

**Excerpted Appendices**

Microformat Specification Reference

Microformat Design Patterns



# MICROFORMATS

Empowering Your Markup for Web 2.0

JOHN ALLSOPP

**friendsof**   
DESIGNER TO DESIGNER™  
an Apress® company

# Microformats: Empowering Your Markup for Web 2.0

John Allsopp



# Microformats: Empowering Your Markup for Web 2.0

Copyright © 2007 by John Allsopp

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

ISBN-13 (pbk): 978-1-59059814-6

ISBN-10 (pbk): 1-59059-814-8

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail [orders-ny@springer-sbm.com](mailto:orders-ny@springer-sbm.com), or visit [www.springeronline.com](http://www.springeronline.com).

For information on translations, please contact Apress directly at 2560 Ninth Street, Suite 219, Berkeley, CA 94710. Phone 510-549-5930, fax 510-549-5939, e-mail [info@apress.com](mailto:info@apress.com), or visit [www.apress.com](http://www.apress.com).

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is freely available to readers at [www.friendsofed.com](http://www.friendsofed.com) in the Downloads section.

Microformats logo used with kind permission of [microformats.org](http://microformats.org).

## Credits

**Lead Editor:** Chris Mills  
**Production Editor:** Laura Cheu

**Technical Reviewer:** Brian Suda  
**Composer:** Lynn L'Heureux

**Editorial Board:** Steve Anglin, Ewan Buckingham, Gary Cornell, Jason Gilmore, Jonathan Gennick, Jonathan Hassell, James Huddleston, Chris Mills, Matthew Moodie, Jeff Pepper, Paul Sarknas, Dominic Shakeshaft, Jim Sumser, Matt Wade  
**Artist:** April Milne  
**Proofreader:** Nancy Sixsmith

**Project Manager:** Toma Mulligan  
Beth Christmas

**Copy Edit Manager:** Nicole Flores

**Interior and Cover Designer:** Kurt Krames

**Assistant Production Director:** Kari Brooks-Copony  
**Manufacturing Director:** Tom Debolski

# APPENDIX A MICROFORMAT SPECIFICATION REFERENCE



This appendix details all of the microformats covered in this book. For each format, the authors or editors, a description, how the microformat is best used, the schema, and any restrictions or important things to keep in mind when using the microformat are noted. Each section also features one or more detailed examples for each microformat and points to publishing tools available to help developers.

### rel-license

rel-license uses the rel attribute on a hyperlink element (<a href=...>) to indicate the license or licenses for the current document. The document at the destination of a rel-license link is the license or one of the licenses for this document.

Where a rel-license is found inside an hReview, the license is for the review, rather than the page itself. This enables aggregating reviews from multiple sources that have different license conditions.

- **More information:** <http://microformats.org/wiki/rel-license>
- **Status:** Specification
- **Author:** Tantek Çelik (<http://tantek.com>)

### Usage

rel-license uses standard hyperlink elements (<a href=...>). The addition of a rel="license" attribute indicates that the destination document is a license for this document.

If a rel-license is embedded in an hReview, the license is the license for the review, not the document.

### Examples

Here are some examples of using rel-license:

```
<a rel="license" href="http://creativecommons.org/licenses/by/2.5/"> ↪  
Creative Commons Attribution 2.5 License<a>
```

```
<a rel="license" href="http://www.gnu.org/licenses/gpl.txt"> ↪  
GNU General Public License<a>
```

## Tools

Some tools that can help you create rel-license content are as follows:

- **Textpattern Microformats Plug-in** ([http://placenamehere.com/TXP/pnh\\_mf](http://placenamehere.com/TXP/pnh_mf)): Created by Chris Casciano, the pnh\_mf plug-in provides a simple way for adding various microformats, including rel-license, to Textpattern-based sites.
- **Dreamweaver Microformats Extension** ([www.webstandards.org/action/dwtf/microformats](http://www.webstandards.org/action/dwtf/microformats)): This suite, created by Drew McLellan and available from the Web Standards Project, provides tools for adding various microformats, including rel-license, to your HTML.

## Services

A couple of services using rel-license are as follows:

- **Yahoo Creative Commons Search** (<http://search.yahoo.com/cc>): This service enables searching for content based on its use license. The search uses rel-license to find content matching certain license criteria, such as content that may be used for commercial purposes.
- **Google Usage Rights Search** ([www.google.com/advanced\\_search?](http://www.google.com/advanced_search?)): This feature of Google's Advanced Search allows you to specify usage rights as a filter for searching. It uses rel-license to facilitate this searching.

## rel-tag

rel-tag uses a rel value of tag on a hyperlink element to “tag” or “label” the page, blog post, or other major part of the page it appears in. When a rel-tag link appears inside another microformat (e.g., in hAtom) it applies to that microformat and not the page as a whole.

By adding a rel-tag link to a page (or other microformat), you make an assertion about what this page or part of a page is about.

- **More information:** <http://microformats.org/wiki/rel-tag>
- **Status:** Specification
- **Authors:** Tantek Çelik (<http://tantek.com>) and Kevin Marks (<http://epeus.blogspot.com>)

## Usage

A rel-tag is a standard hyperlink element, with a rel value of (or including) tag. The href value points to an entry in a tag space. A **tag space** is a set of URLs of the form `http://tagspaceURL/tagvalue`. For example, `http://wikipedia.org/wiki/RSS` uses Wikipedia as a tag space, and the value of this tag is RSS. Also, `http://technorati.com/tag/RSS` uses Technorati as a tag space, but the value of this tag is identical to the previous one—RSS.

The actual value of the tag is the final segment of the url value, after the last /, meaning that the value of a tag is not dependent on the tag space being pointed to. Should the last character be a slash (/), then this final slash is ignored when determining the tag value. A resource should exist at the URL.

Technorati (`http://technorati.com/tag`) and Wikipedia (`http://wikipedia.org/wiki`) are common tag spaces, but anyone may implement his or her own tag space.

## Examples

`<a href="http://technorati.com/tag/microformats" rel="tag">microformats</a>` tags the page or major part of the page it is in as being about microformats.

`<a href="http://technorati.com/tag/cooking" lang="it" rel="tag">cucina</a>` tags the page or major part of the page it is in as being about cooking. This demonstrates how tagging works with internationalization—the human-friendly language here is Italian, the language of the page in which it appears, but the tag value is cooking, meaning that we can aggregate pages based on their tags despite being in different languages.

Of course, there is no reason why we could not use markup like this: `<a href="http://technorati.com/tag/cucina" lang="it" rel="tag">cucina</a>`. But in this case, the value of the rel-tag is cucina. There is no requirement that tag values be English-language words, but by using the English translation of a tag value, pages can be aggregated on the basis of their content, despite their human language.

## Tools

Some tools that can help you create rel-tag content are as follows.

For WordPress:

- **Simpletags**: [www.broobles.com/scripts/simpletags/](http://www.broobles.com/scripts/simpletags/)
- **Bunny's Technorati Tags**: <http://dev.wp-plugins.org/wiki/BunnysTechnoratiTags>

For Textpattern:

- **pnh\_mf**: [http://placenamehere.com/TXP/pnh\\_mf/](http://placenamehere.com/TXP/pnh_mf/)
- **tru\_tags**: [www.truist.com/blog/493/trutags-a-tagging-plugin-for-textpattern](http://www.truist.com/blog/493/trutags-a-tagging-plugin-for-textpattern)

For Bloxsom:

- **Tagging plug-in from Axel Beckett:** <http://noone.org/blog/tags/Tagging>

For Dreamweaver:

- **Dreamweaver Microformats Extension:** [www.webstandards.org/action/dwtf/microformats](http://www.webstandards.org/action/dwtf/microformats)

## Services

A number of services using rel-tag including the following.

