w3conversions.com/2011/09/css3-spread-value-and-box-shadow-on-one-side-only/>

```

/* Angabe mit -webkit- hier ausgelassen */
box-shadow:
  0 2px 6px rgba(0, 0, 0, 0.5),
  0 1px rgba(255, 255, 255, 0.3) inset,
  0 10px rgba(255, 255, 255, 0.2) inset,
  0 10px 20px rgba(255, 255, 255, 0.25) inset,
  0 -15px 30px rgba(0, 0, 0, 0.3) inset;
}

```

Im Beispiel sind sämtliche Schatten schwarz oder weiß mit einem größeren oder kleineren Wert an Transparenz. Das hat den Vorteil, dass die Buttons sich immer an verschiedenfarbige Hintergründe anpassen. Was die einzelnen Schatten bewirken, zeigt Abb. 6-36, wo bei jedem Element jeweils ein Schatten mehr hinzugekommen ist ...

Abb. 6-36

Immer ein Schatten mehr, bis am
Schluss der 3D-Button steht



Abb. 6-37

Die Buttons passen sich immer an die
Hintergrundfarbe der Seite an.



An diesem Beispiel sieht man auch eine weitere Besonderheit von Schatten: Sie passen sich auch Abrundungen an und bilden hier beispielsweise die obere Linie des Buttons nach. Damit können Sie box-shadow auch gut zur Erzeugung eines unteren Randes einsetzen.

Abb. 6-38

Links mit dem durch box-shadow
erzeugten unteren Rand, rechts ohne



Bei diesem Button wird die horizontale Linie übrigens durch einen Farbverlauf erzeugt – Thema von Kapitel 7, »Farbverläufe«. Der untere Rand, auf den es hier ankommt, ist durch folgende Angabe ausgelöst:

Listing 6-29

Ausschnitt aus `button_rand.html`

```
box-shadow: 0 4px 0 hsl(17, 69%, 45%);
```


6.8.3 Glühendes Textfeld

Ein schöner Effekt, den man inzwischen häufiger sieht, ist ein »Glühen« bei Eingabefeldern, wenn der Benutzer in ein solches klickt.



Abb. 6-39

Links: Normalzustand des Textfelds.

Rechts: Bei Klick beginnt es zu »glühen«.

Sehen wir uns an, wie sich das realisieren lässt. Zuerst kommen die klassischen Formatierungen beim Textfeld:

```
.textfeld {
  border: 1px solid #AD9C9C;
  width: 10em;
  line-height: 1.4;
  padding: 6px 12px;
```

Dann die CSS-Zutaten – abgerundete Ecken und ein doppelter Schatten, einer davon nach innen. Zusätzlich wird, damit das Glühen nicht abrupt erfolgt, eine Transition definiert – Thema von Kapitel 9, »Transitions und Animations«.

```
  border-radius: 6px;
  -webkit-box-shadow: 0 1px rgba(34, 25, 25, 0.15) inset, 0 1px #FFFFFF;
  box-shadow: 0 1px rgba(34, 25, 25, 0.15) inset, 0 1px #FFFFFF;
  display: block;
  -webkit-transition: all 0.8s ease-in-out 0s;
  transition: all 0.8s ease-in-out;
}
```

Bei Fokus wird die Rahmenfarbe verändert und ein zusätzlicher rötlicher Schatten mit großem Grad an Verschwommenheit – 14px – definiert, der Kürze halber ist auf die Angabe mit `-webkit-` verzichtet.

```
.textfeld:focus {
  border-color: #993300;
  box-shadow:
    0 1px rgba(34, 25, 25, 0.15) inset,
    0 1px rgba(255, 255, 255, 0.8),
    0 0 14px rgba(235, 82, 82, 0.35);
}
```

Listing 6-30

Textfeld mit Glüheffekt

([gluehen_klick_textfeld.html](#))

Wollen Sie bei einer Zustandsänderung einen weiteren Schatten ergänzen, müssen Sie die ursprünglichen Schatten zusätzlich angeben – sonst würden diese von der neuen Angabe überschrieben. Nur einen Schatten zu ergänzen funktioniert also nicht.

6.8.4 Browsergeschichten und Steckbrief

Alle modernen Browser unterstützen `box-shadow`, der IE ab Version 9. Die vierte Längenangabe, der Ausbreitungsgrad des Schattens, und die Angabe von `inset` sind allerdings nicht in Safari 4 implementiert, sondern erst ab Version 5.

Eigenschaft	Firefox	Chrome	Safari	Opera	IE	Alternativen
<code>box-shadow</code>	ja	ja	vor 5.1 mit -webkit-	ja	9	CSS3 PIE, alternativer border, Filter (kein Standard)

Für ältere IEs gibt es unterschiedliche Strategien:

- Anstelle von `box-shadow` können Sie einen Rahmen anzeigen lassen. Dafür ist folgende Vorgehensweise empfehlenswert: Alle Browser erhalten eine Hervorhebung durch einen Teilrahmen (beispielsweise rechts und unten). Über `Modernizr` ermitteln Sie, ob `box-shadow` unterstützt wird, und entfernen in diesem Fall den Rahmen und lassen einen `box-shadow` einblenden (siehe Kapitel 2, »Umgang mit unterschiedlichen Browsern«, und das dortige Listing `boxshadow_modernizr.html`).
- Sie können zum Nachbilden von `box-shadow` auch CSS3 PIE benutzen. Hierfür müssen Sie zweierlei machen: zum einen bei allen Deklarationen, die eine `box-shadow`-Angabe haben, `behavior: url(PIE/PIE.htc)` ergänzen, und zum anderen müssen Sie die `hsla()`-Angaben durch `rgba()`-Angaben ersetzen. Allerdings wird die Angabe von `inset` nicht unterstützt.

- Außerdem gibt es unterschiedliche Filter, die zur Simulation von box-shadow herangezogen werden, wie der Blur-Shadow-, DropShadow-, Shadow- oder Glow-Filter. Die Ergebnisse sind aber meist wenig überzeugend. Wie man über eine Kombination der unterschiedlichen Filter und zusätzliches HTML-Markup ansprechende Schatten erzeugt, erläutert ein Artikel bei dev.Opera.¹⁰ Hier klappt dann sogar auch ein innerer (inset) Schatten. Aber die Technik ist recht aufwendig.

box-shadow erspart einem den Einsatz von Bildern – genauso wie die CSS-Farbverläufe, die im nächsten Kapitel vorgestellt werden und die uns bei einem Button-Beispiel bereits begegnet sind.

¹⁰ <http://dev.opera.com/articles/view/cross-browser-box-shadows>