Printing

Users may want to print a Web page on the printer. The main thing to keep in mind when designing pages for printing is printer limitations. Some table cell colors may print well, others may not. Background colors and images sometimes look nice, other times may cause readability problems. Flash pages are virtually unpredictable when it come to printing. Many sites designed only to view on the screen and printing on an 8.5 × 11 page is often added as an afterthought.

Identify the importance of page printing early in your Web project. That can guide your design for the site. It is perhaps always good to pick a design that is easy on the printer. Testing in the early stages of development is the key. Using CSS to adjust your text, layout, and colors for the screen and the printer is another very good option (Section 6.21).

Type and Cascading Style Sheets

For a Web page, leading, font size, and other text display properties are controlled through Cascading Style Sheets (CSS) (Chapter 6). Now that you know a little more about type, you can begin to apply that knowledge to creating great looking, smart type, on websites. Below is a list of features which can be controlled through CSS (See Chapter 2 and 6).
- font color, size, and weight
- style: such as normal, oblique or italic
- Uppercase, lowercase, or small caps
- font family in order of preference, separated by commas
- line height or leading, letter and word spacing
- text color, background color, background image
- white space properties
- text indent, alignment, underline, and more.

**Font Family, Color, Size, Weight, Style, and Variant**

Fonts of different designs belong to different *font families*. Within each font family you can set the font color, size, weight (boldness), style (italic, slant), and variant (smallcap or normal). The CSS properties `font-family`, `color`, `font-size`, `font-weight`, `font-style` and `font-variant` are used to set these styles (Section 2.10).

A well selected set of fonts is an important part of site design. Since not everyone has the same set of fonts on the computer, it is a good idea to always specify at least 2 font choices. You may use very specific font names like Berkeley Bold Italic. If unsure of the second choice, you can always fall back on one of the following generic font names: sans-serif, cursive, fantasy and monospace.

If the viewer doesn’t have the font specified in a designer’s italic face, the browser will display text in Oblique. Italic tags can be added and removed depending on the text you’re formatting. (Palatino Italic or Berkeley Italic or you can just say Palatino and then choose italics depending on how much italic text you have and how often you want to change it.) Also consider removing italics as a way to emphasize a term inside italic text.
Line Height and Leading

The style property `line-height` (Section 2.13) can be set to control the vertical separation of lines. For on-screen readability it is recommended that you use plenty of vertical separation, say setting line height to 1.5 the font size.

In CSS, *leading* precisely refers to the difference between line height and font size:

\[
\text{leading} = \text{line-height} - \text{font-size}
\]

See Section 2.13 for how to set values for the style property `line-height`.

Tracking and Kerning

Both tracking and kerning can be controlled through CSS as can indents before paragraphs. To specify tracking type in `word-spacing:length`, where length is a numerical value in pixels, point, ems, etc. To specify kerning, type `letter-spacing:length`, where length is a numerical value in pixels, point, ems, etc. To add indents, type `text-indent:` where the number is an absolute value or a percentage. There are several things which you need to keep in mind when working with kerning and tracking values. You may specify negative values for word and letter spacing but be careful, the display of these features always depends on the browser’s capabilities. Also, word and letter spacing values may be affected by your choice of alignment. When justifying text, it is best to use a zero value for letter spacing to avoiding large open holes. If you want to use default letter and word spacing, be sure to set the value to normal.

Aligning Text

To align text, type in `text-align`, then write any of the following 4 choices:

- **left**—align the text to the left
- **right**—align the text to the right
• center—align the text to the right

• justify—align the text on both the right and left

As a general rule of thumb, justified text tends to look awkward on the screen. Word spacing and letter spacing may be adversely affected. Use with caution and only when absolutely necessary!

![Figure 5.28: Justified Type](image)

Justified type tries to align all text so they it is perfectly flush on left and right sides. This usually results in odd word and letter spacing which is not aesthetically pleasing and is difficult to read. We do not recommend that you use this option.

Underlining Text

Underlining text is generally unpopular with many designers and for good reason. Underlined text tends to be distracting and often looks “dated,” evoking images of old Smith Corona typewriters of times long past. Nevertheless, if you must underline, CSS lets you underline and remove underlining easily throughout your document.

To underline text, type `text-decoration: underline`. For a line above the text, type `overline`. And for strike out text, type `line-through`. To remove underlining, overlining, or strike through text, type `text-decoration: none`.