Tag search:

- **Technorati Tag search** ([www.technorati.com/ping](http://www.technorati.com/ping)): Technorati Tag search indexes tagged blog posts and other content. Use Ping to ensure Technorati indexes your rel-tagged content.
- **IceRocket Tags** (<http://blogs.icerocket.com/tag>): Blog search engine IceRocket also allows for tag-based searching.

Debugging:

- **rel-lint tool** (<http://tools.microformatic.com/help/xhtml1/rel-lint>): You can “validate” your rel-tags using this service from Drew McLellan.
- **Operator** (<http://labs.mozilla.com/2006/12/introducing-operator/>): Created by Michael Kaply, Operator is an extension for Firefox 2 that extracts tag (and other microformatted) data from a page and enables various actions to be performed on it. In debug mode it shows errors to help debug problems.

## Publishers

The following publishers use rel-tag for user tags:

- **ODEO:** <http://odeo.com>
- **Eventful:** <http://evdb.com>
- **LiveJournal:** <http://news.livejournal.com/86492.html?thread=24881884>

## rel-nofollow

rel-nofollow uses a rel value of no-follow for links to external resources to indicate that search engines should not use the link to give additional PageRank to the linked page. Search engines such as Google typically use the number and authority of inbound links to a page to rank its place in search results.



## MICROFORMATS: EMPOWERING YOUR MARKUP FOR WEB 2.0

rel-nofollow, introduced by Google in an attempt to address the issue of link spam, particularly in blog comments, is not without controversy. A thorough criticism can be found at [www.nofollow.net](http://www.nofollow.net).

- **More information:** <http://microformats.org/wiki/rel-nofollow>
- **Status:** Specification
- **Editors:** Tantek Çelik (<http://tantek.com>) and Kevin Marks (<http://epeus.blogspot.com>)

### Usage

To use rel-nofollow, add a rel value of nofollow to links.

### Example

To link to a page while indicating to search engines that the link should not add to the linked page's PageRank, use the following form:

```
<a href="http://westciv.com" rel="nofollow">Style Master</a>
```

### Tools

Many blogging systems automatically add rel="nofollow" to links in comments or make this feature available through plug-ins or other extension mechanisms.

- WordPress 1.5 and higher automatically add nofollow to user-submitted comment links. There are plug-ins to disable this feature, including dofollow (<http://wp-plugins.net/plugin/sem-dofollow>) and Follow URL (<http://wp-plugins.net/?filter=follow+url>).
- Moveable Type and Textpattern both require a plug-in to enable rel-nofollow on user-submitted comments.

### Services

Recent research indicates that

- Google follows a link with the rel value of nofollow but does not index the linked page.
- MSN does not follow a link with rel-nofollow, nor does it index the page linked.
- Yahoo both follows a link with rel-nofollow and indexes the page linked.

## VoteLinks

Whereas `rel-nofollow` allows a publisher to withhold PageRank from a linked page, VoteLinks enables a publisher to explicitly endorse, vote against, or abstain from voting for the resource linked to. The specification leaves open the issue of whether the vote pertains to the page linked to, or what the page linked to represents.

- **More information:** <http://microformats.org/wiki/votelinks>
- **Status:** Specification
- **Authors:** Tantek Çelik (<http://tantek.com>) and Kevin Marks (<http://epeus.blogspot.com>)

## Usage

VoteLinks uses the `rev` attribute of a link, rather than the `rel` attribute. The `rev` attribute describes the relationship between the document containing the link and the linked document. Earlier versions of VoteLinks used the `rel` attribute (which describes the relationship between the linked document and the document containing the link), but this is logically incorrect and is now deprecated.

The `rev` attribute can take one of three values with VoteLinks:

- `vote-for`
- `vote-abstain`
- `vote-against`

## Examples

To vote for a linked resource, use a link like this:

```
<a href="http://westciv.com" rev="vote-for">Style Master</a>
```

To express indifference or abstain from voting, use a link like this:

```
<a href="http://westciv.com" rev="vote-abstain">Style Master</a>
```

To vote against a linked resource, use a link like this:

```
<a href="http://westciv.com" rev="vote-against">Style Master</a>
```

## Tools and services

At present there is not a significant amount of adoption of VoteLinks in tools or services.

## XHTML Friends Network (XFN)

XFN enables publishers to encode in HTML, XHTML, or XML markup for common professional and personal relationships, by means of the `link` element and the `rel` attribute.

- **More information:** [www.gmpg.org/xfn](http://www.gmpg.org/xfn)
- **Status:** Specification version 1.1
- **Authors:** Tantek Çelik (<http://tantek.com>), Matthew Mullenweg (<http://photomatt.net>), and Eric Meyer (<http://meyerweb.com>)

### Usage

XFN uses a set of possible values for the `rel` attribute. These are grouped into seven categories. Some of these values are mutually exclusive. The designations from the XFN specification are as follows:

- **Friendship**
  - **Friend:** Someone you are a friend to. (I have a feeling the developers of XFN kept this one vague, as what constitutes a “friend” will vary perhaps significantly from person to person.)
  - **Acquaintance:** Someone you have exchanged greetings with and not much (if anything) more—maybe a short conversation or two.
  - **Contact:** Someone you know how to get in touch with.
- **Physical**
  - **Met:** Someone you have actually met in person.
- **Professional**
  - **Co-worker:** Someone you work with or who works at the same organization as you.
  - **Colleague:** Someone in the your same field of study/activity.
- **Geographical**
  - **Co-resident:** Someone you share a street address with.
  - **Neighbor:** Someone who lives nearby, perhaps only at an adjacent street address or doorway. (Like “friend,” “neighbor” is a more fluid concept, varying from context to context.)
- **Family**
  - **Child:** Your genetic offspring, or someone you have adopted and take care of.
  - **Parent:** Your biological (or adopted) mother or father.
  - **Sibling:** Someone you share a parent with.
  - **Spouse:** Someone you are married to.
  - **Kin:** A relative, or someone you consider part of your extended family who doesn’t fit into any of the preceding criteria.

- **Romantic**
  - **Muse:** Someone who brings you inspiration.
  - **Crush:** Someone you have a crush on.
  - **Date:** Someone you are dating.
  - **Sweetheart:** Someone with whom you are intimate and at least somewhat committed.
- **Identity**
  - **Me:** A link to yourself at a different URL.

## Examples

The following are some examples of XFN in action:

```
<a href="http://marxandmarzipan.com" ➡
  rel="friend met colleague neighbor">Maxine, the co-founder ➡
  of westciv</a>
```

```
<a href="http://blogs.westciv.com/dog_or_higher" rel="me"> ➡
  Also by me</a>
```

## Tools

Some tools that can help you create XFN content are as follows:

- **WordPress 2** (<http://wordpress.com>): WordPress has direct support for XFN in its standard “add link” feature.
- **XFN Creator** (<http://gmpg.org/xfn/creator>): This online-based tool creates XFN links.
- **Autoxfn** (<http://weblog.scifihihi.com/2004/02/08/autoxfn-10>): This is a plug-in for Bloxsum.
- **Dreamweaver Microformats Extension** ([www.webstandards.org/action/dwtf/microformats](http://www.webstandards.org/action/dwtf/microformats)): The Dreamweaver microformats toolbar supports XFN.

