CREATING KILLER WEB SITES

The Art of Third Generation Site Design

David Siegel

As someone who is used to QuarkXPress (and therefore getting things to look just as I want them to), I have been frustrated by the design limitations imposed by HTML. Your tips are going to make a big difference in my personal site and in sites I am working on for others. *Mason R*

Aaaaaaaaaaah. I've spent more than three hours poking through nearly every link in your web pages. I thank you for your extraordinarily interesting, inspiring, rich, generous and multifaceted creation here on the Web. *Larry A*

Boy am I glad I just found your web site! After schlepping around to other resources for Web Page Developers, yours is the first I've found with the wealth of useful information I've been looking for. *Judy F*

Dave, I am in awe. Your web pages are definitely in the top the class, if not *the* top of the class. I could stay all day reading your material and looking at your page design. To review different web sites as part of my job, then to encounter your pages...life is almost too good. *Mora C*

Most of the Web books in existence are for severe technoweenies, *not* people who actually design stuff. Thank you. *Steve B*

I'm setting up a web publishing company myself, and I've been checking out everything the net has to offer. You probably have the best resources available for web development. *Mike M*

Your pages have provided *the best* and *most direct* explanation of HTML vagaries I've yet encountered, and believe me, I've been looking obsessively. *David R*

I've just finished looking at your site, and I can honestly say, after having looked at about 20,000 sites out there, that from a Web design point of view yours is *by far* the most innovative and best-designed site on the Web! *Eric G*

Contents

This book is dedicated to the surfers from around the world who visit my web sites regularly. Keep the cards and letters coming!

Introduction

The Book Site: http://www.killersites.com

New software emerges on the Web daily. Any CD-ROM included with this book would be hopelessly out of date before pressing.

My company, Studio Verso, maintains a site, http://www.killersites.com, which serves as an up-to-date companion reference to this book. There you will find all the files you need to follow these chapters from your own desktop, as well as pointers to books and software useful for making third-generation web sites.

It is generally referred to as the "Book Site" throughout the text. We hope it will serve you well!

About this book and its contents

This is not a first book on HTML. In addition to a good book on basic HTML, I recommended the following books on design principles:

The Form of the Book by Jan Tschichold, edited by Robert Bringhurst (Hartley & Marks, Vancouver, BC, 1991.)

The Visual Display of Quantitative Information by Edward R. Tufte (Graphics Press, Cheshire, CT, 1983.)

Envisioning Information by Edward R. Tufte (Graphics Press, Cheshire, CT, 1990.)

WE DON'T MAKE WEB SITES the way our parents did. The typical "Welcome to my home page," menu-driven, icon-encrusted model is fast being replaced by a model I call *third-generation site architecture*. Though third-generation sites rely heavily on today's browser technology, the difference is not technology *per se*. The difference is design. This is the first book about the conception, design, and construction of third-generation sites.

I have written this book in three parts. Part I covers the nuts and bolts of making third-generation sites, from theory and structure to implementation and tools. Part II contains case studies that take you through the design and construction of five third-generation sites, ranging from introductory to advanced. This section ends with a chapter that showcases several real-life design solutions from around the Web. Part III introduces Adobe Systems' Portable Document Format (PDF) with a primer and a case study. Finally, I discuss issues beyond HTML as we know it today, presenting a vision for the future of on-line design.

Although this book is more than a survey, it is far from complete. I have tried to cover graphic design as applied to the Web, and that necessarily involves a great deal of technical detail. There is much to know and learn about information design, form design, internationalization, and user interfaces. I touch on these subjects briefly. These are substantial subjects, too big for any one person to put into a single book.

We must start somewhere. The days of paper used frivolously are over. The Web is here to stay. I hope this book

helps designers to make the transition. In her essay entitled *Electronic Typography,* Jessica Helfand, a brilliant designer who has turned her modem into a design tool, wrote:

Here is the biggest contribution to communication technology to come out of the last decade, a global network linking some 50 million people worldwide, and designers – communication designers, no less – are nowhere to be seen.

