

CSE 102

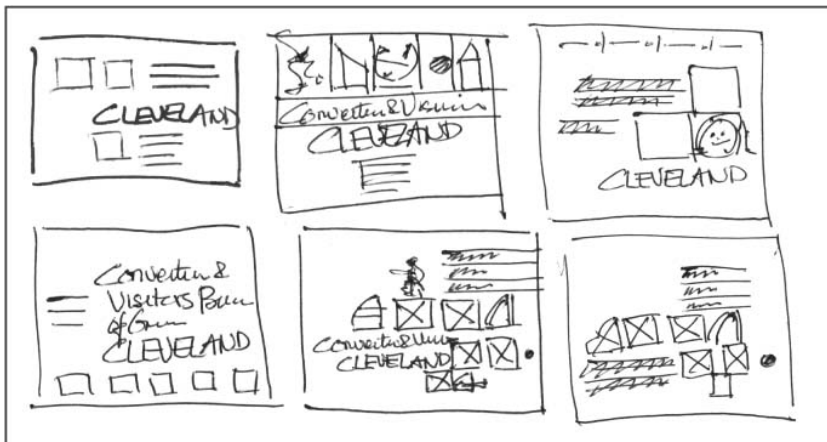
Introduction to Web Design and Programming

Information Architecture

Layout

- Design – the overall process of communicating and editing ideas in the visual arts
- Layout – the actual arrangement and placement of content in a 2D space
 - layout emerges after you have developed the overall design for the project
 - begin with defining its information architecture
- Information Architecture – a blueprint for the site's information delivery, organization, & functionality

Thumbnails: quick, small drawings



Practical IA models

- Book model
 - with table of contents, chapters, sections, etc ...
- Newspaper model
 - with front page, headlines, sections, & quick index
- Organizational model
 - Reflect the departments & administrative hierarchy of a company or business
 - management, customer service, marketing, sales, personnel, etc ...
- Department store model
 - An entry directory of goods & services, departments with their own items
- Audience-oriented model
 - Different types of site visitors are immediately directed to different parts of the site
- Others:
 - Museum model
 - Library model
 - Map model

Logical IA models

- Hierarchical model
 - info is organized into a family-tree structure
- Central dispatching model (star topology)
 - main contact point directs visitors to different parts of the site
- Linear model
 - info is organized in a sequence of previous & next pages
- Others:
 - connected graph model
 - circular model
 - distributed model
 - catalog model

Steps for defining a site's IA

- Should be done before creating any Web pages
1. Define site goals
 - What are purpose, scope of site?
 2. Define audience
 3. Create & organize content
 4. Formulate visual presentation concepts
 5. Develop site map & navigation
 - http://www.iconsupport.demon.co.uk/Map/Site_Map.html
 6. Design & produce visual forms

Layout Grids

- A set of invisible vertical & horizontal lines to guide content placement
- The primary way designers organize elements in a 2D space
 - aligns page elements vertically & horizontally
 - marks margins
 - sets start & end points for element placement
 - ensures unity throughout a site

Web Page Layout Grids

- Issues:
 - different screen sizes & resolutions
 - different browser fonts available
 - different browser default font sizes
 - alternative device browsing (TVs, PDAs, etc ...)
 - print ready pages (8.5" x 11" paper)
- Basic layout strategies:
 - left-justified layout
 - centered layout
 - full-width fluid layout
 - centered fluid layout

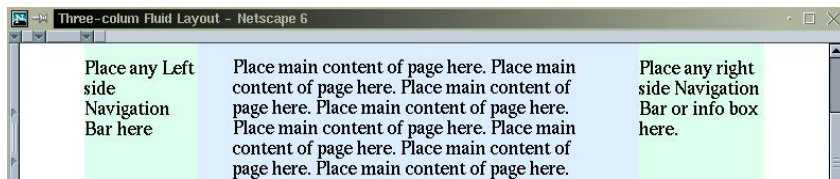
Fixed Width

Fills width of browser window

Fills browser window & has equal left & right margins
- Techniques for fluid layouts
 - CSS techniques (good option)
 - scalable HTML tables (bad option)
 - combination of tables & CSS techniques (also bad option)

