MAX 2007

CONNECT. DISCOVER. INSPIRE.

Stephanie Sullivan

Consultant

W3Conversions

1

Agenda

- What are the CSS Layouts?
- Using the CSS Layouts
- Combining Layouts
- Things to watch for
- Q&A

Stephanie Sullivan

- Founder/Principal - W3Conversions - Web Standards and Accessibility Company
- Corporate Trainer, Consultant & sub-contractor for XHTML/CSS development
- Adobe Community Expert
- Co-Lead Adobe Task Force for Web Standards Project (WaSP)
- Partner - CommunityMX.com
- Author - DW MX 2004 Magic, Web Developer's & Designer's Journal (formerly MXDJ), Adobe's DevNet Center, and other web publications
- List Mom for WebWeavers & moderator for SEM 2.0
- Coming soon - Mastering CSS with Dreamweaver CS3 by New Riders - authored with Adobe's Greg Rewis
- Contact - stef@w3conversions.com

What Are The CSS Layouts?

- CSS Layouts are structure only
  - No images
  - No styling
- Quick starts for CSS development
  - Heavily commented for those new to layout
  - Fast start for those comfortable with CSS

Browser Compatibility

- Firefox (Win and Mac) 1.0, 1.5, and 2.0
- Internet Explorer (Win) 5.5, 6.0, 7.0
- Opera (Win and Mac) 8.0, 9.0
- Safari 2.0

Five Types of Layouts

- Fixed
- Liquid
- Elastic
- Hybrid
- Absolute Positioning

A quick demonstration of the CSS Layouts

Page Structure of the CSS Layouts

- Right and left columns are floated
- Main Content column is margined on the side of the column
- Body class and descendant selectors used to define page type to allow combination - for example: .twoColFixLtHdr
- Heavy commenting explains each page section and some specific property/value pairs
- Internet Explorer Conditional Comments

Fixed

- Specific pixel width - centered

- Pros / Cons

  ✓ Full background color on columns is easy to achieve (faux columns)

  ✓ Easy to know exact dimensions for elements within the main content area and avoid float drop*

  ◉ Columns do not expand with increased text size

Creating Equal Columns

- Within the layouts, background color on the columns only extends to the end of the content

Methods:

- Fixed width can use a single repeating graphic

Liquid

- Overall width and columns based on percentage of user's viewport

- Pros / Cons
  - ✓ Allows for creative use of headers - repeat on X axis or show more when browser is wider
  - ◉ Background column color more challenging (liquid faux columns)
  - ◉ More difficult to know exact dimensions for elements to

Creating Equal Liquid Columns

- Creating faux columns when the layout is liquid

Method:

- Use a graphic with the same faux column ratio as the column you're creating
- Set the background image to the same horizontal position

Elastic

- Width based on user's default text size

- Pros / Cons
  - ✓ Layout and columns expand with text size changes not browser width
  - ✓ Allows for creative use of headers - repeat on X axis or show more when browser is wider
  - ◉ More difficult to know exact dimensions for elements to avoid float drop - use min-width

A demonstration of expanding and flexible headers

Hybrid

- Overall width based on percentage, while the side columns are based upon em's

- Pros / Cons
  - ✓ Column widths expand with increased text size
  - ✓ Allows for creative use of headers - repeat on X axis or show more when browser is wider
  - ◉ Still challenging to know exact dimensions for elements to avoid float drop - use min-width

Absolutely Positioned

- Fixed, pixel-based width

- Pros / Cons
  - ✓ Float drop not a problem since there is no floating
  - ✓ Headers and footers are simple due to set width
  - ◉ Columns are absolutely positioned and taken out of the flow of the document - footer will not “see” them

Absolute Positioning Perils

- Absolute positioning is taken out of the flow of the document - other elements are unaware of their existence and don't react to them

Demo:

- Bump text size up - watch overflow

Basic Principles

Document Flow

- The “flow” is the natural order of occurrence of the elements within the HTML
- When working with CSS for page layout, the document flow impacts the visual position of page elements – depending upon the method of positioning
- Don’t fight the flow, use it!

The Display Property

HTML elements, by nature, have one of two renderings:

- Inline
  - Inline-level elements render horizontally until they run out of space, then wrap to the next line.
  - They only take as much space as they need
    Examples: img, span, a, em, strong
- Block
  - Block-level elements render vertically as if there's a line break above and below them
  - They take up 100% of their parent container
    Examples: p, div, h1, ul, blockquote

