# the design of HTML5

# HTML

| | |
|---|---|
| HTML 2.0 | 1995 |
| HTML 3.2 | 1997 |
| HTML 4.0 | 1997 |
| HTML 4.01 | 1999 |

XHTML 1.0 2000

XHTML 1.1 2001

XHTML 2

*Be conservative in what you send;*
*be liberal in what you accept.*

—Jon Postel,
The Robustness Principle

WHATWG 2004

W3C 2007

# HTML5

# the design of HTML5

# design principles

*This document describes the set of guiding principles used by the HTML Working Group for the development of HTML5. The principles offer guidance for the design of HTML in the areas of* ***compatibility, utility*** *and* ***interoperability****.*

—HTML Design Principles
w3.org/TR/html-design-principles

# avoid needless complexity

Simple solutions are preferred to complex ones, when possible.

# HTML 4.01

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"
   "http://www.w3.org/TR/html4/strict.dtd">
```

# XHTML 1.0

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

# HTML5 <!DOCTYPE html>

# HTML 4.01

```
<meta http-equiv="Content-Type"
content="text/html; charset=utf-8">
```

# XHTML 1.0

```
<?xml version="1.0" encoding="UTF-8"?>
<meta http-equiv="Content-Type"
content="text/html; charset=utf-8" />
```

# HTML5 <meta charset="utf-8">

```
<link rel="stylesheet" type="text/css"
href="file.css">
```

```
<script type="text/javascript">
</script>
```

# HTML5

# support existing content

Existing content often relies upon expected user agent processing and behaviour to function as intended.

```
<img src="foo" alt="bar" />
<p class="foo">Hello world</p>

<img src="foo" alt="bar">
<p class="foo">Hello world

<IMG SRC="foo" ALT="bar">
<P CLASS="foo">Hello world</P>

<img src=foo alt=bar>
<p class=foo>Hello world</p>
```

*Be conservative in what you send;*
*be liberal in what you accept.*

—Jon Postel,
The Robustness Principle

# solve real problems

Abstract architectures that don't address an existing need are less favoured than pragmatic solutions to problems that web content faces today.

# (X)HTML

```
<h2><a href="/path/to/resource">Headline text</a></h2>
<p><a href="/path/to/resource">Paragraph text.</a></p>
```

# HTML5

```
<a href="/path/to/resource">
<h2>Headline text</h2>
<p>Paragraph text.</p>
</a>
```

# pave the cowpaths

section

header

article

footer

aside

details

nav

figure

```
<body>
<div id="header">...</div>
<div id="navigation">...</div>
<div id="main">...</div>
<div id="sidebar">...</div>
<div id="footer">...</div>
</body>
```

```
<body>
<header>...</header>
<nav>...</nav>
<div id="main">...</div>
<aside>...</aside>
<footer>...</footer>
</body>
```

section

article

aside

nav

header

footer

details

figure

```
<div class="item">
<h2>...</h2>
<div class="meta">...</div>
<div class="content">
...
</div>
<div class="links">...</div>
</div>
```

```
<section class="item">
<header><h1>...</h1></header>
<footer class="meta">...</footer>
<div class="content">
...
</div>
<nav class="links">...</nav>
</section>
```