If you don’t wish to use underlining to indicate links you can type the none option for text-decoration. In this case, be sure to also eliminate underlining under visited links, active links and transitional links. However, be careful with this option, you still have to visually clue your viewers to the links on the page. Consider changing text color or changing background behind the text.

**Legibility and Readability**

*Legibility* refers to how well your typeface supports fluent reading. The typeface you choose may help or hinder that process. For example, if you choose a typeface that is too small, the characters may run together and make the words difficult to read. Reversing white type out of a black background can also be difficult to read because the brightness and glare of a white against black, blurs or runs words together making them difficult to distinguish from one another. Text set in all upper case letters needs more word spacing to ensure that each word can be read. Type which has no vertical spacing or leading, may fuse together and appear as one large gray block. Adding at least 2 points (or 20% in your style sheet) of leading will alleviate this problem.

*Readability* refers to the ease of reading type blocks rather than the typeface itself. In other words, you can choose a perfectly legible typeface for your layout, but if the style and design of your layout is poor, viewer’s may find your text unreadable.

**Type as Graphic**

Early Web designers didn’t have the flexibility to choose fonts beyond the standard Times and Helvetica. They didn’t have the option, as they do now in style sheets, to specify leading or indents for running text and even creating drop caps and raised caps (Figure 5.29) with CSS (Section 6.14). The treatment of the initial letter in a paragraph can be used to break up the monotony in chapters, articles or large amounts of text. Instead they relied on type images created with photo manipulation programs Photoshop to add artistic touches to type. The technique is still valid today. There are certain advantages to treating type in...
this fashion. One, you can choose any font, any color, any size, any leading and any tracking you wish, and embed it as a graphic wherever you wish. You can control the look of your headlines ensuring corporate identity type standards, as well as overall aesthetics. Two, you can manipulate, stretch, gradate and apply filters to express virtually any idea with type, adding creative possibilities which can’t be created any other way. To most designers, even with the advent of style sheets, this prospect looks very appealing. However, the argument for not using type as graphic throughout your site is compelling and it should be considered in the design process.

There are four primary reasons for using this type treatment sparingly or in some cases, omitting it all together. One, embedded graphics are not searchable by search engines which make it difficult to find your site. Prospective visitors may find this to be a problem. Two, embedded type graphics can’t be read in text only browsers such as those used in speech devices on most ADA compliant sites, so visitors who rely on those devices (U.S. figures shows 20% of the population have some sort of disability) are not going to be able to hear your site. Three, graphics add more weight to your site and increase download speed. It may not be noticeable on smaller sites, but on bigger sites, say 1,000 pages or more, extra graphics can really bog down the speed significantly. Four, type created as a graphic takes more time to edit. For example, on sites that are updated several times a day, this can significantly slow down maintenance and create lots of extra updating work for the Webmaster or designer.

The text formatting ideas discussed here are further reinforced in Section 6.14 where CSS
coding and examples are covered.

5.11 Reading Type on the Web

When designers first began to work with type on the Web, many tried to force print rules on to the new medium and many have failed. The reason for the failure is due basic control issues. Many of the things which make text successful in print fail on the Web. Such things as kerning character pairs and ligatures, for example, just aren’t practical for most text. Letting go of hanging punctuation or set widths is also necessary. Trying to control browser windows, in most cases, frustrates many viewers who would prefer that your text works in their window and not yours. Print and CD ROMs are fixed media, and as such, allow and encourage fine control. Text on the Web has to be dynamic—a marriage of efficient editing and aesthetically pleasing communication. Designing with respect for your viewers and capitalize on the strength of the medium.

Legibility and Readability

Legibility and readability are two terms used both in print and on the Web. In both mediums, the words mean the same thing. Legibility means being being able to see and identify words on a page. For example, we can safely say that 2-point type is illegible on the screen. Readability refers to the flow and the ease of flow when reading text. It is possible for type to be legibility but not readable. So, we can say that type readability is relative, not absolute.

Ways of Making Type Readable on the Screen

Applying common sense may be the best advise when working with text. The two key points to making type readable on the screen are size and space, which are more challenging than they appear at first glance. On-screen type is contingent upon resolutions, platforms, and browser rendering capabilities. These variables make it difficult to say it is best to always use
10 or 12 point Times. The best way to get consistent cross-platform results when choosing type sizes is to specify a standard font size, such as medium or small (Section 2.13), and pair that with sufficient leading.

