

Bilder in HTML-Seiten

Jobst-Hartmut Lüddecke

7. November 2011

Zusammenfassung

In dieser Lektion geht es um das Einbinden von Bildern in html-Seiten. Diese Bilder können – wie *Anker-Punkte* – *absolut* und *relativ* adressiert werden. Ferner können diese Bilder noch *absolut* und *relativ skaliert* werden. Zum Abschluss kommt noch *Fließtext* hinzu, den man links oder rechts um das Bild *fließen* lassen kann.

Als Ergänzung zur Lektion über die *Anker-Punkte* kommen noch mehrere Definitionen im *Cascading Style Sheet* für die unterschiedlichen Zustände der *Ankerpunkte* (Links) hinzu.

Inhaltsverzeichnis

1	Einbinden von Bildern	2
2	Skalieren von Bildern	2
2.1	Absolute Skalierung	3
2.2	Relative Skalierung	3
2.3	Nachteile der Skalierung	3
3	Bilder mit an der Seite fließenden Text	4
4	CSS-Definitionen für Ankerpunkte	5
4.1	a:link	5
4.2	a:visited	5
4.3	a:active	6
4.4	a:hover	6
4.5	a:focus	6
5	Übungen	7
5.1	Skalierung von Bildern	7
5.2	Fließtext	7
5.3	Bilder im <i>Cascading Style Sheet</i>	7
5.4	Ankerpunkte im <i>Cascading Style Sheet</i>	7

Abbildungsverzeichnis

1	Einbindung eines Bildes in einer html-Seite	2
2	Bild mit absoluter Skalierung	3
3	Bild mit relativer Skalierung	3
4	Fließtext mit <i>
</i> als Bremse	4
5	Fließtext als <i>Paragraph</i>	4
6	Definition eines noch nicht besuchten Ankerpunktes	5
7	Definition eines bereits besuchten Ankerpunktes	5
8	Definition eines gerade <i>angeklickten</i> Ankerpunktes	6
9	Definition eines Ankerpunktes über dem sich gerade der Mauszeiger befindet	6
10	Definition eines Ankerpunktes im Fokus	6

1 Einbinden von Bildern

Bilder lassen sich ähnlich wie *Anker-Punkte* **relativ** oder **absolut adressiert** einbinden.

Der entsprechende Tag nennt sich ** für *Image* (Bild) und die Option *src* für *Source* (Quelle) gehört immer dazu. Mit der Source wird die Bild-Datei wie ein *Ankerpunkt* genau spezifiziert (relativ, oder vollständige absolute URL).

Als zusätzliche Option sollte man noch ein *alt* verwenden, um eine *Text-Alternative* zum Bild anzubieten. Falls die Darstellung von Bildern im verwendeten Browser abgeschaltet ist – um den Datendurchsatz und damit die Geschwindigkeit zu erhöhen, oder um spezielle Ausgabegeräte für Behinderte (Braille) zu verwenden – hat man so wenigstens – auf dem Weg zur *Barriere-Freiheit* – die Möglichkeit eine minimale Information zum Bild zu liefern.

```

```

Abbildung 1: Einbindung eines Bildes in einer html-Seite

2 Skalieren von Bildern

Bilder haben in den häufigsten Fällen nicht genau das für das Seiten-Layout gewünschte Format, und so besteht der Wunsch das betreffende Bild zu verkleinern, oder zu vergrößern, also auf das gewünschte Format zu *skalieren*. Dafür gibt es die Optionen *height* und *width*. Doch Achtung, wenn man das Höhen-Breiten-Verhältnis (Aspekt Ratio) beibehalten will, sollte man entweder die Dimensionen des Bildes genau kennen, oder nur eine der beiden Optionen verwenden, eben die worauf es einem ankommt. Die andere Dimension wird in diesem Fall automatisch proportional, also im gleichen Höhen- und Breiten-Verhältnis angepasst.

```

```

Abbildung 2: Bild mit absoluter Skalierung

2.1 Absolute Skalierung

Im ersten Ansatz kann man die Höhe, oder Breite, des Bildes mit festen Werten genau festlegen. Dieser feste Wert für die Höhe, oder Breite, wird als ganze Zahl eingegeben und als Anzahl der Pixel (Bildpunkte) interpretiert. Im Beispiel ist das Bild also genau 400 Pixel breit. **Das Problem** ist nur, Sie kennen nicht die eingestellte Bildschirmauflösung und die eingestellte Fenstergröße des Benutzers ihrer HTML-Seite.