## Services

The following services use XFN content:

- **Rubhub** (<http://rubhub.com>): This search engine uses XFN to present information about the relationships between sites.
- **Flickr** (<http://flickr.com>): Flickr uses XFN to publish relationships on its profile pages.
- **Upcoming** (<http://upcoming.org>): This site uses XFN to publish user-defined relationships.
- **Cork'd** (<http://corkd.com>): This wine community site uses XFN to publish the relationships between members.

## geo

The geo microformat enables publishers to add geodata about the location of a person, place, event, or thing to a page. The geo microformat is a one-to-one representation of the geo property in vCard, and thus a subset of hCard.

geo microformatted content has a root element with a class value of geo and two properties, longitude and latitude.

- **More information:** <http://microformats.org/wiki/geo>
- **Status:** Draft (see [http://microformats.org/wiki/geo#Draft\\_Specification](http://microformats.org/wiki/geo#Draft_Specification) for the latest draft specification)
- **Author:** Tantek Çelik (<http://tantek.com>)

## Usage

Use the geo format to indicate a geographic location. The geo specification notes the following:

*If the publisher knows and is publishing the address of the location, OR if the address of the location was what was actually entered by a human, and the publisher simply turned that into lat/long using some sort of a service, then the publisher SHOULD use adr to publish the actual human entered address information since that communicates far more semantic information than a simple geo lat/long coordinate.*

## Examples

There are two forms of the geo microformat: a standard “compound” form and an abbreviated form. Here’s an example of the standard compound microformat:

```
<span class="geo">  
  <span class="latitude">27.976628</span>, <span class="longitude"> ↵  
    86.933302</span>  
</span>
```

Here’s an abbreviated form when the longitude and latitude shouldn’t actually be visible, but should still be machine-readable:

```
Apollo 11 splashed down <abbr title="13.150000;169.150000" ↵  
class="geo"> 400 miles (640 km) South West of Wake Island, in the ↵  
North Pacific Ocean</abbr>
```

## Tools

Some tools that can help you create geo content are as follows:

- **GEO Microformats to XML** (<http://suda.co.uk/projects/microformats/geo>): Brian Suda's XSLT service extracts geodata, and it can mash up this data with Google Maps.
- **Operator** (<http://labs.mozilla.com/2006/12/introducing-operator/>): Created by Michael Kaply, Operator is an extension for Firefox 2 that extracts geodata from a page and enables various actions on it.

## Services

The following services use geo content:

- **Flickr** (<http://flickr.com>): Flickr uses geo for all geotagged photos and provides a drag-and-drop interface for geotagging.
- **Multimap** ([www.multimap.com](http://www.multimap.com)): Multimap uses geo to publish the location of a location search result.
- **Open Guides** (<http://openguides.org>) and **Wikitravel** (<http://wikitravel.org>): These open source projects use the geo microformat for publishing location geodata.

## adr

The adr microformat enables publishers to mark up address information in a standard, interoperable way. The adr microformat is a one-to-one representation of the adr property in vCard, and thus a subset of hCard.

- **More information:** <http://microformats.org/wiki/adr>
- **Status:** Draft (see [http://microformats.org/wiki/adr#Draft\\_Specification](http://microformats.org/wiki/adr#Draft_Specification) for the latest draft specification)
- **Author:** Tantek Çelik (<http://tantek.com>)

## Usage

adr elements have a root element with a class of adr and one or more of the following subproperties (i.e., child elements with classes equal to the following values):

- **type:** One of work, home, pref, postal, dom, or intl (these values are taken directly from the vCard specification)
- post-office-box
- extended-address
- street-address
- locality
- region

- postal-code
- country-name

### Example

Here's an example of adr in use:

```
<div class="adr">
  <div class="street-address">1600 Pennsylvania Ave</div>
  <span class="locality">Washington</span>,
  <span class="region">DC</span>,
  <span class="postal-code">20520</span>
  <span class="country-name">USA</span>
</div>
```

### Tools

Because adr is a subset of hCard, the tools that help publish hCard content also help publish adr content. See the “hCard” section of this appendix for more details.

### Services

Because adr is a subset of hCard, the services that use hCard content also use adr content. See the “hCard” section of this appendix for more details.

## hCard

hCard is a one-to-one representation of the properties and values of vCard (IETF RFC 2426) in HTML. hCard extends the limited scope of vCard—essentially people—to include companies, organizations, and places.

- **More information:** <http://microformats.org/wiki/hcard>
- **Status:** Specification (see <http://microformats.org/wiki/hcard#Specification> for the latest specification)
- **Authors:** Tantek Çelik (<http://tantek.com>) and Brian Suda (<http://suda.co.uk>)

### Usage

hCard allows publishers to mark up details (particularly, but not exclusively, contact details) for people, organizations, companies, and places.

hCards have a root element with a class value of vcard and a required subproperty of fn (formatted name), for the name of the entity represented by the hCard. Where an hCard is for an organization, the organization fn also includes a class value of org to specify this.

Optional subproperties include the following:

- Name and personal details
  - N (name): This is for the structured name of the entity, and it must take one or more subproperties from the following list: honorific-prefix, given-name, additional-name, family-name, honorific-suffix.
  - nickname: A nickname.
  - bday: Birthday. This subproperty typically uses the abbr design pattern to have both human-friendly and machine-readable values (see the “The abbr design pattern” section in Appendix B for more information).
  - photo: An image element (<img>) with a class of photo.
  - url: A URL for the person or organization.
- Contact information
  - adr (address): See the “adr” section of this appendix. The adr microformat is a subproperty of hCard.
  - geo: A longitude and latitude for the person, place, or organization represented by the hCard. See the “geo” section of this appendix for details. geo is a subproperty of hCard.
  - email: A contact email address. It can take a type and value, or just a value. The type may be one of the following: INTERNET, x400, pref, or “other IANA registered address types.”
  - tel: A contact telephone number. This subproperty may have multiple numbers, and it can take a type and value, or just a value. The type may be one of the following: voice, home, msg, work, pref, fax, cell, video, pager, bbs, modem, car, isdn, or pcs.
- Organizational details
  - org: Where the hCard is for a person, this subproperty is for an organization the person works at, or has some other role with. Where the hCard is for an organization, this is the name of the organization. In this case, the element with a class value of fn *must* also have a class value of org.
  - role: The role a person plays within the organization specified by the org property.
  - title: The title a person has at the organization specified by the org property.
  - logo: An image element (<img>) with a class value of logo specifies a graphic image for the entity represented by the hCard. For a company, it will typically be the company logo. For an individual, it may be an avatar. See also the photo property, for photographic images of people.
- Other properties

Because hCard is basically vCard represented in HTML, it acquires all the vCard properties. Some of these are not commonly used (even by applications that support vCard), and so are probably going to be rarely used in hCard as well. They are included here for completeness. The descriptions are largely taken from the vCard specification.