Proportional relationships (or size) between type are important because they establish hierarchy for headers, body text, photo captions, copyrights and other legal text. You have control over this aspect of type so take advantage of it. This hierarchy is one way to prioritize information which is one of designer’s most essential functions.

One easy way to make type more readable is line length. The optimal length of a line of text should be between 40 and 70 letters long. Shorter lines tend to disrupt reading flow. Longer than that and viewers have a hard time finding the next line, so it slows reading. When using default browser font, a good rule of thumb is that your text blocks should be around 400 pixels wide.

Another good rule of thumb for making type readable is keep the backgrounds simple. This means eliminating extraneous textures or patterns which may make reading difficult. Textured backgrounds may be interesting when viewed on their own, but often do nothing but add noise behind the text.

When using CSS, it is easy to change horizontal spacing and tracking of words. Tracking refers to the space between letters. Avoid tracking the letters too closely. Too little space between words can lead to words touching and on the screen and individual letters have to be clearly distinguishable for type to be readable.

Designing for ‘scanability’ is also very important when working with text on the Web. In this case, scanning means looking at the page quickly to pick key words, or phrases of interest. Shorter paragraphs and quick reading subheads may be helpful in helping viewers spot the items rather than seeing one large page of gray type. Breaking up the page into smaller portions may also help viewers get to their topics more quickly. Given that people read text on-line becomes very important. Make it easy people to read your message.

Leading or space between lines, is another aspect of type which can improve readability. Leading allows the eye to see the type lines more clearly, without interference from the lines.
So, with all that said...What fonts are more readable on the screen? The answer is both good and bad. The bad news is that the fonts which are used the most on the Web, Times and Courier, are not especially good for readability. Although Times is excellent for printing newspapers, it isn’t particularly good for the screen. The original version of Times was designed to squeeze as many letters as possible on a printed page and still remain legible. The print version is narrow, with medium x-height and sharp, small serifs. Its small ascenders and descenders make it very economical, because they reduce the need for leading, enabling the typesetter to set more lines per page.

The bimapped version of Times does not translate the subtleties of the print version because the smallest possible unit on the screen is the pixel. Half pixels don’t exist to accommodate the details of many of typefaces and even anti-aliasing does not help this problem. Unfortunately, most screen fonts were designed to mimic their printer fonts at 72 ppi in QuarkXPress and were not designed to be legible on screen.

The good news is that many more fonts are being optimized for on-screen reading. Products like Microsoft’s Fontpack and Adobe’s Webtype contain fonts which have been designed to fit into a square pixel grid. The fonts have more open face, wider letters, increased x-height and more letterspacing.

The WDP site uses the following CSS rules for running body text and headings. For example,

```css
body
{
  font-family: Verdana, Geneva, Arial, helvetica, sans-serif;
  font-size: small; /* one notch below medium size */
  line-height: 150%; /* 1.5 spacing */
}
```

See Section 2.13 and Section 6.2 for CSS information on fonts and text styling. Be sure to check your pages using various browser versions on different platforms (PCs, Macs, and UNIX).
5.12 Layout Grids

A grid is a set of invisible vertical and horizontal lines to guide page layout. It is the primary way designers organize elements in a two-dimensional space. A grid aligns page elements vertically and horizontally, marks margins, and sets start and end points for element placement. A well-designed grid makes a page visually clear and pleasing, resulting in increased usability and effective content delivery. A consistent page layout also helps to create unity throughout the site (Section 4.5).

Grids are certainly not a new invention. They have been used for centuries as the basis for ornamental design in screens, textiles, quilt design (Figure 5.31 Ulius Friedman/Walter McCord, Designers; Craig Guyon, photographer), posters (Figure 5.32, Herbert Bayer’s 1926 Exhibition Poster for Kandinsky’s 60th Birthday Exhibition in Dessau) and architecture. In the twentieth century, the grid itself became popular as designers began using the shapes...
themselves as art. Bauhaus architects in particular used the grid in a way that showed both the inside and outside building architecture. Layouts with strong grid structures can be traced to the de Stijl movement where artists like Van Doesburg and Mondrian began dividing their canvases into symmetrical patterns. Swiss designer J. Muller-Brockmann had an important impact on defining and shaping two-dimensional design though the use of the grid in everything from photography, architecture, to logo designs. His legacy still remains in much of today’s popular culture, permeating our computer screens and printed materials with “Swiss design.”