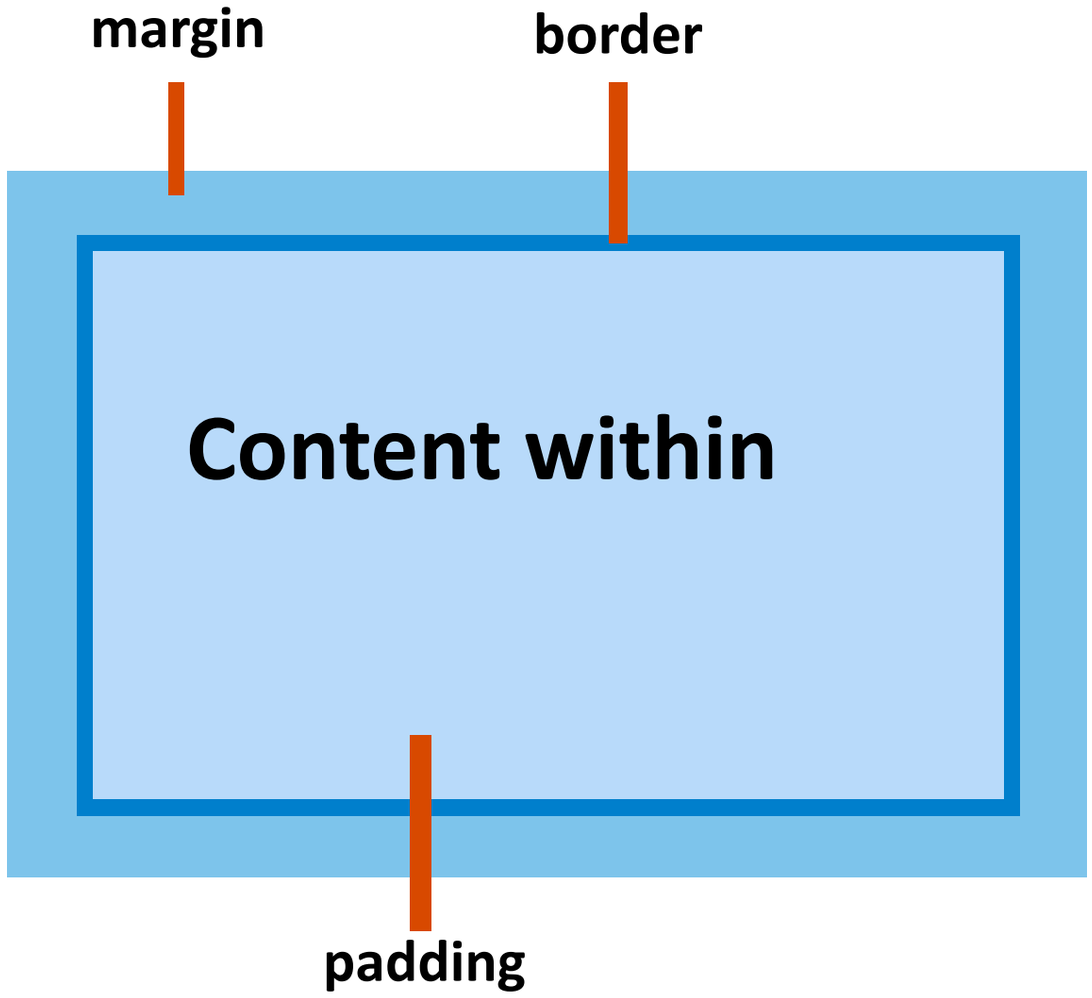
The Display Property

CSS can be used to change the display property of an element

- display: block can be given to a span or an image to make them stack vertically
- display: inline is sometimes used as a fix for Internet Explorer's
  3 px bug (added to your math)
- display: none causes a block to render no box at all

- Changing the display property of an element changes its presentation, but not the nature of the element itself.

Understanding the Box Model



Types of Positioning

- The four types of positioning available using CSS:
- Static
  - The default location of the element in the document flow
- Relative
  - The element's position is relative to its position in the document flow
- Absolute
  - A "XY" coordinate based upon its parent container
- Fixed
  - A "XY" coordinate based on the viewport

A quick demonstration of Positioning

Principles of Floating

- A float must be given a width
- A float must be given a directional value of left or right (there is no top or bottom)
- If you want a float to appear alongside another element, it must precede that element in the source order of the document
- A float never covers text or inline images
- Since a float is taken "out of the flow" of the document, a float inside another container must be cleared in order for the parent container to enclose it properly

A quick demonstration of floating

Points to Remember

Clearing Floats in the #mainContent

- Clearing within a non-floated div will clear **all** floats – including the side columns.

Fixes:

- Float the #mainContent div
- Place a floated div within the #mainContent div

Beware of Float Drop

- Evident when one div starts below the level of the div next to it

Causes:

- An element, like an image, that is wider than can fit in the space provided. The div will move down until it can fit next to the floats. (Make sure clients who are taking care of their own sites with Contribute are aware of their size specs and limitations.)
- 3px text jog in Internet Explorer (unaccounted for in your math)

A quick demonstration of float drop

Hands-on demo - Combining CSS Layouts

Q&A

Resources

- **David Powers - Strip Comments RegEx**
  http://foundationphp.com/tools/
- **Adobe CSS Advisor beta**
  http://www.adobe.com/go/cssadvisor
- **Community MX**
  http://www.communitymx.com
- **W3Conversions**
  http://www.w3conversions.com
- **Amazon.com** (for preorder)
  Mastering CSS with Dreamweaver CS3

Thank You
Danke schön
Dank u wel
Merci beaucoup