- **agent**: Specifies information about another person who will act on behalf of the entity represented by the hCard.
- **categories**: In effect, these are tags for the hCard, implemented using the rel-tag microformat.
- **key**: Specifies the public key or authentication certificate associated with the entity the hCard represents.
- **class**: Specifies the access classification for the hCard (e.g., public, private, or confidential).
- **label**: Specifies the formatted text corresponding to delivery address of the entity represented by the hCard.
- **mailer**: Specifies the type of electronic mail software that is used by the entity represented by the hCard.
- **note**: Specifies supplemental information or a comment that is associated with the hCard.
- **rev**: A date/time value specifying revision information about the hCard.
- **sound**: Specifies digital sound content information that annotates some aspect of the hCard. By default, this type is used to specify the proper pronunciation of the name type value of the hCard.
- **tz**: Specifies information related to the time zone of the entity represented by the hCard.

## Examples

First up, here's an hCard for a person:

```
<div class="vcard">
  <span class="fn n"><span class="given-name">William</span> <span class="additional-name">Henry</span> <span class="family-name">
  Gates</span> <span class="honorific-suffix">III</span></span>
  (born <abbr class="bday" title="1955-10-28">October 28, 1955
  </abbr>)
  <div class="adr">
    <p><span class="street-address">1835 73rd Ave NE</span>,</p>
    <p><span class="locality">Medina</span>,</p>
    <p><span class="region">WA</span> <span class="postal-code">98039
    </span></p>
    <p class="country-name">USA</p>
  </div>
  <p><span class="title">Chairman</span>,<span class="org">Microsoft
  </span></p>
  <p><a href="http://www.microsoft.com/billgates/"
  class="url">http://www.microsoft.com/billgates/</a></p>
  <p><a href="mailto:bill@microsoft.com" class="email">
  bill@microsoft.com</a></p>
</div>
```

Here's an hCard for an organization:

```
<div class="vcard">
  <p class="fn org">Web Directions Conference Pty Ltd
  <a href="http://suda.co.uk/projects/X2V/get-vcard.php?uri=
    http://microformatique.com/book/chapter7/index.html">
  </a></p>

  <div class="adr">
    <p class="street-address">8/54 Mitchell St</p>
    <p><span class="locality">Bondi</span> <span class="region">NSW
      </span> <span class="postal-code">2026</span></p>
    <p class="country-name">Australia</p>
  </div>
  <p>Phone/Fax: <span class="tel"><span class="type">Work</span>:
  <span class="value">61 2 9365 5007</span></p>
  <p>Email: <a class="email" href="mailto:info@webdirections.org">
    info@webdirections.org</a></p>
</div>
```

## Tools

A number of tools can help you create hCard content:

- **Dreamweaver Microformats Extension** ([www.webstandards.org/action/dwtf/microformats](http://www.webstandards.org/action/dwtf/microformats)): The Dreamweaver microformats toolbar from the Web Standards Project, by Drew McLellan, supports creating hCard.
- **hCard Creator** (<http://microformats.org/code/hcard/creator>): From microformats.org, this tool generates the HTML for you, based on your form input.
- **Textpattern Microformats Plug-in** ([http://placenamehere.com/TXP/pnh\\_mf](http://placenamehere.com/TXP/pnh_mf)): Created by Chris Casciano, the pnh\_mf plug-in provides a simple way to mark up microformatted content, including hCard.

## Services

The following services use hCard content:

- **Technorati Microformats search**: <http://kitchen.technorati.com>
- **Pingerati**: <http://pingerati.net>
- **X2V**: <http://suda.co.uk/projects/X2V>

## hCalendar

hCalendar is a one-to-one representation of the properties and values of iCalendar (IETF RFC 2445) in HTML. iCalendar is an interoperable format for calendaring and scheduling applications.

- **More information:** <http://microformats.org/wiki/hcalendar>
- **Status:** Specification (see <http://microformats.org/wiki/hcalendar#Specification> for the latest specification)
- **Authors:** Tantek Çelik (<http://tantek.com>) and Brian Suda (<http://suda.co.uk>)

## Usage

iCalendar is a reasonably complex standard that is often not entirely implemented by supporting applications. As such, there is a current effort to create a simplified subset of iCalendar, called iCalendar Basic. Aspects of iCalendar, while theoretically part of hCalendar, have yet to be implemented in HTML. As such, here we'll only consider the fully implemented aspects of hCalendar.

According to the hCalendar specification, the basic model of hCalendar is to model “iCalendar object/property names in lower-case for class names, and to map the nesting of iCalendar objects directly into nested XHTML.” hCalendar allows publishers to mark up events and other time/date based episodes (such as educational experience and work experience).

hCalendars have an optional `vcalendar` root element. Where only a single calendar of events occurs on a page, where no root element is found, the page itself is assumed to be the root element. Where more than one calendar of events is found on a single page, the root element is any HTML element with a class value of `vcalendar`.

An hCalendar consists of one or more events. These are marked up in hCalendar as descendent elements of the root element, with a class value of `vevent`.

### Required properties of events

At a minimum, a `vevent` requires a `summary` and a `dtstart` to appear as children of it:

- `summary`: This is typically the name of the event.
- `dtstart`: This is the event's start date, an ISO 8601-formatted date/time for the commencement of the event. It is recommended that the `abbr` design pattern be used to give both a human-readable version and machine-readable version of the date, for example:

```
<abbr title="20070207" class="dtstart">Feb. 6th 2007</abbr>
```

### Common optional properties of events

The following are optional properties of events:

- **description:** This property provides a “more complete summary” (according to the iCalendar specification) of the event. An event may only have one description subproperty, so where, for instance, the description spans multiple paragraphs, a single div with class `description` should contain these paragraphs.
- **dtend:** This is the event’s end date (i.e., the date/time at which the event ends), in ISO 8601 format. A date without a time is in effect midnight of the day before the date, so for timeless dates you should use the date *following* the event date to be correct. It is recommended that the abbr design pattern be used to give both a human-readable version and machine-readable version of the date, for example:

```
<abbr title="20070206" class="dtstart">Feb. 6th 2007</abbr>
```

*Due to a quirk of hCalendar, a DTEND date without a time value is midnight between the day before and the date specified—in effect, the day before—hence the mismatch between the title value and the human-readable version.*

If a date and time value were used, the date part would be 20070206.

- **location:** The location of the event. The `adr`, `geo`, or hCard microformat can be used to mark these up in a more information-rich manner.
- **url:** A URL for the event (e.g., for more information).

## Example

Note that the following is an example of an hCalendar for a single event. The containing vCalendar element would typically be optional (unless there were more than one distinct calendar on this page).

```
<div class="vevent">
  <h3 class="summary"><a href="http://north.webdirections.org"
    class="url">Web Directions North</a></h3>

  <div class="description">
    <p>A Web design and development conference in Vancouver
      Canada.</p>
    <p>Featuring a who's who of international web experts...</p>
  </div>
  <p>From <abbr title="20070206" class="dtstart">Tuesday,
    Feb. 6th </abbr> to <abbr title="20070209" class="dtend">
    Thursday, Feb. 8th </abbr> 2007</p>
  <p class="location">Renaissance Vancouver Hotel Harbourside</p>
</div>
```

## Tools

Some tools you can use to create hCalendar content are as follows:

- **Textpattern Microformats Plug-in** ([http://placenamehere.com/TXP/pnh\\_mf](http://placenamehere.com/TXP/pnh_mf)): The pnh\_mf plug-in for Textpattern helps publish hCard-formatted data and also supports publishing hCalendar content.
- **Dreamweaver Microformats Extension** ([www.webstandards.org/action/dwtf/microformats](http://www.webstandards.org/action/dwtf/microformats)): This Dreamweaver microformats extension by Drew McLellan enables easy, form-driven hCalendar creation in Dreamweaver.
- **hCalendar Creator** (<http://microformats.org/code/hcalendar/creator>): This tool provides an easy-to-use, form-driven interface for creating hCalendar content.
- **xfy** (<https://www.xfytec.com>) and **LiveWriter** (<http://ideas.live.com/program-page.aspx?versionId=4372c8c2-b76f-4d44-aea1-9835b61d8dc1>): These recently released desktop-based blogging tools both have hCalendar support. Microsoft LiveWriter is a Windows-only product, while xfy is written in Java and runs on Windows, Mac OS X, and Linux.