Figure 5.32: Grid: Strong Horizontals and Verticals

5.13 Web Page Layout Grids

Grids in Web design are substantially different from grids in print. Print pages have fixed and known dimensions. Whereas Web pages are displayed on monitors of different sizes and resolutions and in resizable browser windows.

Major layout challenges for a Web designer include:

- Most CRT monitors and LCD screens have a 4 × 3 aspect ratio and settable resolutions: 640 × 480, 800 × 600, 1024 × 768, 1400 × 1050, and higher. Sometimes you’ll find irregular Screen Sizes such as 1280 × 1024. Different resolutions can make pages look smaller or...
larger. With the move to HDTV, computer screens may soon have the 16 × 9 aspect ratio.

- The maximum available display area for Web pages is the screen size minus the space taken up by borders and tool bars of the browser used. For example on an 800 × 600 monitor the usable area may become 750 × 425.

- Browsers can use different default font sizes according to user preferences.

- Browser windows can change size dynamically.

- Some users may surf the Web on TV screens, palmtops, text-only devices or Braille readers.

- Users may want to produce hardcopies of pages on a printer. The typical 8.5 × 11 inch paper is very different in size from monitor screens.

Designers generally have used these basic layout strategies:

- Left-justified layout—a fixed-width grid is used and the page starts at the left margin. This design can leave an annoying white space on the right side for larger resolution screens.

- Centered layout—a fixed-width grid is centered horizontally in the browser window.

- Full-width fluid layout—the grid is scalable and fills the width of the browser window and responds to browser resizing dynamically. For larger screens, text lines can become too long for easy reading.

- Centered fluid layout—the layout uses fluid but equal left and right margins and a fluid centered grid.

Surveys show users dislike having to do horizontal scrolling the most. Fixed-width pages runs the risk of causing horizontal scroll for entire pages. According to the Software Usability Research Laboratory of Wichita State University:

Fluid layouts are significantly preferred to both centered and left-justified layouts. In a study by Bernard and Larsen (Winter/2001) participants indicated they perceived the fluid layout ... as being the best suited for reading and finding information, as well as having a layout that is most appropriate for the screen size (for both small and large screens). They also indicated that the fluid layout looked the most professional, and consequently preferred it to the other layout conditions. Conversely, the consistently least preferred condition was the left-justified layout.

Achieving fluid layout with a scalable grid is a design challenge (Section 6.15 and 6.22). The principle technique is to use, instead of fixed widths, relative proportions for the grid columns and margins. Images and font sizes normally will not automatically scale up or down with window size. By creative use of multiple columns, spacing, centering, and automatic reflow of text content, a fluid page can be made to look good under most viewing conditions. A scalable page offers more flexibility and freedom for the user to resize the browser window and can fit on printer paper as a side effect.

The main objective of your design is to create aesthetically pleasing layouts, regardless of the window size. Proportion of elements to one another becomes the key factor in scalable page layouts. In other words, create contrast between elements, consider groupings, determine hierarchy, etc., as it relates to each element. Make the page look ideal for the most popular window sizes, and then make it scale well to smaller and larger window areas.

Specific techniques to use a fluid grid will be discussed in the next subsection. But be sure to test a fluid layout at all different sizes in initial design stages so that final results are pleasing. General suggestion for well designed, proportional layouts in any size is adhere to basic principle of design. Use contrast to establish hierarchy, strive to design the negative space to be as interesting as the positive, be sure to create a focal point, and dynamic relationships between elements (See Section 4.1).

Fluid layout can be achieved with Cascading Style Sheets (CSS) techniques (Chapter 6) or with scalable HTML tables. And the two can also be combined. The \texttt{<table>} element is
flexible and widely used to implement layout grids. We will look at grids with tables here.

**A Fluid Table Grid**

As an example, let’s take a look at a three-column grid (Figure 5.33), centered horizontally in the browser window leaving comfortable left, right, and bottom margins.