```
<section class="item">
<header><h1>...</h1></header>
<footer class="meta">...</footer>
<div class="content">
...
</div>
<nav class="links">...</nav>
</section>
```

*I would in fact prefer, instead of <H1>, <H2> etc for headings to have a nestable <SECTION>.. </SECTION> element, and a generic <H>..</H> which at any level within the sections would produce the required level of heading.*

—Tim Berners-Lee,
1991

# degrade gracefully

HTML 5 document conformance requirements should be designed so that Web content can degrade gracefully in older or less capable user agents, even when making use of new elements, attributes, APIs and content models.

input type="number"

input type="search"

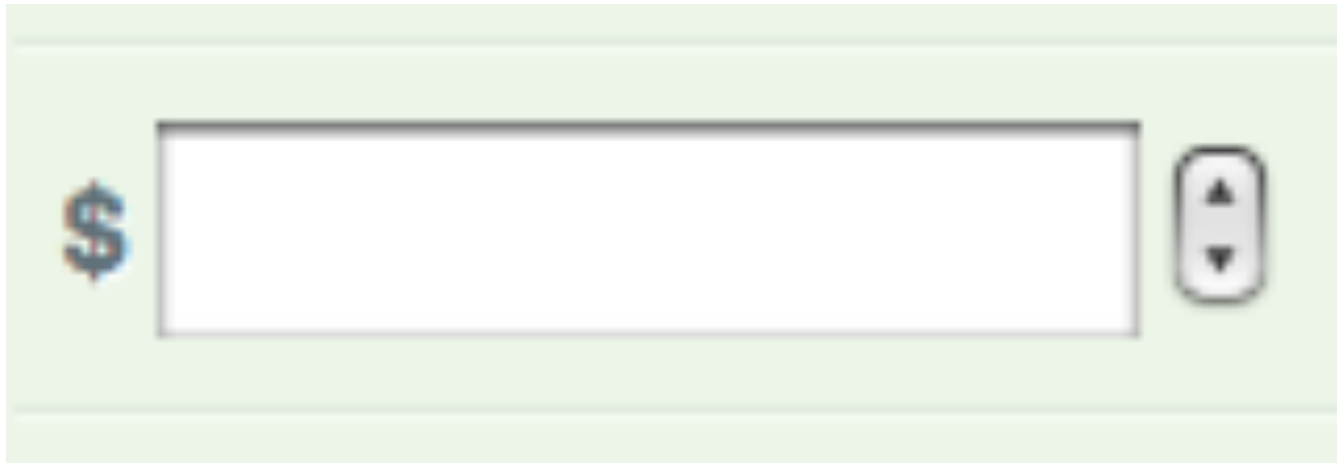
input type="range"

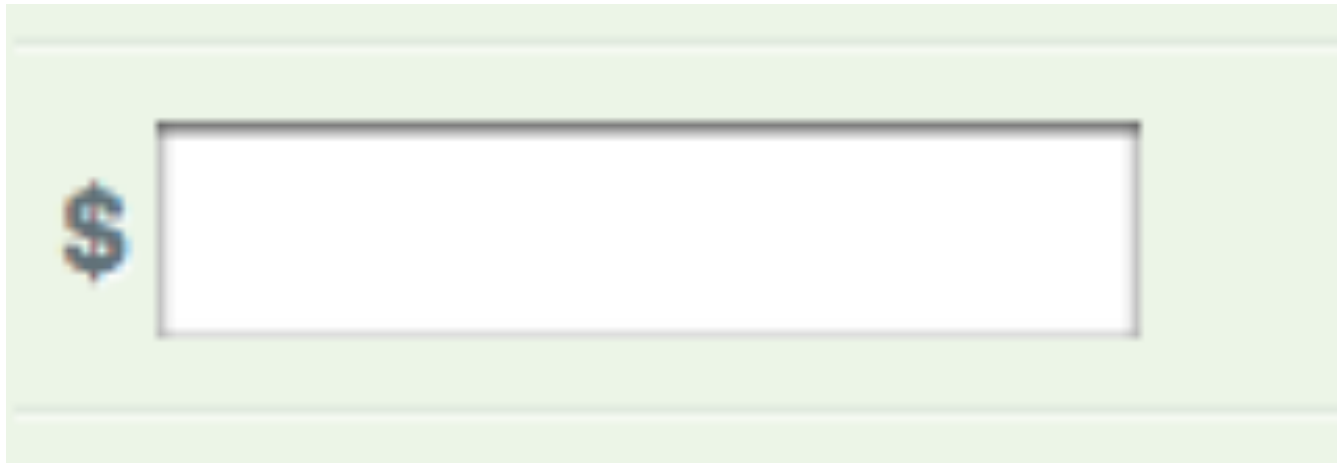
input type="email"

input type="date"

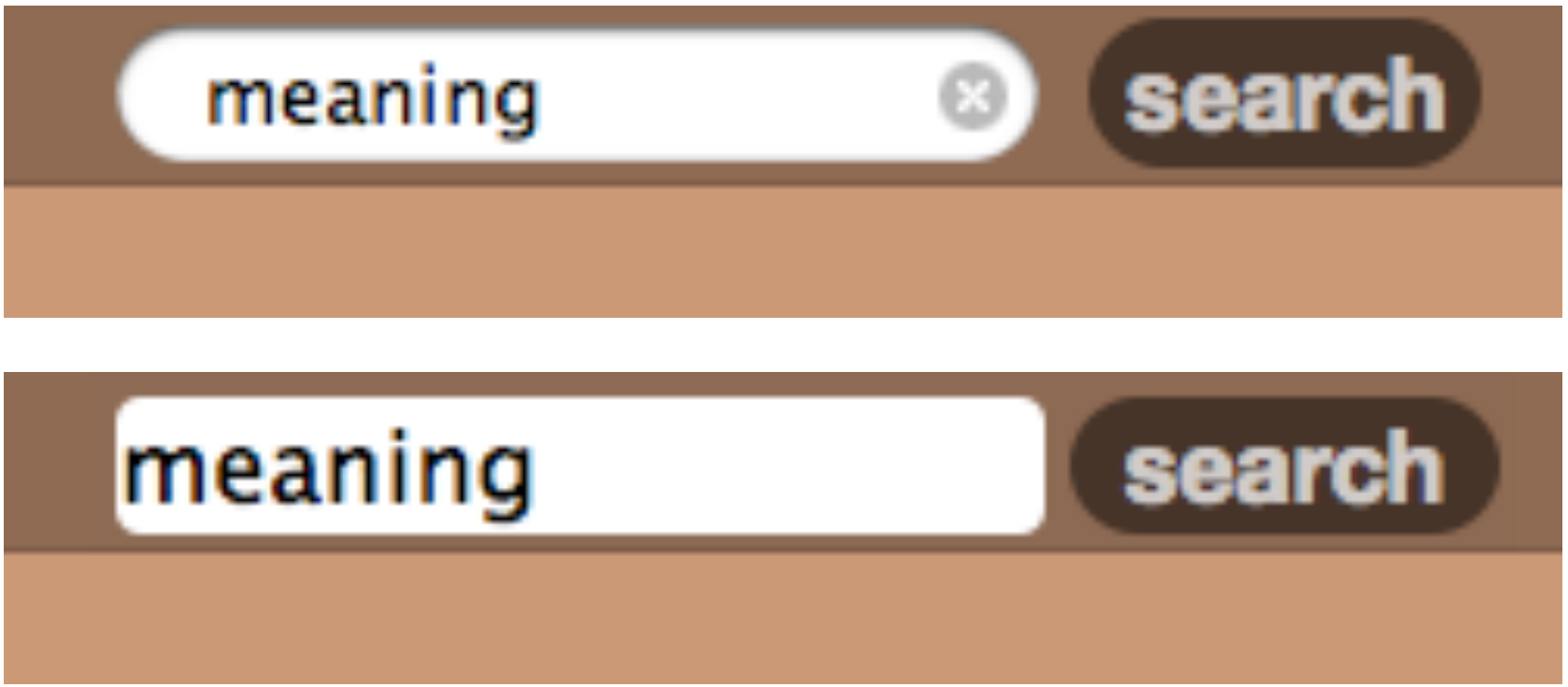
input type="url"

# input type="number"



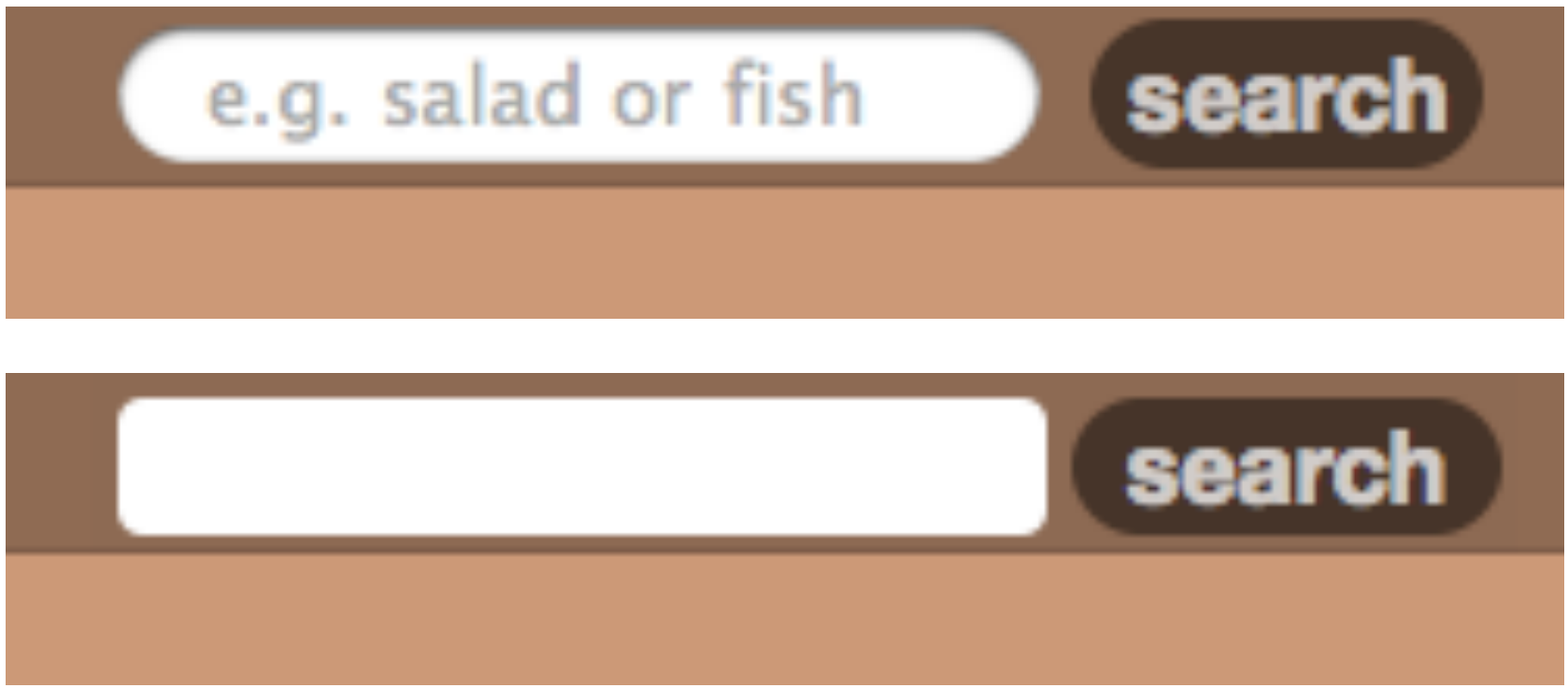



# input type="search"



# input type="search"

**placeholder="e.g. salad or fish"**



HTML5 video

Flash object

