

This is a **low resolution, black and white** version of the article you downloaded. To download the whole of Free Software Magazine in *high* resolution and color, please subscribe!

Subscriptions are free, and every subscriber receives our fantastic weekly newsletters — which are in fact fully edited articles about free software.

Please click here to subscribe:

<http://www.freesoftwaremagazine.com/subscribe>

Stylish XML

Part one: using CSS to make your XML documents look pretty

David Horton

Creating custom markup with XML is pretty easy to do, but making it look good is another feat entirely. Fortunately, a little knowledge of cascading style sheets can go a long way toward making XML easier on the eyes.

Basic XML

Before diving into the subject of styling XML, you need to know a little about what an XML document is and how to create a very simple one using a text editor. Issue one of Free Software Magazine has an excellent article (http://www.freesoftwaremagazine.com/free_issues/issue_01/focus_format_xml) by Kay Ethier and Scott Abel introducing many of the terms and technologies surrounding XML. There are also plenty of tutorials available on the web showing you how to create your own markup with XML. Once you feel comfortable with the basics of XML, read on and let's start styling. Feel free to try experimenting with the CSS provided in the download area with this article.

Unstyled XML

A popular theme of many beginning XML tutorials is to take a recipe for some kind of delicious food and describe it using XML. Since I like to cook I'm going to stick with this theme and show you examples using an XML version

Paper-based recipe converted to XML

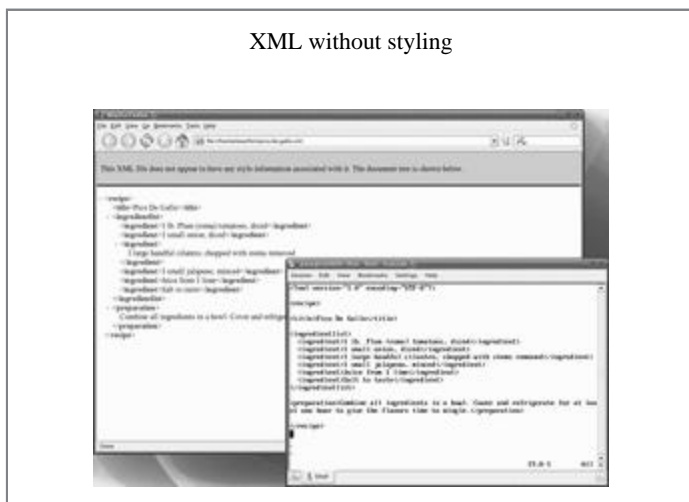


of my *muy delicioso* pico de gallo recipe. So grab some tortilla chips and margaritas and let's get started.

I'll start with a handwritten recipe for pico de gallo. In the past, good old pen and paper is how I would record and share all of my recipes. Then one day, when I was feeling particularly ambitious, I decided to start storing my recipes electronically as XML documents. It wasn't particularly difficult, I just typed the recipes up in my favorite text editor and thought up some tags to describe the different elements. Working with XML is a lot like using HTML except that there are no predefined tags. And, since the recipes are now stored as an electronic document, I can also share them on the internet.

One of the cool features of modern web browsers is that they can now handle XML documents just as easily as

XML without styling



the traditional HTML. In fact, I can even open up my pico de gallo recipe using Mozilla Firefox (<http://www.getfirefox.com/>) which makes sharing XML documents over the internet very easy. No more passing around handwritten copies, instead I can just refer people to my web site.

Unfortunately, the XML recipe displayed in the browser looks just as unattractive as it does in the text editor. Can you imagine trying to share this kind of recipe with family and friends? I can hear the conversation now, “Yeah, Mom, just ignore the stuff between the angle brackets... Right, the ones that look like greater-than and less-than... Because it’s XML and it’s cool, that’s why...”

Styling with CSS

The simplest answer to improving the look of XML in the Firefox browser is to use Cascading Style Sheets (CSS). This is the same CSS that is used for web pages and can be a very attractive choice if you fancy yourself as somewhat of an HTML guru. All I have to do to jazz up my pico de gallo recipe is to link it to a CSS file.

XML uses the `xml-stylesheet` processing instruction to reference style sheets. In the specific case of my pico de gallo recipe, I would place the line `<?xml-stylesheet href="recipe-style.css" type="text/css"?>` near the top of the XML file. For you web gurus out there this probably conjures up images of the HTML syntax `<LINK rel="recipe-style.css" type="text/css">` which is not surprising since it works in much the same way.