![Figure 5.33: A Three-Column Fluid Grid](image)

Let’s look at the HTML code (Ex: Grid3) for the grid ready to receive page content and graphics:

```html
<?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:0; margin-right:8%;
   margin-bottom:50px; margin-left:8%">
    <table width="100%" cellpadding="0" cellspacing="0">
      <tr valign="top">
        <td style="width:14%">
          <!-- Left Navbar Begin-->
          <!-- Left Navbar End-->
        </td>
        <td style="width:64%">
          <div style="margin-left:8%; margin-right:8%;">
            <!-- Main page content begin -->
            <!-- Main page content end -->
          </div>
        </td>
        <td style="width:14%">
          <!-- Right Navbar begin-->
          <!-- Right Navbar End-->
        </td>
      </tr></table></body></html>
```

The body element specifies a bottom margin of 50 pixels. It is important to leave some bottom margin and not run the text to the very bottom edge of a page (line 1). The body style also reserves 8% each for scalable left and right margins. A full-width table with cell padding and spacing both set to zero, is used for the three column layout grid (line 2). The left most column, receiving 14% of the table width, is a good place for a navigation bar whose graphical images with matching background color can make this a decorative and functional part of the page (line 3). The same percentage table width is also given to the third column for placing any informational boxes, news items, or links to pages external to the site (line 5). The center column, receiving the remaining 72% of the table width, displays the main content of the page (line 4). To give the contents breathing room, we add a left and right margin each of 8% of the center column width.

Follow the same proportion principles in this example to design your own grids using tables.

**Tips for Creating Fixed and Scalable Grids with Tables**

It would be unreasonable to say that you should design for all screen resolutions all the time, by always making tables scalable in order to ensure resolution-independent design. This kind of approach applied across the board could unnecessarily limit your design. You have to design for your target audience.

Whenever possible, try to avoid using fixed sizes and specify layout as percentages of the available space. For example, when attempting to put as much information as possible on the home page, as in newspaper/magazine style format, it may be more dynamic to use scalable tables, but the complicated nature of these layouts may make construction more difficult and time consuming.

Pages that are designed to display in fixed width which may look poor when printed. Either they come out as a thin stripe wasting loads of paper, or they are cut off because they are too wide for the printer. Be sure to include separate documents for printing.

A typical 600-pixel-wide layout that works on most normal computer monitors (although
not on WebTV and small-screen devices) will be 8.3 inches wide when printed at the standard resolution of 72 pixels to the inch.

If you’re trying to design for desktop computers as well as small-screen devices or television sets, you may be better off to create 3 different style sheets which will customize your site for optimal viewing on all devices. It may be more time consuming and expensive to do that, but at least you’ll be assured that your entire viewing audience will see your site.

In situations where it possible to design with a scalable browser window in mind, here are a few things to consider:

In general, sites that have lots of running copy seem better suited to take advantage of scalable pages. Pages that are copy light can look odd when stretched to full screen width.

Making good use of screen real estate is smart, but be careful when creating layouts where lines of text that go on and on across the screen. Such long lines are very hard to read.

Using a 2-4 column approach is a good place to start to avoid long lines of text, but also consider what will happen to the layout when the browser window is reduced to a narrower width.

Using tables, and in some cases using tables and frames, you can combine fixed-width columns along with columns that stretch and reflow with the browser window—controlling part of the layout while other elements reflow may help combine both approaches.

If continuity and ease of maintenance is important, it may be best to create all fixed or all layouts for your site.

Page headers or headlines created as graphics—as opposed to straight HTML, can help prevent variable column widths from collapsing when a browser width is reduced. Inline images in a line can be placed inside a div to prevent them from flowing into multiple lines when the browser window narrows. Text and images can also be kept on the same line but setting the white-space style property to pre or nowrap.

Standardizing the sizes of inline graphics (photographs associated with a story, for instance) can help make text reflow more manageable—it’s easier than using many different
sizes of inline images.

Treat the inline graphic as a thumbnail which pops up a detail (larger) view image in a new window.

5.14 Designing Layout Grid Systems

Once you have grouped and prioritized content, you are ready to design a layout. A layout can be defined as grouped set of elements arranged to create aesthetically pleasing balance and proportion in a two-dimensional space. Unfortunately, there isn’t a mathematical formula for creating layouts. Layouts are a product of sensitive eyes using dynamic arrangements and contrasts to lead viewers eyes in a specific way revealing specific message. So, which do you create first, the layout or the grid? There aren’t any steadfast rules, only suggestions for possible methods on how to begin. To construct a grid first is to arbitrarily force elements into specific arrangement which may not make sense for the subject. To create a whole layout without first considering the grid is equally impractical since the grid will serve as the basis for structure. Perhaps the best way to begin is to refer back to your thumbnail sketches. Look at where you have indicated text, logo, photographs, illustrations. Don’t look at just the home page, but a series of thumbnails throughout the site. Look to see what grid would make the most sense.