```
<video src="movie.mp4">

<!-- fallback content -->

</video>
```

```
<video src="movie.mp4">
<object data="movie.swf">
<!-- fallback content -->
</object>
</video>
```

```
<video src="movie.mp4">
<object data="movie.swf">
<a href="movie.mp4">download</a>
</object>
</video>
```

```
<video>
<source src="movie.mp4">
<source src="movie.ogv">
<object data="movie.swf">
<a href="movie.mp4">download</a>
</object>
</video>
```

```
<video>
<source src="movie.mp4">
<source src="movie.ogv">
<source src="movie.webm">
<object data="movie.swf">
<a href="movie.mp4">download</a>
</object>
</video>
```

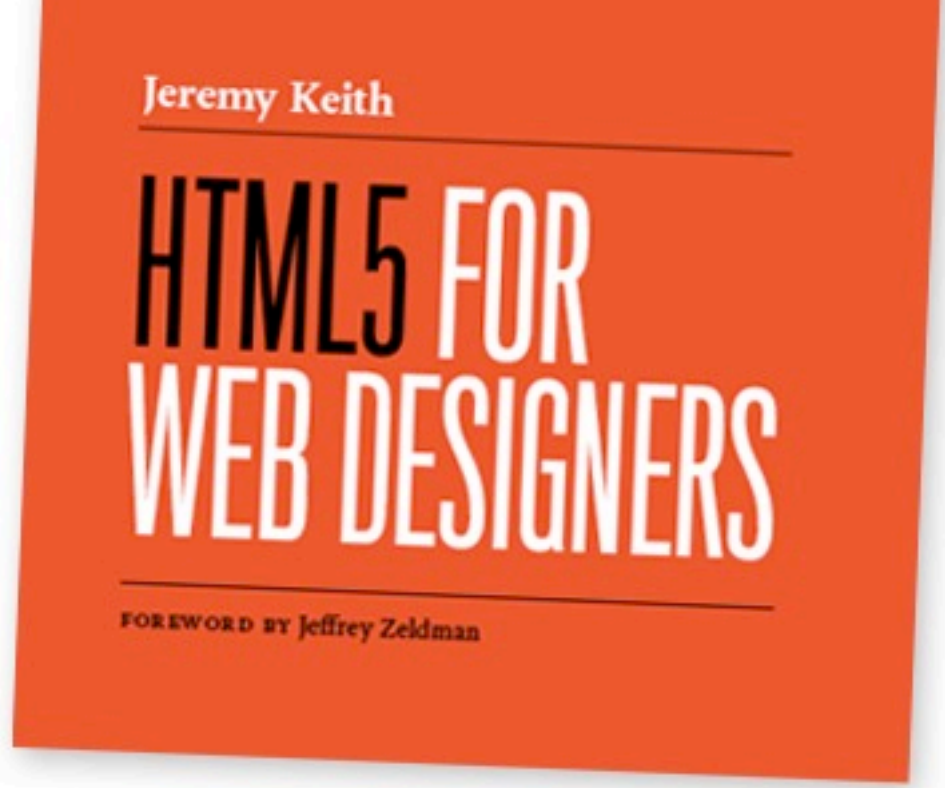
```
<video>
<source src="movie.mp4">                          1
<source src="movie.ogv">                          2
<source src="movie.webm">                         3
<object data="movie.swf">                         4
<a href="movie.mp4">download</a>                  5
</object>
</video>
```

*The value of a network is proportional to the square of the number of connected users of the system (n2).*

—Robert Metcalfe

# priority of constituencies

In case of conflict, consider **users** over **authors** over **implementors** over **specifiers** over **theoretical purity**.




@adactio

adactio.com

books.alistapart.com