

Kapitel 17 - Teil 4

Responsive Bilder

Inhalte in diesem Kapitel

- Bilder für hochauflösende Displays definieren
 - Unterschiedliche Bilder je nach Layouttyp definieren
 - Bildausschnitt auf dem Smartphone anzeigen lassen
-

Ansprüche an responsive Bilder

- Anzahl an Geräten, auf denen Webseiten dargestellt werden ist explodiert
- Normale Display stehen den hochauflösenden Displays gegenüber
- Unterschiedliche Bildgrößen für unterschiedliche Viewports und Layouts
- Manchmal je nach Größe auch unterschiedliche Bilder bereitstellen
- Unterschiedliche Bildformate
- Möglich mit: <https://responsiveimages.org>

Hochauflösende Displays

- iPhone 8 = 750px x 1.334px = hochauflösendes Display; verwendet aber nur 375 x 667 Referenzpixel => kann passieren, dass Bilder nicht so scharf dargestellt werden, weil intern mehr Pixel benutzt werden
- `srcset`

- mehrere zusätzliche Bilder angeben mit x-Deskriptor geschrieben = verrät, für welche Art von Display das Bild geeignet ist
 - 2x => Display mit doppelter Pixelanzahl in der Breite und in der Höhe
 - 3x => Display mit dreifacher Pixelanzahl
- Descriptor => beschreibt Eignung des Bildes für hochauflösende Displays

```

```

- Im Namen angegeben, welche Größe sie haben, Browser orientiert sich aber an dem x-Deskriptor
- Standardbild bei `src` schreiben, für höher auflösende Displays `srcset` mit Deskriptor
- **Beispiel: hochaufloesend.html -> Testen im Browser -> Entwicklertool -> Bildschirmgrößen testen (Icon)**

Unterschiedliche Bilder je nach Layout und Viewport

- Verschiedene Bildgrößen für unterschiedliche Viewports und Layouts
- Kleine Screens Layout oft einspaltig, größere Screens meist mehrspaltig => wissen, wie viel Platz das Bild in dieser Situation im Layout einnehmen wird/kann
- Browser weiß erst wie groß Bilder sind, wenn geladen sind (eigentlich)

```
<img sizes="(min-width: 600px) 33vw, 100vw"
srcset="http://placeholder.it/400x150.jpg 400vw,
http://placeholder.it/600x300.jpg 600vw"
src=http://placeholder.it/400x150.jpg alt="Alternativtext">
```

- **Beispiel: layout-rwd-img.html**
- `src`: Fallbacklösung, also Bild, das in altmodischen Browsern gezeigt wird, die mit `srcset` nicht umgehen können
- `srcset`: liefert Browser Liste von Bildern mit Deskriptoren; Breite steht im Deskriptor mit `w`
Dateiname = kann beliebig sein
- `sizes`: bei welchem Viewport das Bild in welcher Größe angezeigt werden soll

Art Direction: nicht nur für Künstler

- unterschiedliche Bilder zeigen
- `picture`-Element: dient als Container für mehrere Bildangaben, umfasst gesamte Bilddeklaration
- `source`-Element: einzelne Bildquellen mit dem `srcset`-Attribut
- `media`: Abfrage, wann das jeweilige Bild ausgewählt werden soll

```
<picture>
<source srcset="katze_auf_treppe.jpg" media="(min-width:
700px)">

</picture>
```

- wenn Browser `picture` und `source` Angaben nicht versteht wird `img` Bild angezeigt
- Beispiel für drei Bilder:

```
<picture>
<source srcset="extralarge.jpg" media="(min-width: 1100px)">
<source srcset="large.jpg" media="(min-width: 700px)">

</picture>
```

Verschiedene Bildformate

- Mit picture alle Browser mit optimalen Bildformaten versorgen
- Immer wieder neue Formate, die noch nicht alle Browser verstehen => z.B.: WebP-Format

```
<picture>
  <source srcset="bild.webp" type="image/webp">
  
</picture>
```

Info

- Responsive Images: Use Cases and Documented Code Snippets to Get You Started: <https://dev.opera.com/articles/responsive-images>
- <https://caniuse.com/#feat-srcset>
- <https://caniuse.com/#feat-picture>
- A responsive image polyfill: <http://scottjehl.github.io/picturefill/>