## Services

These services use hCalendar:

- **Technorati Microformats search** (<http://kitchen.technorati.com>): This search engine aggregates events marked up using hCalendar. You should also ping Pingerati (<http://pingerati.net>) so that Technorati and other hCalendar aggregators know about your new or updated hCalendar content.
- **X2V** (<http://suda.co.uk/projects/X2V>): This service by Brian Suda converts hCalendar content to the iCalendar format, enabling applications that support iCalendar to import hCalendar content. It also enables applications such as Thunderbird and iCal, which support subscribing to iCalendar content to subscribe to hCalendar marked-up calendars on web pages.
- **Life Lint parser** ([www.lifelint.net](http://www.lifelint.net)): Life Lint is similar to X2V in that it converts hCalendar to iCalendar (and RDF), and it has optional Outlook 2002 compatibility (although bear in mind that Outlook's support for iCalendar format is not perfect).

## hReview

hReview is a simple format for marking up reviews on the Web. The schema of hReview was developed as a minimal subset of a broad range of review formats currently being published by the likes of Amazon, Epinions, Zagat, iTunes, and many more, as well as earlier attempts at common review formats, such as RVW, RDF Review Vocabulary, and Simple-Review XML.

- **More information:** <http://microformats.org/wiki/hreview>
- **Status:** Draft (see [http://microformats.org/wiki/hreview#Microformats\\_Draft\\_Specification\\_2006-02-22](http://microformats.org/wiki/hreview#Microformats_Draft_Specification_2006-02-22) for the latest draft specification)
- **Authors:** Tantek Çelik (<http://tantek.com>), Ali Diab (<http://360.yahoo.com/alidiabali>), Ian McAllister (<http://spaces.msn.com/members/ianmcallister>), John Panzer (<http://journals.aol.com/panzerjohn/abstractioneer>), Adam Rifkin (<http://ifindkarma.com/blog>), and Michael Sippey (<http://sippey.typepad.com>)

## Usage

hReview is designed for publishing structured reviews of people, businesses, films, products, websites—indeed anything that can be reviewed online. It is used as follows.

- **root:** An hReview has a root element, which can be any HTML element, with a class value of hreview.

## Required properties

hReview has one required property, `item`, with the following further requirements:

- When the review is for a person, the `item` property must be an hCard.
- When the review is of an event, the `item` property should be an hCalendar.
- Any item must have at least a nested `fn` subproperty, for the name of the item. Where the item is an hCard or hCalendar, this `fn` property will be automatically part of the nested microformat.
- Where an item has a photo or URL, subproperties of `item` with class values of `url` and `photo` should be used to mark these up.

## Optional properties

An hReview may also have the following optional properties:

- **summary:** The title of the review itself.
- **type:** The type of item being reviewed, which is one of the following list: `product`, `business`, `event`, `person`, `place`, `website`, or `url`.
- **reviewer:** The author of the review. This is must be an hCard. Where the reviewer is anonymous, the value of the hCard `fn` is `anonymous`. For example, the following is the markup for a review by Francine Smith:

```
<span class="reviewer vcard"><span class="fn">Francine Smith ↪
</span></span>
```

An anonymous review would be marked up as follows:

```
<span class="reviewer vcard"><span class="fn">anonymous ↪
</span></span>
```

- **dtreviewed**: The date on which the review was written or published, in ISO 8601 date/time format. Using the abbr design pattern for providing both human- and machine-readable dates and times is recommended. For example, a review written on May 17, 2007, in Sydney, Australia, might be marked up as follows:

Written `<abbr class="dtreviewed" title="20070517">` ➤  
 May 17, 2007`</abbr>`

- **description**: The full text of the review.
- **rating**: The reviewer's rating for the item reviewed. By default, this is a fixed-point integer to one decimal point from 1.0 to 5.0 (e.g., 2.5 is a valid rating value, but 4.99 is not). Optionally, integer values from a specified "best" to "worst" may be used. For example, a standard score of 4.5 out of 5 may be marked up like this:

I gave the film `<span class="rating">4.5</span>` out of 5

To give a nondefault rating, you could use something like this:

full marks, `<span class="rating">` `<span class="value">10</span>` ➤  
 out of `<span class="best">10</span>``</span>`

- **version**: The version of hReview used for marking up the review. The current version is 0.3, while this and all future versions will be backward compatible with version 0.2.
- **tags**: A list of rel-tag tags describing the review. No particular class value is required for the tag links to indicate they are tags; the rel="tag" value is sufficient for this.
- **license**: To indicate a license for the review, a rel-license link may be added to the review. In this instance, the license applies to the review, not the page containing the review.
- **permalink**: A link with rel values of bookmark and self indicates the permanent URL for the review.

## Example

This example demonstrates the core features of hReview:

```
<div class="hreview">
  <h2 class="summary">Pirates of the Caribbean: Dead Man's Chest ➤
  - Disappointing at best</h2>
  <p class="item"> ➤
    <a href="http://disney.go.com/disneypictures/pirates/" ➤
      class="fn url">Pirates of the Caribbean 2 - Dead Man's Chest</a>
  </p>
  <p class="reviewer vcard"><span class="fn">John Allsopp</span></p>
  <p>Date reviewed <abbr title="20061118" class="dtreviewed"> ➤
    November 18 2006</abbr>.</p>
  <p>Rating: <span class="rating"><span class="value">4</span> ➤
    out of <span class="best">10</span></p>
```

```

<div class="description">
  <p>After the success and excitement of the original Pirates ↵
    of the Caribbean...</p>
  <p>The characters lack any of the real freshness and ↵
    cleverness of the original, even <a ↵
      href="http://en.wikipedia.org/wiki/Johnny_Depp" rel="tag"> ↵
      Depp's </a> Jack Sparrow has become more...</p>
  <p>Disappointing, to say the least.</p>
</div>
<p><a href="http://creativecommons.org/licenses/by-sa/2.5/" ↵
  rel="license">Licensed under a creative commons attribution ↵
  share alike 2.5 license</a></p>
<p>Review type: <span class="type">product</span></p>
<p>hReview version<span class="version">0.3</span><p>
</div>

```

## Tools

Some tools you can use to create hReview content are as follows:

- **hReview Creator** (<http://microformats.org/code/hreview/creator>): Like microformat.org's other microformat creators, this is a form-driven interface that produces microformatted HTML for your review.
- **Textpattern Microformats Plug-in** ([http://placenamehere.com/TXP/pnh\\_mf](http://placenamehere.com/TXP/pnh_mf)): For Textpattern users, Chris Casciano's pnh\_mf supports publishing in the hReview format as well as a number of other microformats.
- **hReview WordPress Plug-in** ([www.aes.id.au/?page\\_id=28](http://www.aes.id.au/?page_id=28)): Andrew Scott has created this form-driven hReview plug-in, which allows for easy publishing of individual or multiple reviews on a single page or in a single post.
- **WP Movie Ratings** (<http://paulgoscicki.com/archives/2006/11/wp-movie-ratings-v14-released>): This plug-in from Paul Goscicki integrates with IMDb to allow pain-free reviewing of movies in hReview format.