The layout for a site will most likely be implemented by HTML tables (Section 3.11), perhaps with the help of style sheets (Chapter 6). The code can be generated by a tool such as Dreamweaver and used as a template (Section 11.12) for creating actual pages your site.

The key point to remember when constructing a grid is flexibility. Remember one grid will be used for the entire site! You don’t want to construct a limited one-column grid which will only give you one option for placement. With a one-column grid you can only place graphics into one width. A two-column grid is a little more versatile because it allows you to place elements into one or two columns (Figure 5.34). A four-column grid is more versatile yet, allowing table cells to span from one to four columns (Figure 5.34 and 5.35). With a
six-column grid, normally you won’t use all columns for text. Some columns can provide negative space, before, between, or after columns for page content. Text placed in an overly narrow column becomes difficult to read due to poor and frequent line breaks.

You can get very elaborate with your grid design and if that kind of complexity makes sense for your design that’s great. In the case of the San Francisco Museum of Modern Art (Figure 5.36), this kind of solution made perfect sense. The six-column grid was versatile enough to support many different kinds of information ranging from an exhibit section, to general museum info, to a gift shop, creating a unique look for the museum. The unusual, stationary fifth column was designed to support additional links for membership and donor information. This grid was used through most the site.

note!

5.15 Grids on the Web: Critique

Now that you know little more about grids, let’s examine several sites to see how they designed their grids and what makes them successful.

One of the sites which uses a strong, interesting grid is the Sapient site (sapient.com). If you examine Figure 5.37, you will see that each pages uses the same grid, yet the permutations

note!
on each page vary depending on the sections. Permutations in this case means to vary or rearrange a set of elements such as type, photographs and graphics to appear differently.

Grids, at their best, should not make every page look exactly alike. Yes, its is true that designers strive for unity, but there is a fine line between unity and monotony. Unless you’re laying out a telephone directory or another totally uniform document, each section of your site should vary somewhat. Why should it vary? The simple reason is the nature of the content may vary. You would not use the exact same layout on a home pages as you would on a sitemap page, for example. And, you would probably not use the exact same layout on the merchandise ordering page. The information on each of the major sections is different, therefore, the arrangement of the elements reflects that. The inspiration for the variations
in form is content. If you begin with that premise, all of your formal arrangements will make sense. So, consider the layout needs of the entire site with its different sections and contents then design a grid system that can be used flexibly. A flexible grid allows you to vary the basic layout by combining columns (or rows) to form a wider column (or row) and to divide a column (or row) into smaller ones. The variation is done while preserving certain unifying features of the basic grid and page style.

Now, let’s examine the Sapient grid more closely. The home page (Figure 5.37) divides the page into halves, introducing a thin white line which will later be used to organize type and images. Main navigation appears on the red and white rectangle floating on top of the photograph here and throughout the site. Further sub-navigation appears on the far right hand side on a vertical, an alignment choice which usually does not work on the Web. In this case, it happens to fit into the design and it is not difficult to read. Other sections
on this site (Figure 5.37) use a two column grid for text and single column on the left for navigation.

Resize the window on the Sapient site entry and you’ll find a fluid grid that responds well to different window sizes. The vertical navbar on the right hand side stays visible while the middle parts of the page gives up space. All the pictures on the site entry are background images! The right-most table column, which displays the vertical navbar, has "width: 1%" while the buffer columns are left without width specifications. Thus, HTML table uses background images and cells with black background in the center columns to act as buffers for changes in window size (Section 6.18).

Grids don’t have to be complicated or clever to be effective. MoMa, The Museum of
Modern Art (moma.org) in New York uses a simple grid for most of their pages. The homepage grid is exactly like all the other grids in the site Figure 5.38. Large, horizontal space at the top allows for display of exhibit photos and the area underneath introduces a simple, 3-column structure for navigation and text.

Dotted horizontal rules organize the navigation bar type and solid rules to organize body text. We should also add here that one of the things that makes this site aesthetically pleasing is the sensitivity to type, choice of color and generous use of negative space.