With the freedom of the Web comes new responsibility; it will take new thinking to make the Web more visually rewarding as it continues to grow more interesting.
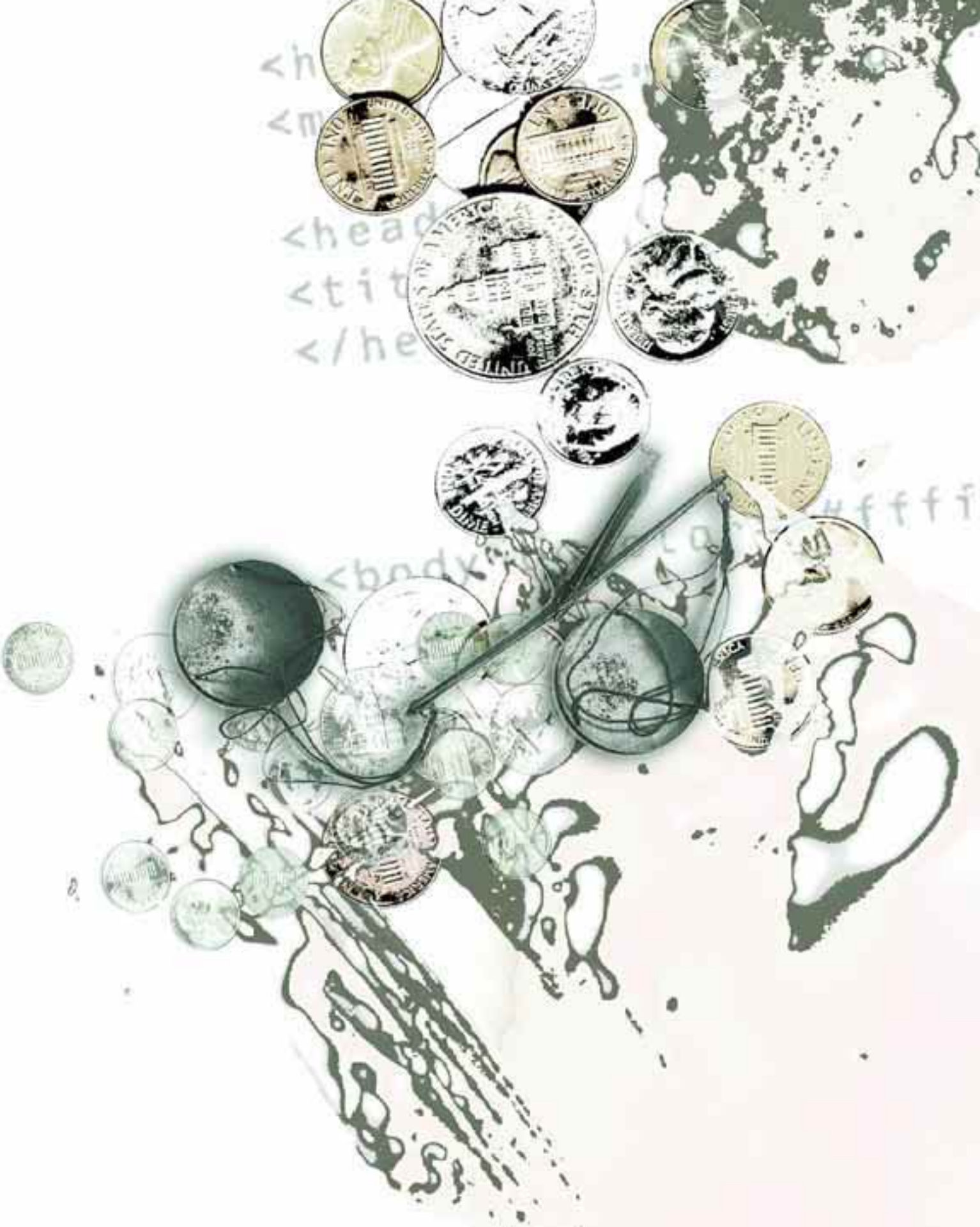
Form versus Function
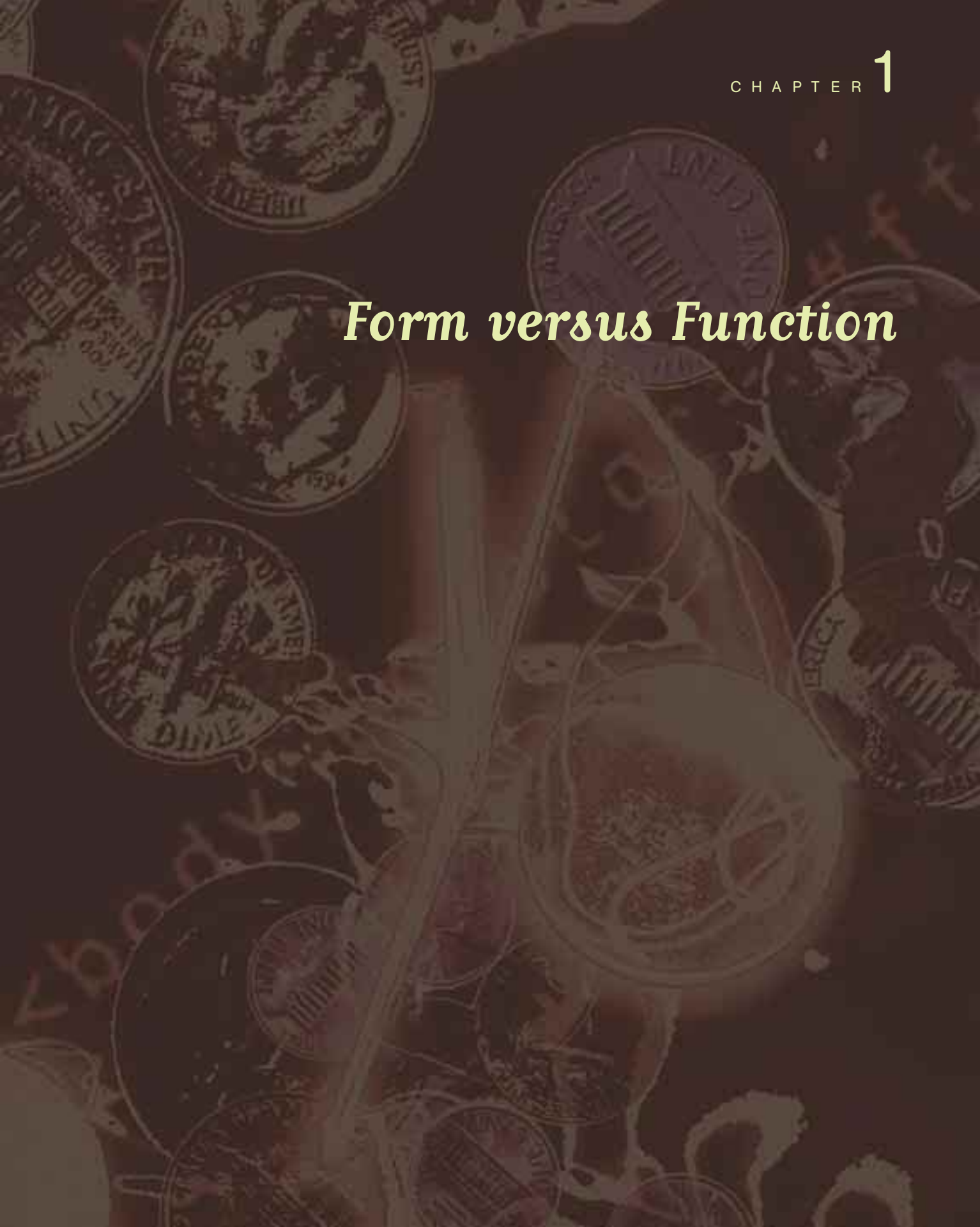
1.1 A, B Information (ABOVE) versus Design (BELOW): Who will win?