## Services

Services using hReview content include the following:

- **Revoov** (<http://revoov.com>): Revoov is a review aggregation and publishing service, which both aggregates reviews published on the Web in hReview format and publishes reviews in hReview format.
- **Cork'd** (<http://corkd.com>): This wine community site features reviews created using a form-driven interface and published using hReview.
- **Yahoo Tech** (<http://tech.yahoo.com>): This technology review site also uses hReview to publish technology reviews from users.



## hResume

hResume is a format for semantically marking up resumes or curriculum vitae (CVs) for publishing on the Web, typically in HTML. The set of properties of hResume (its schema) was developed through research of existing publishing schemas for online resumes and resume formats such as Europass and HR-XML.

- **More information:** <http://microformats.org/wiki/hresume>
- **Status:** Draft (see <http://microformats.org/wiki/hresume#Status> for the latest draft specification)
- **Author:** Ryan King (<http://theyanking.com>)

## Usage

hResume may be used by individuals to publish their resumes or by aggregators to publish resumes or CVs aggregated from the Web or otherwise obtained.

The root of an hResume is any HTML element with a class value of hresume. Its required property and optional properties are as follows.

### Required property

hResume has a single required property, contact, which provides the name and contact details for the person whose resume this is. contact must be an hCard.

### Optional properties

An hResume may have the following properties (the heading for each property is the class value of that property when marking up in HTML):

- **summary:** This property is for an “overview of qualifications and objectives” (per the hResume specification).
- **education:** This property is an hCalendar, the individual events of which are each an education event (e.g., secondary schooling, undergraduate degree, etc.). Each event has a class value of education.
- **experience:** This property is an hCalendar, the individual events of which are each a work event (e.g., jobs). Each event has a class value of experience. hCard should be used for job titles or roles. In this case, to prevent tedious repetition, the include design pattern can be used to include the fn from the contact hCard into all such hCards.
- **skill:** The person’s skills can be listed as a rel-tag link for each skill, with the addition of the class value of skill on each such link.

- **affiliation:** Organizations with which the person has an affiliation can be marked up with hCards, with the additional class value of `affiliation` on the root element for each organization's hCard.
- **publications:** Articles, books, and other publications by the person can be added to an hResume simply by using the `<cite>` element. Where a `<cite>` element appears inside an hResume, the cited work is taken to have been authored by the person whose resume this is.

## Example

This is my (somewhat truncated) resume, shown here as an example, using all the preceding features from hResume. The class values specific to hResume are emphasized.

```
<div class="hresume">
  <h1>Resume: <address class="vcard contact"><a ↵
    href="mailto:john@webdirections.org" class="fn email"> ↵
    John Allsopp</a></address></h1>
  <p class="summary">Software developer, conference organizer, ↵
    speaker, and author John Allsopp has spent the last 15 years ↵
    running and developing software for Western Civilisation ↵
    Pty. Ltd., writing and publishing training courses, speaking ↵
    at conferences, organizing conferences, and running training ↵
    seminars in standards-based web development.</p>
  <h3>Education</h3>

  <ol class="vcalendar">
    <li class="vevent education"><abbr class="dtstart" ↵
      title="19850101">1985</abbr>-<abbr class="dtend" ↵
      title="19891231">1989</abbr> <abbr title="Bachelor of ↵
      Science "> B.SC</abbr> <span class="vcard"><a class="fn ↵
      org url"href="http://www.usyd.edu.au/">University of ↵
      Sydney, Australia </a></span> <span class="summary">Majoring ↵
      in Computer Science and Mathematics. Also studied Law, ↵
      English Literature, and Philosophy.</span></li>
    <li class="vevent education"><abbr class="dtstart" ↵
      title="19790101">1979</abbr>-<abbr class="dtend" ↵
      title="19841231"> 1984</abbr> <abbr title="Higher School ↵
      Certificate">H.S.C</a><span class="vcard"><a class= ↵
      "fn org url" href="http://www.riverview.nsw.edu.au">Saint ↵
      Ignatius College, Riverview</a></span></li>
  </ol>
```

```

<h3>Experience</h3>
<ol class="vcalendar">
  <li class="vevent experience"><abbr title="19940101"
    class="dtstart">1994</abbr> - present. Chief Technology
    Officer, <span class="vcard"><a class="fn org url"
    href="http://westciv.com">Western Civilisation Pty. Ltd.</a>
    </span> <span class="summary">Responsible for the development
    of Software, including Palimpsest, Style Master, and Layout
    Master</span></li>
  <li class="vevent experience"><abbr title="20060101"
    class="dtstart">2006</abbr>- present Director, <span
    class="vcard"><a class="fn org url"
    href="http://webdirections.org">Web Directions Conferences
    Pty. Ltd.</a></span> <span class="summary">Responsible
    for a broad range of logistical and management aspects of
    organizing the Web Directions conferences.</span></li>
  <li class="vevent experience"><abbr title="20030101"
    class="dtstart">2003</abbr> - <abbr title="20051231"
    class="dtstart">2005</abbr> Director, <span class="vcard">
    <a class="fn org url" href="http://webessentials.com">WE04 and
    WE05 conferences</a></span> <span class="summary">Responsible
    for a broad range of logistical and management aspects of
    organizing the WE04 and WE05 conferences.</span></li>
</ol>
<h3>Skills</h3>

<ul>
  <li><a class="skill"
    href="http://en.wikipedia.org/wiki/Software_engineering"
    rel="tag">Software Engineering</a></li>
  <li><a class="skill"
    href="http://en.wikipedia.org/wiki/Programming"
    rel="tag">programming</a></li>
  <li><a class="skill"
    href="http://en.wikipedia.org/wiki/Web_development"
    rel="tag">web development</a></li>
  ...
</ul>

<h3>Affiliations</h3>

```

```

<ul>
  <li class="vcard affiliation"> ↵
    <a href="http://www.webstandards.org/" class="fn org url"> ↵
      Web Standards Project</a></li>
  <li class="vcard affiliation"> ↵
    <a href="http://www.northbondisurfclub.com/" class="
      "fn org url"> North Bondi SLSC</a></li>
</ul>

<h3>Publications</h3>

<ol>
  <li><cite href="http://www.alistapart.com/articles/dao/"> ↵
    <a href="http://www.alistapart.com/articles/dao/">A Dao of Web ↵
      Design</a></cite> A List Apart, April 2000</li>
  <li><cite ↵
    href="http://www.westciv.com/style_master/ ↵
    house/good_oil/not_paper/"> ↵
    <a href="http://www.westciv.com/style_master/ ↵
    house/good_oil/not_paper/"> ↵
    Web Pages aren't printed on paper</a></cite>September 1999</li>
</ol>

</div>

```

## Tools

Some tools you can use to create hResume content are as follows:

- **hResume Creator** (<http://hresume.weblogswork.com/hresumecreator>): This form-based resume creator from Spur, Inc. is much like the other microformats creators, such as hCard Creator and hCalendar Creator.
- **hResume WordPress Plug-in** ([http://hresume.weblogswork.com/?page\\_id=3](http://hresume.weblogswork.com/?page_id=3)): Another tool from Spur, Inc., this plug-in for WordPress helps publish hResumes on WordPress-based blogs.
- **Microformat Resume Plug-in for WordPress** ([www.ssdesigninteractive.com](http://www.ssdesigninteractive.com)): Sajid Sayid created this easy-to-use plug-in for adding hResume content to a WordPress-based site.