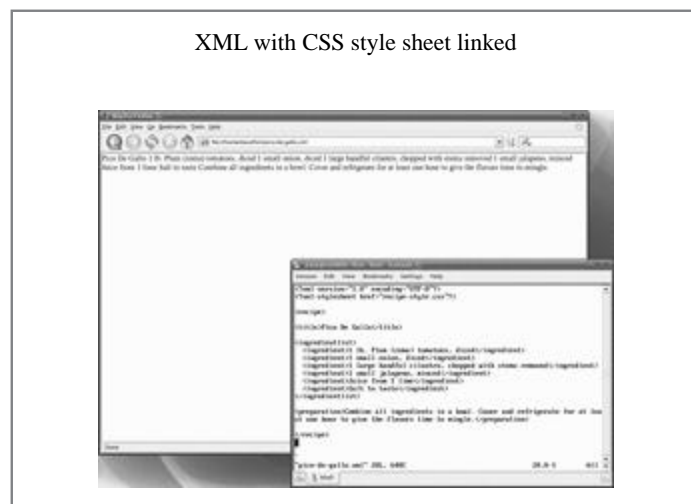
The big difference between using CSS for XML and using CSS for HTML is that browsers don’t have any built-in styles for XML. With HTML, if you specified a level-one heading, you knew the text between the `<h1>` and `</h1>` tags was going to be displayed in a large, bold font and you could then use CSS to do some tweaking like changing the color to bright red. But, when it comes to my XML recipe there is no `<h1>` tag. In fact, all of the tags are made up so I can’t really expect the browser to have built-in styles for them. Every single aspect of the document layout is up to me and must be defined within my CSS file. This includes such mundane styling tasks as telling the browser to place each individual ingredient on a new line and not run them together. While this may sound like a big hassle, it’s actually not too difficult when taken one step at a time.

Defining block elements

As a first step, I decide which elements need to start on a new line or, in CSS speak, need to be formatted as a block element. This is easier than you might think, because we humans are smarter than computers. For my pico de gallo recipe, I can simply refer to the handwritten recipe and look at how my brain unconsciously decided how to format it. Thinking in terms of where I placed line breaks it’s easy to see that the title, individual ingredients and the preparation instructions all qualify as block elements.

Once the block elements are determined, the next step is to record this information in `recipe-style.css`. CSS defines styles using the format `selector { property: value; }` so to make the recipe title a block element I just need to add the

XML with CSS style sheet linked



Block elements styled with CSS



The final CSS-styled XML recipe



line `title { display: block; }` to my CSS file. Each element that I want to style needs to be recorded this way so I should add similar lines for the ingredient and preparation elements. With these three simple CSS directives, my pico de gallo recipe should be much more legible.

Tweaking the style

The final step is to tune the aesthetics of the recipe. You may have noticed that the CSS-styled XML is still kind of boring, even compared to my handwritten recipe. Mainly it's because everything looks the same. Same font. Same line spacing. Same everything. I can improve the look of my recipe by defining more CSS properties in the `recipe-style.css` file. For example, I might want to put some emphasis on the recipe's title to make it stand out and catch the eye. This can be accomplished by using the `font-weight` property so that the title appears in bold. I can visually separate different sections of the recipe by applying CSS margin properties to the elements "ingredientlist" and "preparation". If I want to get really fancy, I can even place labels on various sections using CSS's `:before` pseudo-element. Anything that I could do with HTML and CSS I can do with XML and CSS as long as I view my document in a modern, XML-capable browser.

Pros and cons of XML/CSS

My pico de gallo recipe is looking pretty good and it didn't take too much effort to get it that way thanks to the XML/CSS capabilities of the Firefox browser. However,

if I share my recipes on the internet there is no way I can guarantee everyone else is going to be using the latest release of Firefox to view them. If somebody visits my site using an older, proprietary browser or a text-only browser this XML/CSS thing is not going to work. In that case, I'm going to have to come up with a format that their browser can handle. Luckily, there is another type of style sheet for XML documents called eXtensible Style sheet Language (XSL) that can help in this situation. XSL can do wonderful things including transforming my XML recipe into an HTML or PDF file that older browsers can deal with. But for that you'll have to wait for part two of this article.

Copyright information

Copyright © 2005 David Horton.

Subject to terms and conditions of the Creative Commons Attribution-ShareAlike 2.5 License (<http://creativecommons.org/licenses/by-sa/2.5/>).

About the author

David Horton is fascinated by many things. Currently he is intrigued by XML and the wonderful things it can do for his collection of original recipes. Dave's recipes, (<http://www.happy-monkey.net/recipes/>) and the custom, XML-based recipe markup language (<http://www.happy-monkey.net/recipebook/>) he has created to share them, can be found on his web site (<http://www.happy-monkey.net/>).