2.2 Relative Skalierung

Die *absolut* skalierten Bilder sehen nur mit **genau einer** eingestellten Bildschirmauflösung und **genau einer** Fenstergröße des Browsers gut aus. Da kann man zwar als Hinweis auf seine Seite schreiben *diese Seite ist für eine Auflösung von x mal y Bildpunkten optimiert* aber eigentlich ist dies ein schwacher Trost nur, weil man nicht richtig *relativ* skalieren kann. Dabei geht es doch ganz einfach:

```

```

Abbildung 3: Bild mit relativer Skalierung

Damit hat das Bild die Breite von 80% des **aktuell eingestellten** Browser-Fensters und dazu ist es egal, wie hoch die Bildschirmauflösung ist und egal, wie viel vom Bildschirm das Browser-Fenster verwendet. Dies ist aber auch genau der Punkt, wo sich viele **Generatoren** *die Karten legen*.

2.3 Nachteile der Skalierung

Einen Nachteil will ich aber nicht verschweigen: Die **Bildschärfe** ist nur optimal, wenn das Bild überhaupt nicht skaliert ist. Nur wenn wirklich ein Punkt des Bildes auch ein Pixel auf dem Bildschirm ist, kann es *chili-scharf* sein.

Bei allen Skalierungen gibt es irgendwelche Rundungsfehler, bzw. bei großen Vergrößerungen werden einzelne Bildpunkte zu *Mosaik-Steinchen* aufgebläht. Hier kommen wir aber jetzt in den Bereich von speziellen Bild-Formaten und Verfahrensweisen der Beschreibung und Abspeicherung von Bildern, die wiederum eine eigene Vorlesung füllen können. Ein sehr interessanter Aspekt dabei sind *fraktale Bilder*. Dabei wird der Inhalt des Bildes mathematisch beschrieben und das Bild für die entsprechende Vergrößerung neu generiert und kann bei Vergrößerung noch neue Details offenbaren. Diese Technik hat sich im Internet leider nicht durchsetzen können¹ und kommerziell ist es mir bisher nur in der CD-Version des *Otto-Katalogs* aufgefallen.

¹Die Entwicklungsfirma wurde aufgekauft und die Technologie verschwand von der Bildfläche – leider!

3 Bilder mit an der Seite fließenden Text

Möchte man ein noch komplizierteres Layout der Bilder machen, gibt es noch die Option *style*. Innerhalb von *style* gibt es wiederum diverse Optionen. Für uns interessant sind zunächst:

float: mit den Werten *left* und *right*. Bei *left* steht das Bild **links** und der Text umfließt das Bild am **rechten Rand**.

margin-left: definiert einen linken Rand, also zusätzlichen Platz links vom Bild. Entsprechend gilt: **margin-right** rechter Rand, **margin-top** oberer Rand und **margin-bottom** unterer Rand.

Sollen diese Ränder nicht nur für ein Bild, sondern für alle verwendeten Bilder gelten, definiert man diese Ränder geschickterweise besser in seinem *Cascading Style Sheet* für den Tag ``.

Wichtig bei der Verwendung von *float* ist die Kennzeichnung am **Ende des Textes**, also dort wo das Fließen wieder aufhören soll. Dies macht man mit der Option `style="clear:both"` innerhalb eines folgenden **Tags**.

```
<IMG SRC="bild.png"
  alt="ein Bild"
  width="40%"
  style="float: left;
        margin-left: 15pt;
        margin-right: 15pt;
        margin-top: 15pt;
        margin-bottom: 15pt">
<p>Text Text Text Text Text Text Text Text Text
Text Text Text Text Text Text Text Text Text
Text Text Text Text Text Text Text Text Text Text</p>
<br style="clear:both">
```

Abbildung 4: Fließtext mit `
` als Bremse

```
<IMG SRC="bild.png"
  alt="ein Bild"
  width="40%"
  style="float: left;
        margin-left: 0;
        margin-right: 5%;
        margin-top: 0;
        margin-bottom: 5%">
<p>Dies ist der Text, der neben dem Bild steht.
Dies ist der Text, der neben dem Bild steht.
</p>
<p style="clear:both">Hier geht der Text jetzt
unter dem Bild in voller Breite weiter.
</p>
```

Abbildung 5: Fließtext als *Paragraph*

4 CSS-Definitionen für Ankerpunkte

Mit diesen Definitionen können unterschiedliche Zustände von *Ankerpunkten* mit Hilfe von vordefinierten *Pseudo-Klassen* deutlich gemacht werden. D.h. der Browser kennt und verwendet verschiedene Klassen für die unterschiedlichen Zustände des *Ankerpunktes*. Mit der Definition dieser *Pseudo-Klassen* in Ihrem *Cascading Style Sheet* bestimmen Sie das Aussehen.

