

Progressive Enhancement

From CSS Discuss

Jump to: [navigation](#), [search](#)

Contents

[\[hide\]](#)

- [1 Progressive enhancement using CSS](#)
 - [1.1 What it is](#)
 - [1.2 Why do it?](#)
 - [1.3 The Techniques](#)
 - [1.3.1 Super-fluid design](#)
 - [1.3.2 Site navigation](#)
 - [1.3.3 Emphasising the active form element](#)
 - [1.3.4 Summary using headings](#)
 - [1.3.5 Bullet-point summary using projection mode](#)
 - [1.3.6 List counters](#)
 - [1.3.7 Internal anchors](#)
 - [1.3.8 Change of language when linking](#)
 - [1.3.9 Instant Question and Answer](#)
 - [1.3.10 Acronyms and abbreviations](#)
 - [1.4 References](#)
 - [1.5 About this page](#)

Progressive enhancement using CSS

What it is

The term "progressive enhancement" has been used as the flip-side to "Graceful Degradation". Both terms refer to the use of CSS to give users a different, yet always acceptable, experience depending on the CSS capabilities of their browser. The advanced CSS support of the most advanced browsers will be exploited, whilst users of less capable browsers will still receive readable and usable pages. There are many techniques of allowing graceful degradation for older browsers. Amongst the best known is the @import method for coping with [Netscape Four](#) and MsIEfour.

The present document concentrates on answering the following specific question:- What contribution can CSS make in persuading (Windows) users to move from legacy

browsers such as IE to browsers with more advanced and less buggy CSS support such as Opera, KHTML-based and Gecko-based browsers?

Why do it?

MsIEsix was a good browser when it was launched but is now over eight years old. Along with the even older IE5 and IE5.5, it dominates the browser market and is significantly holding back advanced CSS-based design. As such, it is a source of frustration for progressive designers. This situation will worsen over the next 6-7 years until IE7 (due end-2005 but only for Windows XP) and IE for Vista (the Windows XP successor, due in 2006) significantly supplant it. It's been suggested that IE6 is the new [Netscape Four](#). #Konqueror (the default browser for the KDE desktop on UNIX/Linux) users can switch stylesheets using the View > Use Stylesheet menu.

What should we, as designers, do over the next few years? We would like users to move beyond IE6 but they will ask: "Why should we move from the browser which came with Windows [IE] to one of these more advanced browsers you tell us about [Opera, Firefox and so on]? What will they actually do for us?"

One of our answers would relate to the many innovative features now delivered through the browser interface, e.g. tabbed browsing, popup blocking, RSS Newsfeeds. But another answer should relate to the advanced CSS support of these browsers. Optimize the site and support will certainly make development and maintenance easier, quicker and cheaper. It also enables us to make sites look more attractive. But CSS offers more than cosmetics.

This page offers techniques that go beyond cosmetics by offering additional functionality and usability. Without ignoring IE users (i.e. ensuring our sites will still work for them) we will give them solid reasons for upgrading and obtaining an improved browsing experience, not just through the browser "chrome" but through CSS as well.

The Techniques

There may be [High Pass Filter Css Hacks](#) which can discriminate between IE6 (and earlier) and more advanced browsers. At present, most of the the techniques described here rely on advanced CSS selectors and properties which are simply ignored by IE as being unknown. It's believed that IE behaves correctly here. If any part of a selector is unrecognised, the entire rule is ignored (i.e. all of its declarations). If a property is unrecognised, that declaration (property: value;) is ignored but not the other declarations in the same rule. The techniques come without guarantees: you need to have a sound knowledge of CSS and to do your own cross-browser testing. "Gecko", as used below, refers to browsers based on the Gecko engine such as Firefox, Mozilla and Netscape 7.

Some of the techniques are given minimal-case demonstrations here:-

<http://www.syntacticweb.co.uk/cssd/>

Super-fluid design

[Fluid Design](#) involves creating Web documents that adapt themselves well to the width of the user's viewport, from the largest monitors down to tiny handheld screens. With CSS2 technologies, a great deal can be achieved but there are limits. For example, many designs will break when viewed on handheld devices.

[CSS3 Media Queries](#) is an extension to Media Types. Media Queries can enable a big increase in document fluidity within a single medium. Currently (December 2005) only Opera has support for this Candidate Recommendation but expect implementation in other browsers soon. If you use Media Queries now on your site, your Opera users gain the benefit of increased fluidity (and usability) whilst other browsers will harmlessly ignore the rules. See [Fluid Design](#) for details and links.

Site navigation

Opera and Gecko support the use of the <link> element to provide site navigation, e.g:

```
<link rel="start" href="." title="Home page">
<link rel="prev" href="previous.htm" title="Previous page">
<link rel="next" href="next.htm" title="Next page">
<link rel="contents" href="sitemap.htm" title="Site map/contents">
<link rel="glossary" href="glossary.htm" title="Glossary">
<link rel="index" href="siteindx.htm" title="Index">
<link rel="help" href="sitehelp.htm" title="Site help">
<link rel="search" href="sitesrch.htm" title="Site search">
<link rel="copyright" href="copy.htm" title="Copyright statement">
<link rel="bookmark" href="#anchor1" title="Bookmark 1">
<link rel="bookmark" href="#anchor2" title="Bookmark 2">
```

Of course, you will still need to provide the most essential site navigation on the pages themselves for users of lesser browsers. It might not be quite as comprehensive. Example:-

[M2 Tutorial](#)

Emphasising the active form element

The element could be given a lighter background, or a 'ridge' or other distinctive border (or outline):- `input[type="text"]:focus, textarea:focus { border: 1px solid #FC0; }`
Supported by Opera, KHTML, Gecko. Contributed by [Thomas Scholz](#).

