

Media Stylesheets

From CSS Discuss

Contents

- 1 Stylesheets for Media
 - 1.1 User agent support
 - 1.2 How Media Types work
 - 1.3 Media stylesheet strategies
 - 1.4 Media Queries - a CSS3 extension to Media Types
 - 1.5 About this page

Stylesheets for Media

This document outlines how to write stylesheets for various media types and provides links to further pages for each medium.

The CSS Recommendation lists the media types and gives a description of the characteristics of each of them. It also summarises how to specify media-dependent stylesheets. Briefly, you can do it in four ways but see below regarding browser (user agent or UA) support for these methods:-

- the HTML `<link>` element;
- the HTML `<style>` element;
- the CSS `@import` rule;
- the CSS `@media` rule.

If you don't specify the media value(s) then a default is applied: see Media Defaults for the defaults for each method. Netscape 4 has a particular quirk with the first of the four methods: see Media Defaults also.

- CSS2.1 Section 7 Media Types
(<https://web.archive.org/web/20181115022837/http://www.w3.org/TR/CSS21/media.html>)

See:

- Print Stylesheets - how to design stylesheets for the print medium.
- Handheld Stylesheets - how to design stylesheets for handheld devices.
- Projection Stylesheets - how to design stylesheets for projection devices.
- Speech Stylesheets - how to design stylesheets for speech synthesisers.

Opera's site has several good articles on using CSS to support multiple media:-

- Designing pages to work on all devices
(<https://web.archive.org/web/20181115022837/http://dev.opera.com/articles/mobile/>)

User agent support

UA support for media types is both incomplete and buggy.

Firstly, of the four methods (above) for invoking media-dependent stylesheets, the `<style>` and At Rule methods may not be as well supported by UAs. It's best to stick to the HTML `<link>` element.

Secondly, UAs may apply stylesheets to a medium when they shouldn't and may fail to apply them when they should. This is a particular problem when designing for handhelds.

To see which media type your browser is currently applying (and therefore the range of media types it's capable of applying), see Tarquin's Media Type test (<https://web.archive.org/web/20181115022837/http://www.howtcreate.co.uk/operaStuff/testMedia.html>)

Lachlan Hunt has a comprehensive media type test suite (<https://web.archive.org/web/20181115022837/http://www.lachy.id.au/dev/css/tests/media/>) although there are very few UA results recorded as yet. You might want to report your findings to him.

How Media Types work

This section should really be entitled "How Media Types are supposed to work". Reality, in terms of current UA support, has some catching up to do - see the individual media pages above.

What follows is my personal understanding of media types and of the possibilities they offer. I've kept it simple because there are many readers of css-d and of this Wiki who are new to CSS and perhaps even newer to the concept of designing for different media types.

A user agent (browser) is installed on a processor. That processor will have one or more output devices attached (e.g. a monitor, printer, projector or loudspeaker). It should be possible to specify to the UA which output device is currently being used, so that the UA can invoke the appropriate media stylesheet(s).

We're used to regarding the monitor as the default output device but we also expect our browser to invoke any `media="print"` stylesheets when we print or print-preview. All the major browsers do this.

By analogy, the Opera browser invokes any `media="projection"` stylesheets when we switch into full-screen mode (the assumption being that we may also be outputting to a projector). Likewise, it invokes any `media="handheld"` stylesheets when we switch it into "Small Screen" mode to emulate handhelds. Opera is unique in providing this support.

By further extension, it should be possible to specify to a UA that we are outputting to a speech synthesiser, so that `media="aural"` or `media="speech"` stylesheets are invoked. Such a UA would be a true "speech browser" and not merely a screen reader. Even Opera doesn't do this (yet). When browsers begin to offer this level of support, they will surely render existing screen readers (such as JAWS) obsolete.

A UA installed on a handheld device could support both `media="handheld"` (for its small screen) and `media="aural"` or `media="speech"` for its speaker. The latter possibility is exciting because it would offer an alternative to the physical limitations of very small screens when rendering Web content.

Media stylesheet strategies

There are two broad strategies.

The first is to have a complete, stand-alone stylesheet for each medium. For example, you might have one stylesheet for `media="screen"` and one for `media="print"`. Anyone viewing your site in projection mode or on a handheld device should see unstyled output. (But some UAs on handhelds will wrongly invoke `media="screen"` stylesheets - see Handheld Stylesheets .)

The second strategy is to have a common ("vanilla") stylesheet with `media="all"` (or without any media specified, which amounts to the same). You then have a smaller stylesheet for each medium you wish to support which modifies the common stylesheet by overriding its rules or by introducing extra rules.

My own preference is for the second strategy. As browsing using handhelds becomes more popular, we may need to move certain CSS rules from our common stylesheet to a separate screen stylesheet. At

present, it's all too easy to design our single stylesheet for screen only and to include rules that are not applicable to other media.

Media Queries - a CSS3 extension to Media Types

Opera (from 7.20) and Safari 3 have partial support for CSS3 Media Queries (<https://web.archive.org/web/20181115022837/http://www.w3.org/TR/css3-mediaqueries/>) (Candidate Recommendation as at January 2008). Opera uses them extensively on its own site.

Media Queries can be used, for example, to customise the display for narrower screens, thus:- `@media screen and (max-width: 480px) { rules ... }` You will see the effect only in Opera and Safari. Other browsers will harmlessly ignore the rule. Having narrowed your window, you may need to reload the document. Media Queries used in this way can enable a big increase in document fluidity within a single medium.

For more information see [Fluid Design](#) .

About this page

This page was created in June 2004 by Jim Wilkinson, who welcomes questions, contributions, comments and corrections.

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

- This page was last modified on 22 March 2010, at 02:35.