THE FIRST TIME I saw the same web page through two different browsers at the same time, a feeling of panic swept through me. Why should the pages look so different? I knew various systems had different color spaces and resolutions, but these programs actually presented the pages in different ways. As a graphic designer, how could I design web pages if I didn't know how they will look? Could I let browsers reinterpret my work as they wished?

As it turns out, that is precisely what the Framers of the Web had in mind. They saw browser differences as beneficial. Because every document is marked with structural *tags* (headlines, list items, captions, and others) they believed users should control the presentation of documents they see. HTML (**H**yper**T**ext **M**arkup **L**anguage) lets the visitor choose how her browser displays the page.

That's like telling the artist how to hold the brush! I wanted more control. It was either toss in the mouse and go back to print design, or jump into the Web and color outside the lines.

I threw my HTML book in the trash and started from scratch. I built my own tags and made pages the way I wanted them to look. I began using images to lay out pages in two dimensions, rather than one. When tables became available, I poured columns of text in them. I reduced the colors and made the file sizes small and – guess what? – people came! The numbers actually broke my access counter.

I started teaching others what I had discovered. Those of us who broke the rules challenged each other with every

new page we made. Like the painters of Russia's avant garde in the Twenties (Rodchenko, Malevich, Popova), we felt like pirates. I received flaming threats from people who said I didn't understand the Web, I was ruining the information, and I should go back where I came from because I didn't understand the power of the medium. I fired back with more pages. I made it clear we were not going to go away. If the Puritans wanted a fight, we would give them one [**1.1 A, B**].

Structural Markup Versus Layout

Print and on-line design are very different. As a print designer, I use programs like Adobe PageMaker and Adobe Illustrator – programs that write PostScript to describe pages. PostScript is a page-description language. As a web site designer, I am required to use HTML, rather than PostScript to make my sites (although this is changing, as you will see in Chapters 12 and 13). Designers want to make web pages with the same degree of control they have over paper. Designers must have layout capabilities if they are to design pages for artistic, entertainment or marketing purposes. To quote hypermedia visionary Ted Nelson:

Multimedia must be controlled by dictatorial artists with full say on the final cut.

HTML is a *structural markup language.* It has no ability to display a fixed-size page or a fixed-size text container. Standard HTML gives the author almost no formatting control. It is meant

mainly to describe the *structure* of information in a document, rather than its *appearance*. In theory, HTML browsers are responsible for handling display and layout. In practice, browsers make standard HTML look about as compelling as tax forms.

Structure has its uses. Structured documents can be fed into databases and searched easily. Their formatting can improve as HTML becomes more capable, enabling companies with large sets of documentation to enter information only once, then present it many different ways. This is of critical importance to people publishing large reference works, databases, searchable indexes, and documents for people with visual disabilities. As Tim Berners-Lee, the inventor of the Web, states:

Information to be published on the web must also be formatted according to the rules of the Web standards. These standardized formats assure that all Web users who want to read the material will be able to view it.

Different audiences have different needs. In a perfect world, markup and layout would be in balance. Instead, they are at polar extremes. Today, HTML is under heavy pressure from designers to be more visual, even as PostScript is being pushed to carry more structure.

What will happen in the future? Is the Web on a collision course with disaster? Will visually unimpaired designers battle against academic library scientists in a winner-take-all contest for the eyes and minds of web surfers everywhere?

21

Surfer Software

The browser wars are over – Netscape's *Navigator* and Microsoft's *Internet Explorer* are the only two left standing.

Now we have *feature wars*. Even though third-generation sites are distinguished by design, they will also be made differently when everyone has the next generation of browsers (Explorer 3.0 and Navigator 4.0).

I design web pages for Navigator and Explorer, Macintosh and Windows. That's not everyone, but it is 90 percent of the market. (I don't design sites for people who use other browsers or who surf with images turned off.) If a client needs a text-only version of a site, I'm happy to provide one, but third-generation site design is about competing for surfer mindshare and bringing in results.

Surfer Hardware

Consumers have come to the Web. According to Mary Meeker and Chris DePuy, authors of *The Internet Report*, "We believe consumers will begin connecting in earnest during 1996-1997, and that the rate of growth in American Internet connectivity will continue at a near 100% annual rate until the late 1990s."

Most consumers surf the Web on their lunch hour, which means they want to be entertained. They have 640 × 480 screens, but more are starting to log in with 800 × 600 screens, which will be the new entry-level standard. Consumer systems can see only 256 simultaneous colors, because they don't have sufficient VRAM to see more.

The answer to that question may surprise you. I'll return to it in the final chapter, after I've had a chance to make the case for using visual layout principles on your web pages today. For now, think visually! Throw away your old HTML tags. Replace them with the concepts and techniques that follow.

Summary

As the Web mutates and evolves, site designers find themselves in the middle of a roaring debate on standards and tools. Although this book mostly covers techniques of visual layout control, its real emphasis is on design. When the techniques become obsolete, good design endures.