## Services

Here are some services using hResume:

- **Emurse** (<http://emurse.com>): This service for building, hosting, and publishing resumes online uses the hResume format.
- **LinkedIn** ([www.linkedin.com](http://www.linkedin.com)): This professional online networking service uses hResume for publishing 9 million resume details.

## hAtom

hAtom is a microformat based on a subset of Atom, the XML syndication format for web feeds. hAtom specifically focuses on the aspect of Atom concerned with blog posts. Despite being a draft specification, because it is based on the nearly 1.0 standard Atom, hAtom's schema is quite stable.

- **More information:** <http://microformats.org/wiki/hatom>
- **Status:** Draft (0.1; see [http://microformats.org/wiki/hatom#Draft\\_Specification](http://microformats.org/wiki/hatom#Draft_Specification) for the latest draft specification)
- **Authors:** David Janes (<http://blogmatrix.blogmatrix.com>), with contributions by Tantek Çelik (<http://tantek.com>) and Benjamin Carlyle (<http://members.optusnet.com.au/benjamin Carlyle/benjamin/blog>)

## Usage

hAtom is used to mark up feeds in HTML, rather than in separate files, such as RSS or Atom. Rather than replacing RSS or Atom, hAtom's aim is to use the schema of Atom to create a standardized format for publishing blogs and other feed-based content in HTML. Because Atom and RSS feeds typically list only recent posts, hAtom is particularly beneficial for archiving posts.

### Schema

hAtom feeds have an optional root element of hfeed. It can be any HTML element, as long as it has a class value of hfeed. Where there is no feed element, the page itself is assumed to be the feed (in the same way that the hCalendar element is optional).

A feed consists of individual posts. A post element is any HTML element with a class value of hentry.

## Required properties

An entry must have the following properties (the heading for each property is the class value of that property when marking up in HTML):

- **entry-title**: The title of a particular entry (usually a heading of some level).
- **author**: The author of the entry. This must be marked up using an hCard, and it should, where possible, use an <address> element. Because the <address> element is an inline element, this may cause problems for complex contact details (e.g., postal addresses), but with hAtom, typically contact details are the URL of a form-based contact page or email address, so this is less of a potential problem than the suggested use of <address> in hResume.
- **updated**: The date the entry was last updated. Use the datetime design pattern to publish both human- and machine-readable versions.

## Optional properties

An entry may have the following properties (the heading for each property is the class value of that property when marking up in HTML):

- **entry-summary**: A short summary or extract from the entry.
- **entry-content**: The full content of the post (an entry may have more than one entry-content).
- **published**: The date the entry was published. Use the datetime design pattern to publish both human- and machine-readable versions. A single element can be both the published and updated dates like so:
 

```
<abbr title="20070101" class="updated published"> ➤  
1st January 2007</abbr>
```
- **permalink**: A link with a rel value of bookmark is the permalink for an entry. No specific class value is required for the link.
- **tag**: Both a feed and individual entries can have one or more “categories.” Both feed and entry categories are defined by the use of a link with a rel value of tag. Feed categories appear inside a feed root element, but outside any particular hEntry elements. Where a link with a rel value of tag appears inside an hEntry element, it defines a category for the entry. No specific class value is required for the link.

## Example

This hFeed example demonstrates all the features of hAtom. The hAtom class values are emphasized.

```

<div class="hfeed">
<!-- these are the categories for the feed because they appear ↪
  inside the hfeed element but outside any specific hEntry element -->
<ul>
<li><a href="http://www.technorati.com/tag/microformats" rel="tag" ↪
  microformats</li>
<li><a href="http://www.technorati.com/tag/semantics" rel="tag" ↪
  semantics </li>
<li><a href="http://www.technorati.com/tag/HTML" rel="tag">HTML</li>
<li><a href="http://www.technorati.com/tag/patterns" rel="tag" ↪
  patterns</li>
</ul>
<!-- there may be many hentries here before we get to the entry we ↪
  have been marking up -->
<div class="hentry">
<h3 class="entry-title">Introducing Web Connections</h3>
<p>Published:<abbr class="updated published" title="20060920" ↪
  September 20, 2006</abbr></p>
<div class="content">
<p><span, class="summary">Super smart, far too young and good ↪
  looking, <a href="http://toolmantim.com/">Tim Lucas</a> and ↪
  Cam ("<a href="http://www.themaninblue.com/">the man in ↪
  blue</a>") Adams, have just taken the wraps off a fantastic ↪
  web app, "~<a href="http://connections.webdirections.org">Web ↪
  Connections </a>"</span> they've built to help networking at ↪
  our upcoming (next week!) conference, ↪
  <a href="http://www.webdirections.org/"> Web Directions</a>.</p>
<p>Tim and Cam have incorporated reltag, XFN, hCard, and ↪
  hCalendar. <strong>Seriously microformatted</strong>.</p>
  ...
<p>Categories</p>
<!-- these are the categories for the entry because they appear ↪
  inside the hEntry element -->
<ul>
<li><a href="http://www.technorati.com/tag/microformats" ↪
  rel="tag" ↪
  rel="tag"> microformats</li>
<li><a href="http://www.technorati.com/tag/webapps" ↪
  rel="tag" ↪
  rel="tag">webapps</li>
<li><a href="http://www.technorati.com/tag/WD06" ↪
  rel="tag" ↪
  rel="tag"> WD06 </li>
</ul>
</div> <!-- end hEntry -->
<!-- there may be many more entries after this one in the feed -->
</div> <!-- end hFeed -->

```

## Tools

Some tools you can use to create hAtom content are as follows:

- **hAtom Creator** (<http://dichotomize.com/uf/hatom/creator.html>): This is a form-based creator for individual hAtom entries.
- **WordPress Sandbox theme** (<http://www.plaintxt.org/themes/sandbox>): This “skeleton” theme for WordPress (which you can use to develop your own themes on top of), by Scott Allan Wallick and Andy Skelton, supports hAtom and is rumored to be the basis for the next default WordPress theme.
- **WordPress loop with hAtom support** ([www.fberriman.com/?p=86](http://www.fberriman.com/?p=86)): If you are interested in adding hAtom to your existing WordPress theme or install, use this WordPress loop by Frances Berriman to add hAtom to your blog.

## Services

The following services use hAtom:

- **hAtom2Atom** (<http://rbach.priv.at/hAtom2Atom>): This XSLT processor is for transforming hAtom marked-up content to Atom.
- **Tails** (<http://blog.codeeg.com/tails-firefox-extension-03>): The venerable Firefox extension supports hAtom.



# APPENDIX B MICROFORMAT DESIGN PATTERNS



Microformat design patterns are not themselves microformats; rather, they formalize commonly reused pieces of code that are generally useful in developing new microformats (and may also be useful in developing semantic HTML other than just in relation to microformats).

