HTML5 and CSS3 for Mobile Applications

Scaling the Layout

Prof. Paul Krause, University of Surrey

Objectives for this lecture

- Responsive Web Design = Fluid Layouts + Media Queries

- Responsive typography

Media Queries

- General format:

```
@media screen and (max-width: 400px) {
  /* Query specific CSS goes here */
}
```

- Media queries can also test for:

  - width; height; device-width; device-height; orientation; aspect-ratio; color; resolution; scan ...

The basic structure

```html
10  <body>
11    <article>
12    <header>
13      <h1>Some Basics of Responsive Design</h1>
14    </header>
15    <nav>
16      <ul>
17        <li><a href="#section1">In the beginning</a></li>
18      </ul>
19    </nav>
20    <section id="sidebar">
21      <h1>Background</h1>
22      <p>
23        Lorem ipsum dolor sit amet....
24      </p>
25
26    </section>
27    <section id="section1">
28      <h1>In the beginning</h1>
29      <p>
30        Lorem ipsum dolor sit amet, ...
31      </p>
32    </section>
33    <footer>
34      This article © Paul Krause
35      <time datetime="2014-06-23">23 June 2014</time>
36    </footer>
37    </article>
38  </body>
```

Basic styling

```css
1  article {
2    margin-left: auto;
3    margin-right: auto;
4    width: 960px;
5  }
6
7  header {
8    width: 940px;
9  }
10
11 nav ul li {
12   display: inline-block;
13 }
14
15 section {
16   width: 700px;
17   float: right;
18 }
19
20 section#sidebar {
21   float: left;
22   width: 220px;
23 }
24
25 footer {
26   clear: both;
27 }
```

Basic styling

```css
1  article {
2      margin-left: auto;
3      margin-right: auto;
4      width: 960px;
5  }
6
7  header {
8      width: 940px;
9  }
10
11 nav ul li {
12     display: inline-block;
13 }
14
15 section {
16     width: 700px;
17     float: right;
18 }
19
20 section#sidebar {
21     float: left;
22     width: 220px;
23 }
24
25 footer {
26     clear: both;
27 }
```

Basic styling

```
 1  article {
 2    margin-left: auto;
 3    margin-right: auto;
 4    width: 960px;
 5  }
 6
 7  header {
 8    width: 940px;
 9  }
10
11  nav ul li {
12    display: inline-block;
13  }
14
15  section {
16    width: 700px;
17    float: right;
18  }
19
20  section#sidebar {
21    float: left;
22    width: 220px;
23  }
24
25  footer {
26    clear: both;
27  }
```

Basic styling

```
1   article {
2       margin-left: auto;
3       margin-right: auto;
4       width: 960px;
5   }
6
7   header {
8       width: 940px;
9   }
10
11  nav ul li {
12      display: inline-block;
13  }
14
15  section {
16      width: 700px;
17      float: right;
18  }
19
20  section#sidebar {
21      float: left;
22      width: 220px;
23  }
24
25  footer {
26      clear: both;
27  }
```

Basic styling

```css
1   article {
2       margin-left: auto;
3       margin-right: auto;
4       width: 960px;
5   }
6
7   header {
8       width: 940px;
9   }
10
11  nav ul li {
12      display: inline-block;
13  }
14
15  section {
16      width: 700px;
17      float: right;
18  }
19
20  section#sidebar {
21      float: left;
22      width: 220px;
23  }
24
25  footer {
26      clear: both;
27  }
```

Basic styling

```css
1   article {
2       margin-left: auto;
3       margin-right: auto;
4       width: 960px;
5   }
6
7   header {
8       width: 940px;
9   }
10
11  nav ul li {
12      display: inline-block;
13  }
14
15  section {
16      width: 700px;
17      float: right;
18  }
19
20  section#sidebar {
21      float: left;
22      width: 220px;
23  }
24
25  footer {
26      clear: both;
27  }
```

Basic styling

```css
article {
    margin-left: auto;
    margin-right: auto;
    width: 960px;
}

header {
    width: 940px;
}

nav ul li {
    display: inline-block;
}

section {
    width: 700px;
    float: right;
}

section#sidebar {
    float: left;
    width: 220px;
}

footer {
    clear: both;
}
```

Basic styling

```css
1   article {
2      margin-left: auto;
3      margin-right: auto;
4      width: 960px;
5   }
6
7   header {
8      width: 940px;
9   }
10
11  nav ul li {
12     display: inline-block;
13  }
14
15  section {
16     width: 700px;
17     float: right;
18  }
19
20  section#sidebar {
21     float: left;
22     width: 220px;
23  }
24
25  footer {
26     clear: both;
27  }
```