Research has indicated surfers prefer fluid layouts

A Fluid Table Grid



- 3-column grid
- centered horizontally in the browser window
- comfortable left, right, & bottom margins
- use tables, assign column widths as %s

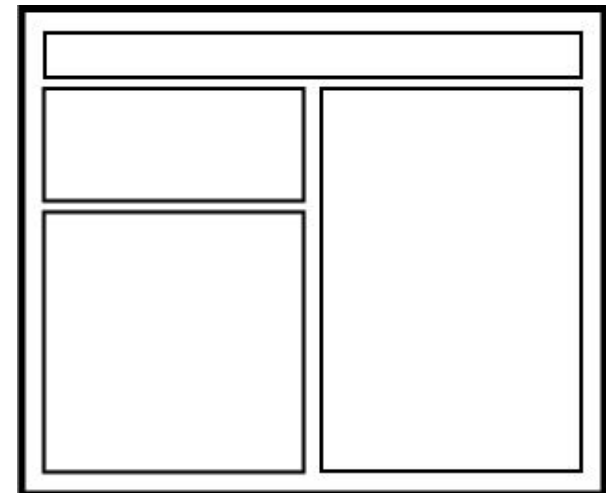
Scalable Grids with Tables

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
      xml:lang="en" lang="en">
  <head><title>Three column fluid layout</title></head>
  <body style="margin-top:0px; margin-right:8%; margin-bottom: 50px; margin-
    left: 8%">
    <table width="100%" cellpadding="0" cellspacing="0">
      <tr valign="top">
        <td style="width:14%">Place any left side navigation bar here</td>
        <td><div style="margin-left:8%; margin-right:8%;">Place main content of page
          here. Place main content of page here. Place main content of page her. Place
          main content of page here. Place main content of page here. Place main content
          of page here.</div></td>
        <td style="width:14%">Place any right side navigation bar or info box here</td>
      </tr>
    </body>
  </html>
```

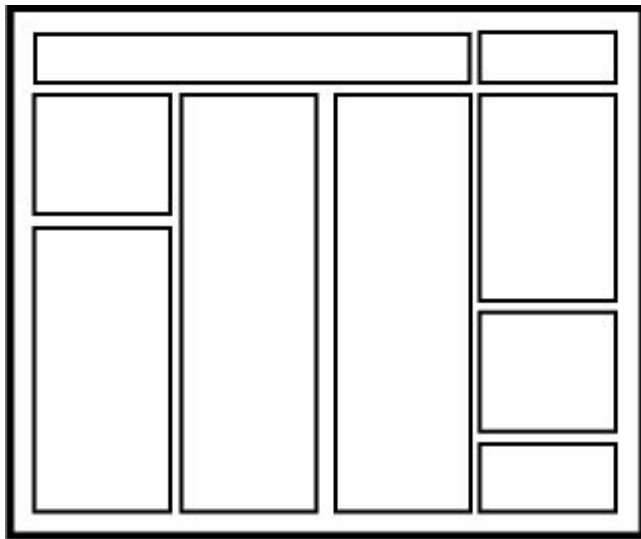
Designing Layout Grid Systems

- Remember layout?
 - A grouped set of elements arranged to create aesthetically pleasing balance & proportion in a 2D space
- Use grids
 - first draw grid by hand (thumbnails)
 - use tables and/or CSS

Two Column Grid



Four Column Grid



Site using Four Column Grid

STAGE	STEP	DELIVERABLE
Strategic Planning	<ul style="list-style-type: none"> Determine/clarify goals Conduct SWOT analysis Review current technical systems Validate or reject proposed activities 	<ul style="list-style-type: none"> Completed questionnaires Checklists SWOT/competitive analysis document(s) Preliminary strategic technical plan
Tactical Planning/Estimating	<ul style="list-style-type: none"> Select specific methods, technologies Define functionalities Develop budget parameters Review interface and branding considerations 	<ul style="list-style-type: none"> Site maps Estimates Project binder Business requirements document Timeline document(s) (Calendar, Gantt Chart) Creative brief
Project Detailing, Architecture	<ul style="list-style-type: none"> Define and specify technical systems Create project architectures Determine data flow Develop screen layouts Create user interfaces 	<ul style="list-style-type: none"> Detailed data flow docs Screen layouts Database document(s) (ERD, data requirements document)

XHTML & Tables

- XHTML discourages you from using Tables for layout
 - tables should only be used for data
 - not yet fully supported
 - in reality tables will be in common use for layout for near future
 - CSS layout will gain in popularity

Why not use tables for layout?

- Not valid XHTML
- Tables add complexity
- Pages with tables (especially nested tables) load more slowly
- Tables hurt search engine optimization
- Tables don't always print well
- <http://webdesign.about.com/library/weekly/aa111102a.htm>
- Alternative to tables?
 - CSS positioning

CSS Positioning

<http://webdesign.about.com/library/weekly/aa102102a.htm>

- Use the <div> element to define the different content components
 - Ex: left navigation, right navigation, content, banner, etc ...
- Use CSS to place the components on the page using the following format (use the properties you want):