This appendix covers the design patterns you'll encounter most frequently in your work with microformats.

### The abbr design pattern

The abbr design pattern uses the `<abbr>` (abbreviation) element of HTML to provide a mechanism for marking up both human-friendly and machine-readable values for the same element. This is important to meet the microformats principles of being “designed for humans first, machines second” and to “enable and encourage decentralized development, content, services.”

#### Usage

The human-readable value is the content of the `<abbr>` element, while the machine-readable part is the value of the `title` attribute of the `<abbr>` element. The pattern is commonly used with dates and times (see also the “The datetime design pattern” section), but also with a specific form of the geo microformat. It can also be used where both a human- and machine-readable version of the marked-up information is required.

Some common uses are as follows:

- **Date/time:** The pattern is used to specify a human-readable date/time and an unambiguous machine-readable date/time value (using ISO 8601, as recommended by the W3C for dates on the Web).
- **Geodata:** The pattern is used to specify a machine-readable decimal longitude and latitude value, and a human-readable location.

#### Examples

Here's an example for date/time:

```
<abbr title="20061028T17:53:17+09:00">5:53pm, 28th October 2006, in ➤  
  Sydney Australia</abbr>
```

Here's an example for geodata (latitude precedes longitude):

```
<abbr title="13.150000;169.150000" class="geo">400 miles (640 km) ➤  
  South West of Wake Island, in the North Pacific Ocean</abbr>
```

## The class design pattern

The class design pattern is a fundamental microformat design pattern. It is used to include the semantics of a schema, either existing, like vCard, or novel, like hReview, into HTML, by the use of specific `class` attribute values on HTML elements.

### Usage

For more complex microformats, there is a vocabulary of possible `class` values. These `class` values are added to HTML elements to give the semantics of the particular microformat to that element. Microformats never mandate the use of particular HTML element with this design pattern (e.g., requiring a `div` or `heading` with a specific `class` value), but they do at times recommend elements—such as the `<address>` element for the contact property of an `hResume`.

To use the class design pattern, simply add a specific `class` value from the schema for the microformat to an HTML element. Multiple `class` values are permitted and are not uncommon.

There are some restrictions on the use of `class` values. `class` values for the subproperties of a microformat must be on descendent elements of the root element, not on the root element itself. However, a root element may take multiple classes, where the microformat is nested inside another microformat. In this case, it takes both the `class` value for the root element and the `class` value for the property it represents in the microformat it is nested inside.

### Examples

A simple use of the class design pattern is to add the semantic value `country-name` to an element:

```
<span class="country-name">Australia</span>
```

In the next example, the value `vcard` is the root for the `hCard` element, while `reviewer` is the `class` value for the role this element plays in the `hReview` it is part of. `hCard` requires an `fn` property, but this must be on a descendent element of the root element, not on the root element itself.

```
<span class="vcard reviewer"><span class="fn">Francine ➡  
Smith</span></span>
```

In some cases, multiple `class` values are required. In `hCard`, where the details are for an organization, the `fn` (formatted name) property must also have an `org` value for its `class`:

```
<span class="fn org">IBM</span>
```

## The datetime design pattern

The datetime design pattern is a specific use of the abbr design pattern. It enables marking updates in both human-friendly and unambiguous machine-readable formats, using the ISO 8601 date/time format. See the section “The abbr design pattern” for details and examples.

## The include pattern

The same microformatted information, with small changes, may appear a number of times on the same page. For example, in hResume, hCard is recommended for each experience property where a title or role is involved. Because fn is required in an hCard, this necessitates repeating the fn property each time the hCard for a specific person is used. Not only is this more verbose than necessary, but it also requires some mechanism for hiding the formatted names, as their appearance numerous times would appear incongruous.

The include pattern enables a property from one microformat (e.g., the fn from an hCard) to be included in other microformats on the same page (the include pattern does not allow including parts of a microformat on another page into a microformat—it is strictly intrapage).

## Usage

Because the initial include pattern proposal used the <object> element, which caused difficulties with Safari, the include pattern is, at the time of this writing, not entirely finalized.

The basic concept of the include pattern is to uniquely identify one microformat using an id attribute, and then include this microformat in other microformats by using a link to this uniquely identified element from within subsequent microformats. This has the effect of including or “importing” the properties of the included microformat into the other microformats.

## Examples

The first part of the include pattern is to add a unique identifier to a microformat using the id attribute:

```
<div class="hResume">
  <h1 class="summary">Resume: John Allsopp</h1>
  <div class="vcard contact" id="jafn">
    <a href="mailto:john@webdirections.org" ↪
      class="email fn">John Allsopp</a>
  </div>
</div>
```

Now you include this in a subsequent microformat on the same page, using a link of class `include` pointing to this element, as part of your microformat:

```
<li class="vevent experience"><abbr title="19940101" class="dtstart"> ➡
  1994</abbr> - present.
  <span class="vcard">
    <a href="#jafn" class="include"></a>
    <span class="title">Chief Technology Officer</span></span>
</li>
```

## The rel design pattern

Links often define the relationship between documents. Where that is the case, the `rel` attribute of HTML can be used to encode this relationship. This aspect of HTML is used extensively by microformats.

### Usage

The rel design pattern uses a keyword value for the `rel` attribute and a link to indicate the relationship between the document specified by the link `href` and the document containing the link. A number of microformats use a single `rel` value (`rel-license`, `rel-tag`, `rel-nofollow`), while XFN uses a small set of `rel` values.

### Example

`rel-tag` uses a `rel` value of `tag` to identify a link as a microformatted tag link:

```
<a href="http://wikipedia.org/wiki/Pareto_principle" rel="tag"> ➡
  Pareto Principle</a>
```

## Attribute conventions

While these are not strictly design patterns in themselves, I'll touch on a number of conventions regarding the use and priority of certain HTML attributes. Because attributes typically carry their own semantics (or, more relevantly, their own particular purpose), it's a convention in microformats that when a given attribute is most relevant, it should be used.

You can find examples of this throughout the book, but no particular mention has been made of the convention. For example, where a microformat (such as `hCard` or `hCalendar`) has a `url` property, rather than marking this up as follows:

```
<span class="url"><a href="http://north.webdirections.org">Company ➡
  Web Site</a></span>
```

## MICROFORMATS: EMPOWERING YOUR MARKUP FOR WEB 2.0

where the `url` value would actually be `Company Web Site`, we make use of the semantics and purpose of the `href` attribute and mark it up like this:

```
<a href="http://north.webdirections.org" class="url">Company  
  Web Site</a>
```

Now, why don't microformat parsers take `Company Web Site` as the value of the `url` property in this instance? Because parsers understand the convention, which says that the `href` attribute makes the most sense for the kind of data we expect to find for a `url` property.

Similarly, where a property is applied to an image or other replaced element (such as an object element), because these kinds of elements have no content, we can use the `alt` attribute where we'd typically use the content of the element. So, for example, instead of the following:

```
<span class="fn">Brian Suda</span>
```

if we wanted to have both a photo and `fn` combined, we could use the `alt` attribute on the `<img>` element, like so:

```

```

Because we know the value of `fn` is text, and because the `alt` attribute is alternate text for the image (in effect, the text equivalent), the convention is that we use the value of the `alt` attribute for the value of the `fn` property. Similarly, as the standard way of including image data on the Web is via the `src` attribute of an `<img>` element, the value of the `photo` property is the image at the `src` location.