Some sites don’t appear to use grids at all. These types of layouts are referred to as free placement and are based on “intuitive” placement of elements inspired by the content. Although most commercial sites rely on internal structure to organize complex information, there are some sites which effectively use minimal or no structure. One such site is Potlach Brooks/Cole book/January 28, 2003
Papers ([www.potlatchpaper.com](http://www.potlatchpaper.com), Figure 5.39). Much of the site is done in Flash, appropriately chosen for its animation. At first glance, it may appear that the elements are freely placed on the page. However, if you look closer, you will notice a simple, clean grid underneath. Effective use of negative space and animation resonate the spirit of the message, while the well placed typography flows the content from one page to another.

### 5.16 Summary

Page layouts give a website its look and feel. You derive the layout based on two major considerations: design concepts and information architecture (IA). The six steps to create IA are: (1) Define Goals (2) Define Audience (3) Create Content (4) Determine Concept (5)
TextMap and Navigation and (6) Visual Form.

With a good idea on IA and design themes, website designers can create page layouts that can present the various kinds of information to be places on a site in a two-dimensional arrangement that achieves the desired visual effects.

The fundamental steps in creating layouts begins with thumbnails and results in a comprehensive layout, or “comp” as is commonly referred to in design circles.

Typography is an important layout element. But, many find it difficult to master. It is important for a Web designer to become familiar with typography basics: anatomy of type, font styles and families, and spacing, etc. Detailed information help you select and manipulate type to achieve intended effects. For a Web page, type designation and spacing control are achieved with CSS declarations.

Grid systems help the relative positioning of elements on a page and are central to creating layouts. A well-designed grid can make messages clearer and contents easier to read. Creating grids that is flexible enough to fit on windows of different sizes and resolutions is a particular challenge for website design. The grid system in a layout is usually implemented by HTML tables.

**Exercises**

**Review Questions**

1. Analyze a current site of your choice. Go through the steps in IA and see if you can answer the following questions:

   - What do you think the goals of this site are?
   - Who is the target audience?
   - What is the concept for the site?
   - How would you describe the corporate identity?
   - How effective is the navigation for the site?
• How effective are the visuals for the site?

2. Choose an existing website. Analyze the corporate identity of that client. How is their logo used on the site? Is it consistent? Articulate in two sentence or less the image this company is trying to project. Is this image consistent with their other marketing materials such as billboards, print advertising or TV commercials? What is the corporate color?

Assignments

1. Review the steps in Section 5.3. Choose an existing web site which in your estimation has poor IA. Recreate the website’s IA. Be sure to follow all the steps indicated in this section. Be as through as possible in your investigation.

2. Refer to the first review question in this chapter. Use the same company for this question. Now that you’ve analyzed the IA for this company, see if you can understand the formal aspects of this site. Let’s begin with the grid. Print out several pages of this site. See if you can identify the underlying grid used in these layouts. Notice any consistencies or inconsistencies with the grid. Be sure to draw the actual lines on the printed website.

3. Create your own grid structure. Refer to the Williams-Sonomma examples in Figure 5.8, 5.9 and 5.10. Rework the existing, Williams-Sonoma grid. Change the column width, logo position, horizontal rules and margins. Create three new layouts based on this new grid; homepage, level 1 and level 2 pages of the site. Compare your new layout with the current site. Critique the two grid possibilities. Which one is more aesthetically pleasing? Which one is easier to read? Perhaps they’re simply variations and each is equally successful?

4. Choosing Typefaces Choose a topic for a single page of a site. This can be your personal site or one for your client. Try to find one or two typefaces which may look appropriate
for the subject matter. Remember to avoid typefaces which overpower the message. For example of that may be found in Figure 5.23. Let’s say that you’re designing a website for a company which produces rust prevention products and your job is to select an appropriate headline and subhead font. The first choice can be described as an “illustrative” because it literally mimics the idea of rust. Illustrative fonts need to be used sparingly because they can overpower the layout and tire the reader after several lines.

5. Navigation Variations Choose a website for this exercise which has poorly designed navigation system. Design 3 variations on the current navigation improving the placement of links, logo and text. Place these elements in effective.

6. Subtle Text Variations Take a large block of text which may appear on your personal website. Experiment with different leading and fonts. Try to select an appropriate type with both vertical and horizontal spacing which you think looks aesthetically pleasing. Increase leading by 1 point to see how it affects readability. Find suitable size headlines and subheads which look pleasing with existing text.