```
#component_name{
  position: absolute;
  left: ?;
  top: ?;
  width: ?;
  z-index: ?;
  margin: ?;
  color: ?;
  padding: ?;
}
```

NoTablesLayout.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>
<head>
<title>Tableless Web Pages</title>
<link href="/css/NoTableLayout.css" rel="stylesheet" type="text/css" />
</head>
<body topmargin="0" leftmargin="0">

<div id="leftnavigation">
<h1>Left Navigation</h1>
<ul>
<li><a href="http://webdesign.about.com/">Home</a></li>
<li><a href="http://webdesign.about.com/library/weekly/aa102102a.htm">Tableless Browsing</a></li>
<li><a href="http://webdesign.about.com/library/layouts/bl_layouts.htm">XHTML CSS Layout Templates</a></li>
</ul>
</div>

<div id="content">
<h1>Sample Content</h1>
<p>This is the content section of the page. As you can see, I'll use structural markup like
&lt;p&gt;&lt;strong&gt;&lt;/strong&gt;, and &lt;em&gt;&lt;/em&gt;. This will help keep the page valid in XHTML.</p>
<p>Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem
ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum
sit dolor amum.</p>
<h2>Lorem Ipsum</h2>
<p>Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem
ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum.</p>
<p>Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem
ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum.</p>
</div>

<div id="rightnavigation">
<h1>Right Navigation</h2>
<p>Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem
ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum. Lorem ipsum sit dolor amum.</p>
<h2>Lorem Ipsum</h2>
</div>

</body>
</html>
```

NoTablesLayout.css

```
body {
  margin : 0;
  padding : 0;
  border-color : #cc0000;
  background-color : #ffffff;
  color : #000000; }

p, ul, ol, li {
  font : 10pt Arial,geneva,Helvetica; }
h1 {
  font : bold 14pt Arial, Helvetica, geneva; }
h2 {
  font : bold 12pt Arial, Helvetica, geneva; }

#leftnavigation {position : absolute;
  left : 0;
  width : 150px;
  margin-left : 10px;
  top : 20px;
  color : black;
  padding : 3px;
  }

#rightnavigation {position : absolute;
  left : 80%;
  top : 20px;
  width : 140px;
  padding-left : 10px;
  z-index : 3;
  color : #000000;
  padding : 3px;
  }

#content {
  top : 0px;
  margin : 0px 25% 0 165px;
  padding : 3px;
  color : #000000;
  }

#content h1, #content h2 { color : #cc0000; }
```