Web Surfer Demographics	July 96	1998	2000	
Number of US web surfers	7.5 m	60 m	95 m	
	Percent	*Percent*	*Percent*	
Modem (dial-up) users:	70	90	?	
Surfers at 14.4KBPS	30	3	—	
Surfers at 28.8KBPS		40	80	10
Surfers with ISDN (128KBPS) or better	3	12	80	
Female (US)	30%			
Male (US)	70%			
Netscape Navigator Users	70%			
Internet Explorer Users	9%			
PC Users	68%			
Macintosh Users	20%			

Source: Interpolated from *The Internet Report*, by Mary Meeker and Chris DePuy (Harper Business, New York, 1996).
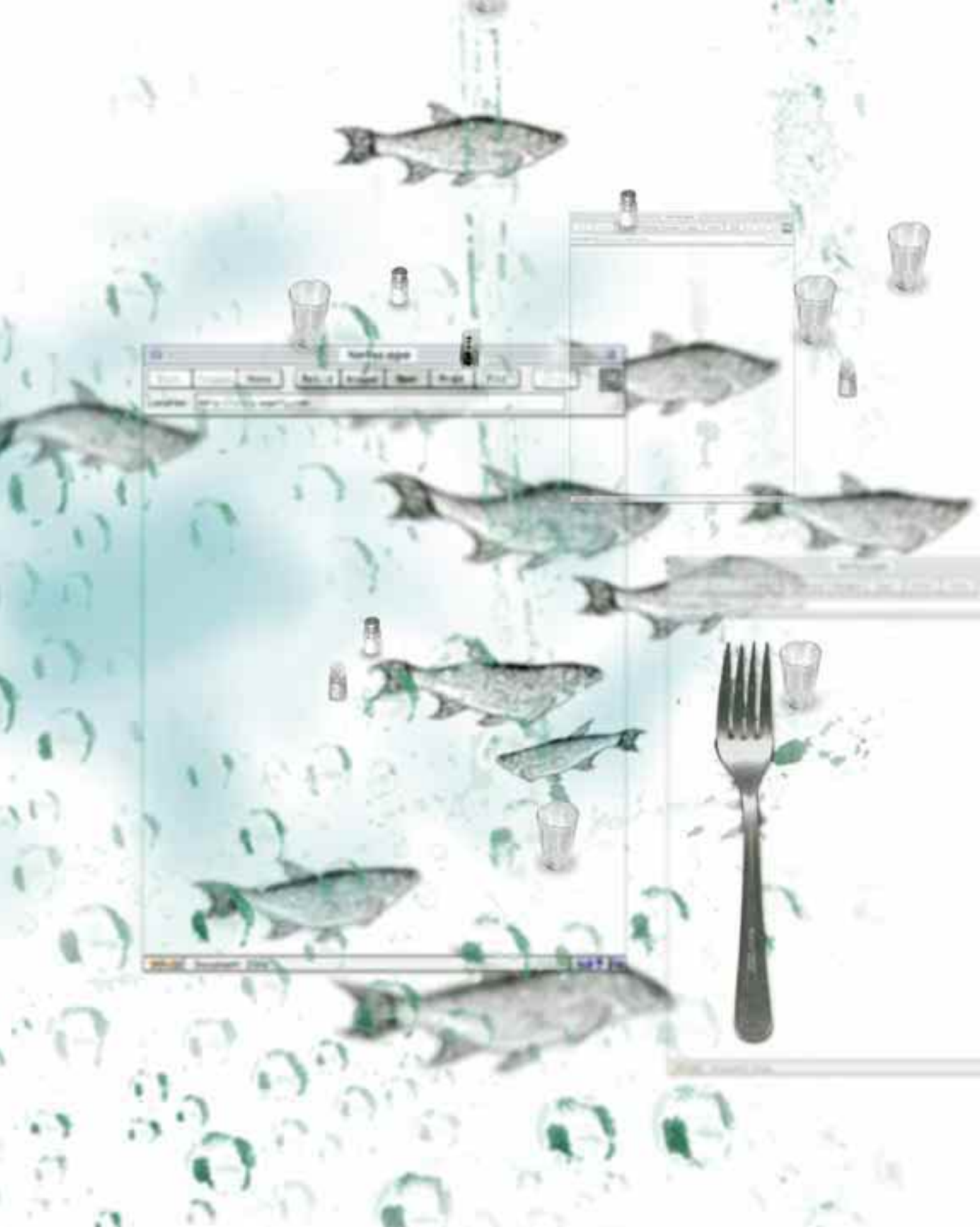
Third-Generation Sites
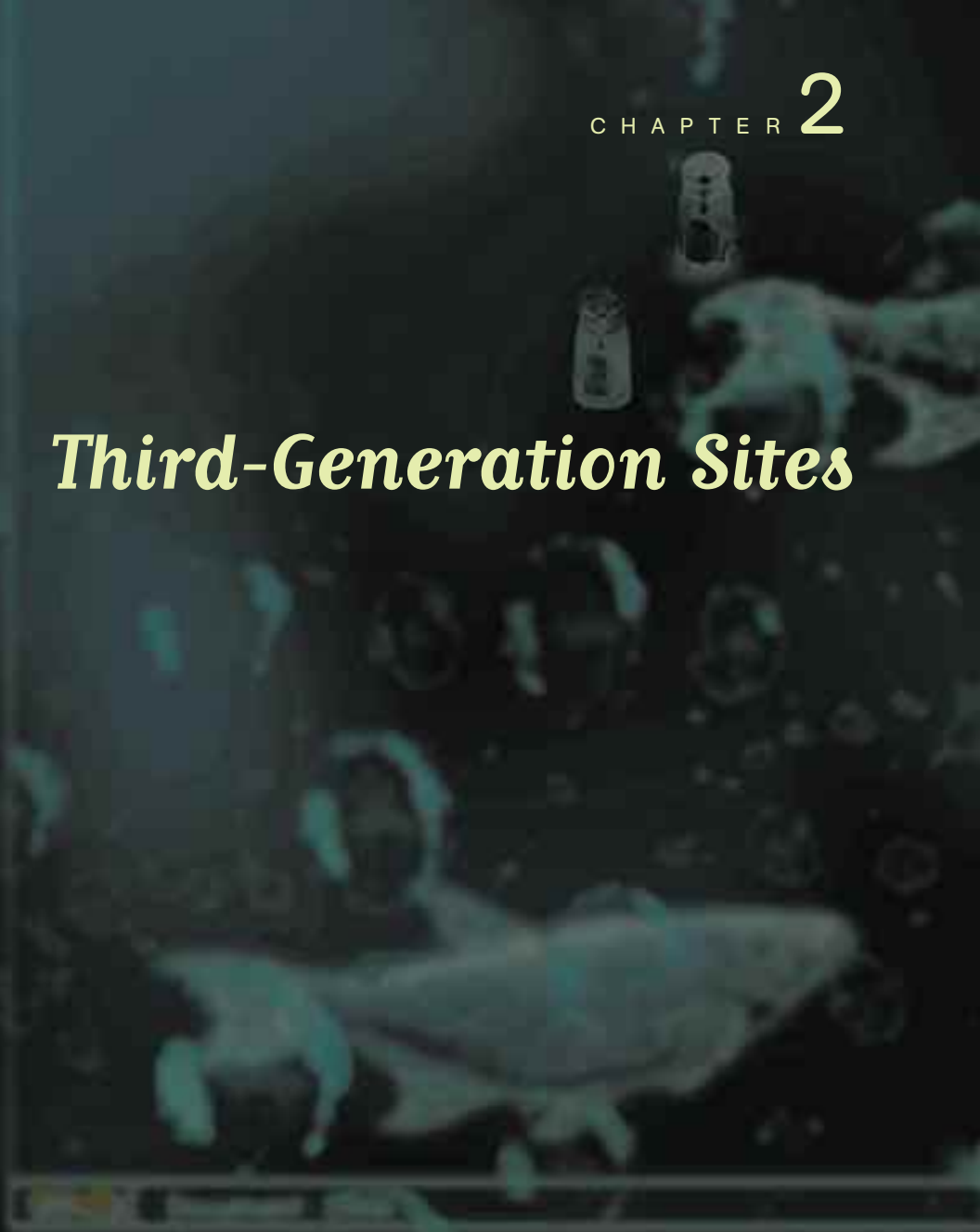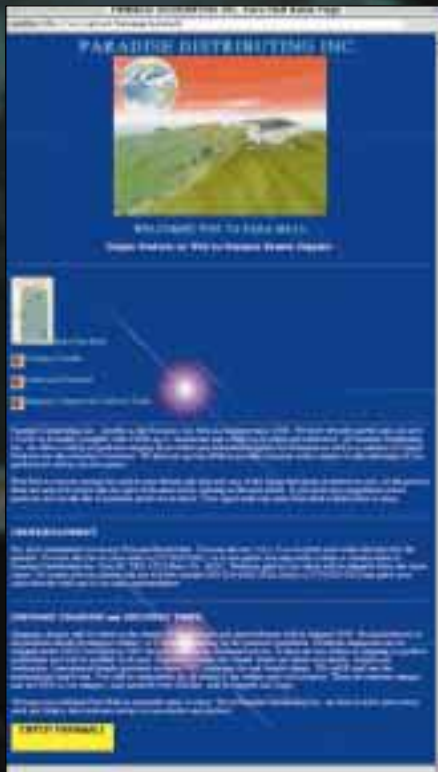
2.1 A, B First-generation sites: horizontal rules (ABOVE) and wall-to-wall text (BELOW) are the norm.