Achtung: Die selbst definierten **Klassen** verwenden einen **Punkt** als Trennsymbol, die vordefinierten **Pseudo-Klassen** einen **Doppelpunkt**.

4.1 a:link

Dies ist der noch nicht besuchte *Ankerpunkt*. D.h. dieser *Link* wurde noch nicht *angeklickt*. In diesem Beispiel ist der *Ankerpunkt* *dunkelblau* und *unterstrichen*.

```
a:link {
  color:navy;
  text-decoration:underline;
}
```

Abbildung 6: Definition eines noch nicht besuchten Ankerpunktes

4.2 a:visited

Diese *Pseudo-Klasse* bestimmt das Erscheinungsbild der bereits besuchten *Links*. In diesem Beispiel ist die Schriftfarbe des Anker-Punktes *Grün* und eine *Dekoration* ist *nicht vorhanden* (none). Dieser *Link* gilt jetzt so lange als *besucht*, solange die *Historie* des *Browsers* dies eingetragen hat. Löscht man die *Geschichte der Besuche* im *Browser* und startet diesen neu, ist alles wieder *jungfräulich*.

```
a:visited {
  color: green;
  text-decoration:none;
}
```

Abbildung 7: Definition eines bereits besuchten Ankerpunktes

4.3 a:active

Mit dieser Pseudo-Klasse wird bestimmt, wie ein Link der *gerade geklickt wird* aussehen soll. In diesem Beispiel wird er, solange die Maustaste gedrückt ist *Rot* und *unterstrichen*.

```
a:active {
  color: red;
  text-decoration:underline;
}
```

Abbildung 8: Definition eines gerade *angeklickten* Ankerpunktes

4.4 a:hover

Mit dieser *Pseudo-Klasse* wird ein *Link* angesprochen, über dem der *Maus-Zeiger* steht. In diesem Beispiel wird der Link *dunkelblau* und *unterstrichen* und die *Hintergrundfarbe* dieses Links wird 80% Rot, 80% Grün und 100% Blau, was ein *helles Blaugrau* ergibt.

Die Pseudo-Klasse *hover* lässt sich auch für interessante Effekte im Zusammenhang mit anderen Tags verwenden. Auf meiner Homepage habe ich diese Pseudo-Klasse dafür verwendet die Schriftfarbe des Absatzes über dem der Mauszeiger steht etwas zu verändern und damit hervorzuheben.

```
a:hover {
  color:rgb(0%,0%,40%);
  text-decoration:underline;
  background-color:rgb(80%,80%,100%);
}
```

Abbildung 9: Definition eines Ankerpunktes über dem sich gerade der Mauszeiger befindet

4.5 a:focus

Diese Variante macht Sinn bei Links auf der gleichen Seite. In diesem Fall wird der Link und die Marke auf die der Link zeigt farblich hinterlegt. Damit wird deutlich, wohin man mit diesem Ankerpunkt springt.

```
a:focus {
  color:rgb(0%,0%,40%);
  text-decoration:underline;
  background-color:rgb(100%,60%,60%);
}
```

Abbildung 10: Definition eines Ankerpunktes im Fokus

5 Übungen

5.1 Skalierung von Bildern

- Binden Sie ein beliebiges Bild mit **absoluter Skalierung** in ihrer bisherigen html-Übungsseite ein.
- Schauen Sie sich Ihre html-Übungsseite an und verändern Sie die Größe des Browser-Fensters und beachten sie dabei das Aussehen Ihrer Seite.
- Binden Sie ein beliebiges Bild mit **relativer Skalierung** in ihrer bisherigen html-Übungsseite ein.
- Schauen Sie sich Ihre html-Übungsseite an und verändern Sie die Größe des Browser-Fensters und beachten sie dabei das Aussehen Ihrer Seite.
- Vergleichen Sie beide Veränderungen.

5.2 Fließtext

- Bauen Sie verschiedene Variationen mit fließendem Text in Ihrer bisherigen *html*-Übungsseite ein und probieren diese mit verschiedenen *Browsern* aus.
- Informieren Sie sich über die verschiedene Möglichkeiten der *clear* Option und probieren diese aus.

5.3 Bilder im *Cascading Style Sheet*

- Überlegen Sie sich und machen sich bei *Stefan Münz* kundig, welche Style-Definitionen sinnvoll sein könnten.
- Spendieren Sie ihren Bildern im CSS einen Rand (oben, unten, links und rechts) von 5 bis 8mm.

5.4 Ankerpunkte im *Cascading Style Sheet*

- Bauen Sie Ihre Version der verschiedenen Definitionen der Ankerpunkte in Ihr bisheriges *Cascading Style Sheet* ein, dokumentieren es und probieren es aus.