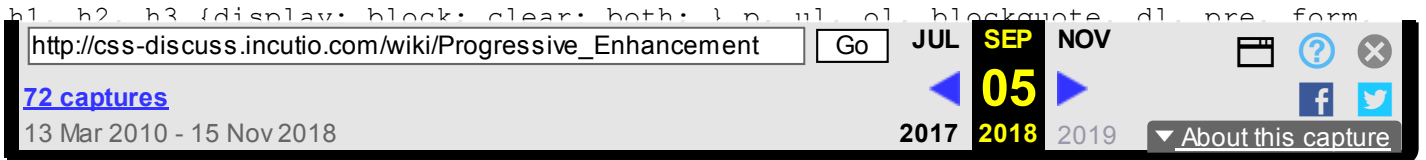
Summary using headings

It's sometimes appropriate to publish lengthy documents on the Web. The alternatives of providing a WORD or PDF document have their own drawbacks:-

<http://www.useit.com/alertbox/20030714.html>

An alternate stylesheet can be offered which suppresses the body text (and perhaps some or all of the graphics) so that only the headings (h1, h2 ...) are left. The users can then see an overview of the document structure. You could even replace the text of each section with a summary.

```
<link rel="alternate stylesheet" href="headings.css" type="text/css"
title="Headings only">
```



<http://www.syntacticweb.co.uk/cssd/>

Bullet-point summary using projection mode

Opera uses the media type "projection" when the user switches to full-screen mode (F11). The document contents can be re-purposed by presenting them in bullet-point form or comparable summarisation. Example:-[M2 Tutorial](#)

(although Mark doesn't reformat the material in this case - another example, anyone?).

List counters

Counters for nested ordered lists can be given increased readability, e.g:- 1.

- 1.a
- 1.b

instead of:- 1.

- a
- b

Supported by Opera. Contributed by [Thomas Scholz](#).

Internal anchors

The use of :target to indicate the target of internal anchors:-`body *:target { border: 1px dotted; }` Supported by Gecko, KHTML (CSS3?). Contributed by [Thomas Scholz](#). See also:-

http://devedge-temp.mozilla.org/viewsource/2003/target-selector/index_en.html

Change of language when linking

Useful when linking to a site in another language, or for multi-lingual sites:-

`[hreflang]:after { content: " (" attr(hreflang) ")"; }` Alternatively:-

`a[hreflang="de"]:after { content: " (german, deutsch)"; }` Supported by Opera, KHTML, Gecko. Contributed by [Thomas Scholz](#).

Instant Question and Answer

You will need to supply a link to a separate page containing the answer for users of lesser browsers and for accessibility. You could publish an entire quiz or teaching aid in this way.

```
.answer {
    display: none;
}
.question:hover .answer {
    display: block;
}
```

Here is a question. Hover over it to reveal the answer. Here is the answer.

Supported by Mozilla, KHTML, Opera.

Acronyms and abbreviations

A simple addition to the Q&A technique above: `acronym:hover:after, abbr:hover:after { content: " " attr(title); }`

References

- [Inclusive Web Design For the Future with Progressive Enhancement](#) by Steven Champeon and Nick Finck
- [Progressive Enhancement and the Future of Web Design](#) by Steve Champeon, on Webmonkey.
- [Progressive Enhancement: Paving the Way for Future Web Design](#) by Debra Chamra on hesketh.com.

About this page

This page was created in August 2003 by [Jim Wilkinson](#). Please treat it as a work in progress. Comments welcome on- or off-list as appropriate. I will try to elaborate on the techniques mentioned and add links to examples, as time permits. If you can add techniques, or information about suitable high-pass filters (see above), please do. Don't forget that the techniques must add value (be more than cosmetic) but must not be essential (unless an alternative is provided). I've drawn the techniques from various sources that I've read in the past and whose identity I've forgotten. Grateful

acknowledgments to them nonetheless.

Retrieved from "http://css-discuss.incutio.com/wiki/Progressive_Enhancement"

Views

- [Page](#)
- [Discussion](#)
- [View source](#)
- [History](#)

Personal tools

- [Log in](#)

Navigation

- [Main page](#)
- [Community portal](#)
- [Current events](#)
- [Recent changes](#)
- [Random page](#)
- [Help](#)

Search

Toolbox

- [What links here](#)
- [Related changes](#)
- [Special pages](#)
- [Printable version](#)
- [Permanent link](#)



- This page was last modified on 22 March 2010, at 01:47.
- This page has been accessed 51,175 times.
- [Privacy policy](#)
- [About CSS Discuss](#)
- [Disclaimers](#)