Add a media query to the .css

```css
41  @media screen and (max-width: 768px) {
42     article {
43        width: 768px;
44     }
45
46     header, section, section#sidebar, footer {
47        width: 748px;
48     }
49  }
```
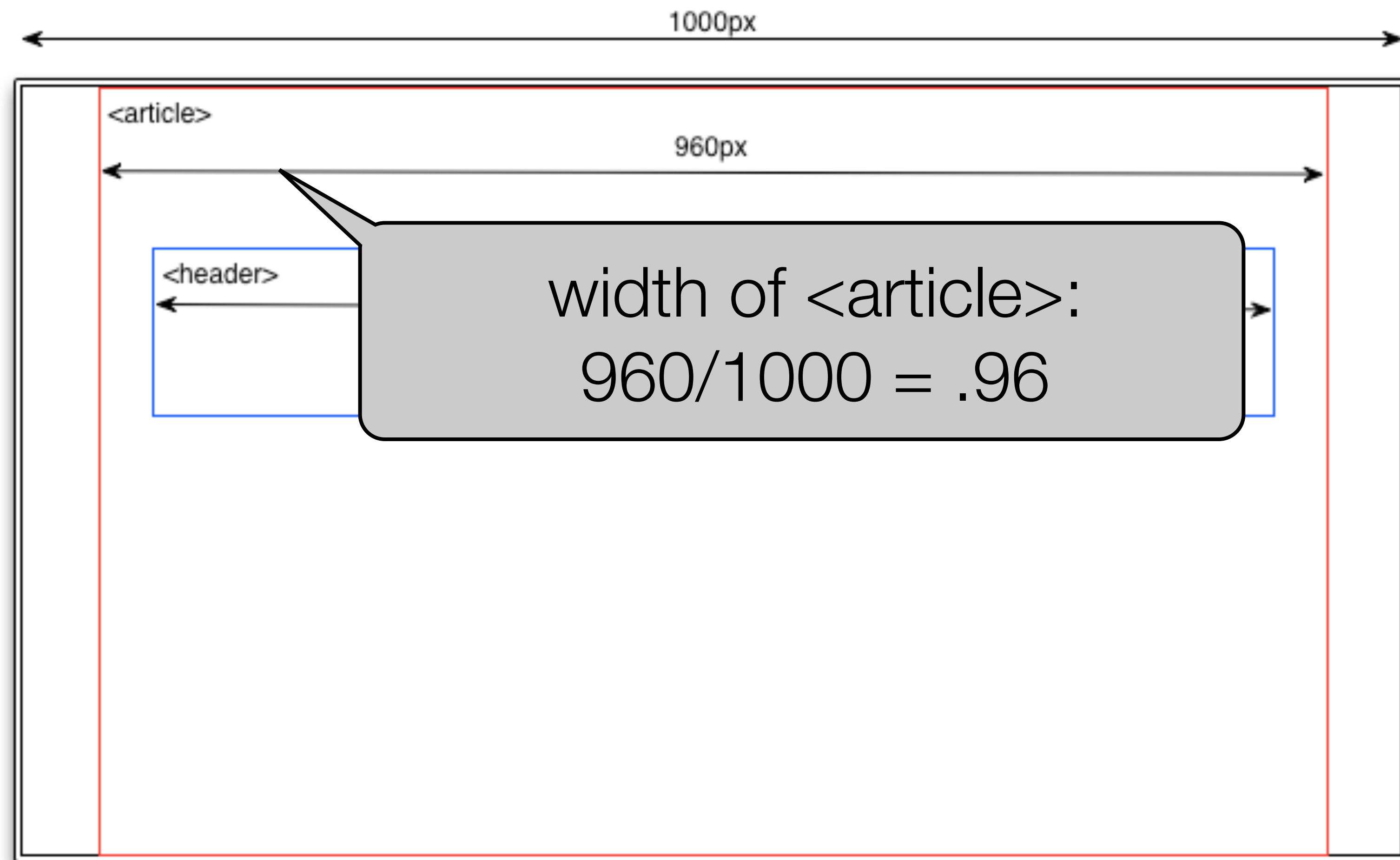
From Fixed to Proportional Layouts

- Our fixed width layout "clicks" into different versions as the media queries respond to specific breakpoints in viewport size

- Better to have our layout scale in proportion to the viewport size

  - With breakpoints to reconfigure the layout

- To perform the translation we use the magic formula:

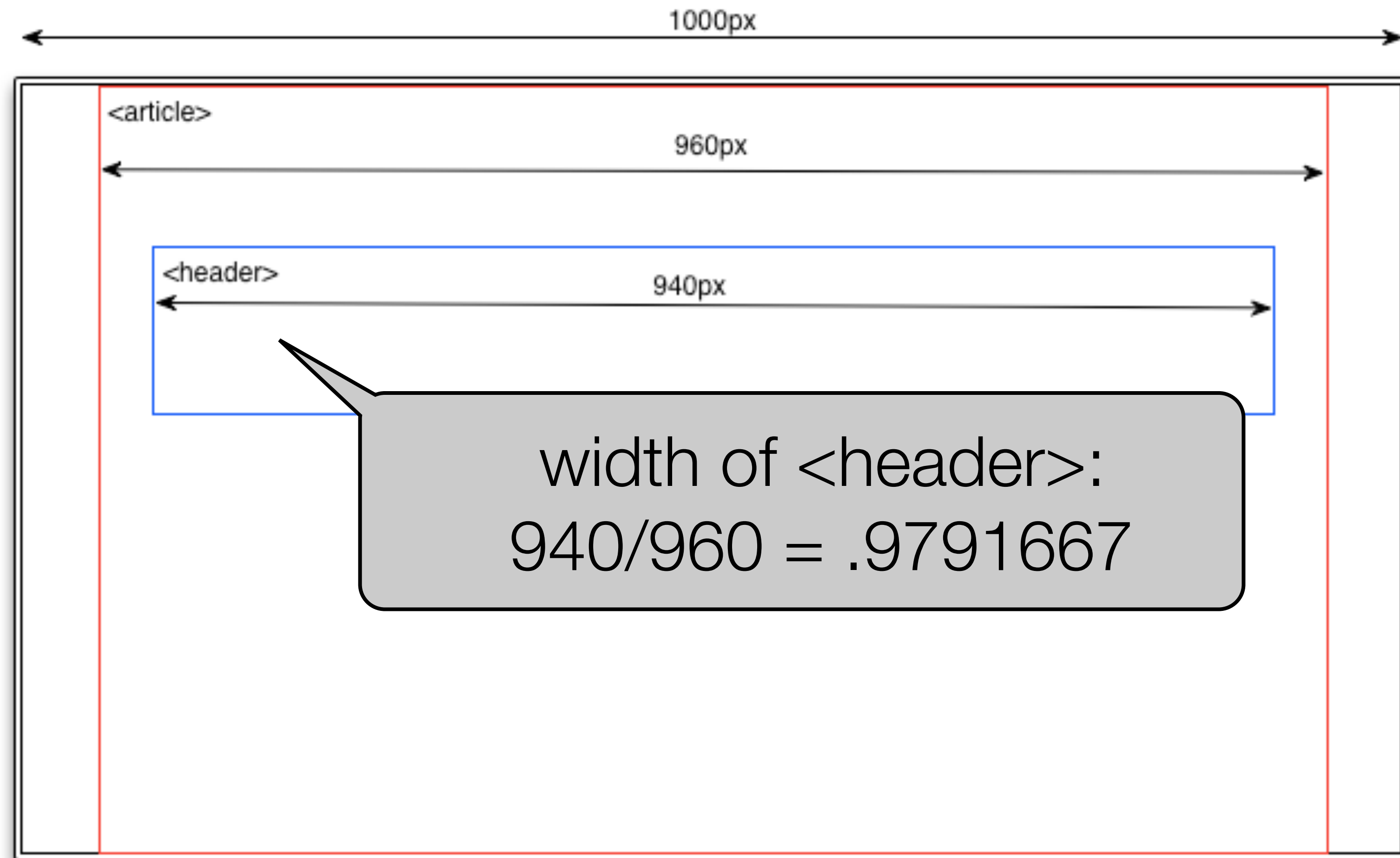 - target / context = result

Our Layout in Proportion

Our Layout in Proportion

Our Layout in Proportion

Proportional CSS

```css
1  article {
2    margin-left: auto;
3    margin-right: auto;
4    width: 96%;
5  }
6
7  header {
8    padding: 1%;
9    width: 98%;     /* 940/960 */
10   background-color: #779307;
11   color: #fff;
12   text-align: center;
13 }
14
15 nav ul li {
16   display: inline-block;
17 }
18
19 section {
20   padding-left: 2%;
21   padding-right: 2%;
22   width: 73%;   /* 700/960 */
23   float: right;
24   background-color: #dedede;
25 }
```

Responsive Typography

- Use "em"s rather than pixels for font size

- Remember the default font size is 16px

- To specify a font size in em use again:

  - **target/context = result**

  - e.g. 12px font would be 12/16 = .75em

- But remember the context will change as you move around within the document!

What we have done

- Start to think "relative sizes"

- Learnt to place media queries at natural breakpoints