WHAT IS A THIRD-GENERATION web site? A third-generation site uses typographic and visual layout principles to describe a page in two dimensions. Third-generation site designers carefully specify the position and relationships of all elements on the page, retaining fine control of the layout.

Third-generation sites use metaphor and visual theme to entice and guide. They strive to make a site feel familiar and easy to navigate, with clear typography and high production values. This chapter covers the evolution of third-generation sites and discusses their structure in relation to well known models of consumer behavior. This chapter also addresses the design needs of information-based sites.

First-Generation Sites

The Framers of the Web designed HTML to look good on ASCII (text-only) terminals, black-and-white monitors, and low-resolution color displays. It needed to be functional, so scientists around the world could share their findings.

First-generation sites are linear. Looking at a typical first-generation page, you can see the restrictions imposed by slow modems and monochrome monitors. The page displays a top-to-bottom, left-to-right sequence of text and images, interspersed with carriage returns and other data-stream separators, like bullets and horizontal rules. All the initial HTML constructs were designed around this teletype model for laying out pages.

First-generation sites were gray [2.1 A]. Some sites had banners and

were well organized; most had edge-to-edge text that ran on for pages, separated by meaningless blank lines **[2.1 B]**. At best, they looked like slide presentations shown on a cement wall.

Second-Generation Sites

In the spring of 1995, Netscape announced a set of extensions to HTML. People played with the extensions and had fun with the <BLINK> tag. Second-generation sites began to sprout **[2.2 A-C]**. Second-generation sites are basically first-generation sites with icons replacing words, tiled images replacing the gray background, red and blue borders around the images, and banners replacing headlines. They use a top-down, home-page model to present a hierarchy of information **[2.3]**. Although first-generation sites still dominate by volume, second-generation sites are more popular because they have more graphics. Somewhere along the way, legibility went completely out the window.

Technology has always driven the Web. Exciting new capabilities appear weekly, urging designers to try them. Second-generation site design continues to be menu-, icon-, and technology-driven. These sites tend to follow the

A

B

C

2.2 A-C Second-generation sites push visitors by using simile and hierarchy.
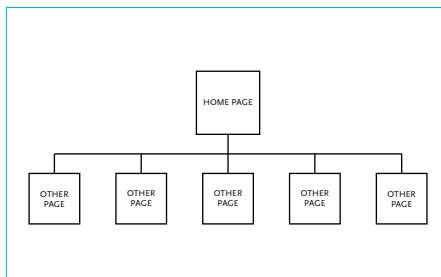
2.3 First- and second-generation sites start with a home page and a list of options.

The Restaurant Model

I use a restaurant metaphor when thinking about sites. You hear about a restaurant from an advertisement or a friend, or discover it while passing by. You check out the daily specials chalked on a board out front and smell the aroma in the doorway.

Once through the door, you make a quick stay-or-bail decision. In a popular restaurant, you might have to wait for a table. If you stay, someone shows you to a table and hands you the menu. You make your selection.

When the food arrives, you have no urge to rearrange the various items on the plate. The food and presentation are the creations of the chef. You sample the various items, skipping among them, mixing flavors and textures.

When you are finished, you have dessert, ask for the bill, and pay. You leave a tip, pick up the card next to the cashier, and maybe exchange a few pleasant words with the owner. Later, when you are hungry again, you return or you don't, based on the quality of that first experience.

2.4 A, B Early third-generation sites.

2.5 Third-generation sites pull people in the front door and guide them through.

home page model, where the first page you see is adorned with icons and 3-D graphical representations of buttons, windows, and pictures. At worst, noisy backgrounds and interminable waits for sound files make these sites unbearable. At best, they are nice white sites with color-coordinated icons.

Third-Generation Sites

A third-generation site is wrought by design, not technological competence. Third-generation sites give visitors a complete experience, from entry to exit. Design makes the difference. Creative people have made third-generation sites with all generations of graphical browsers **[2.4 A, B]**.

Third-generation sites pull visitors through using metaphor and well-known models of consumer psychology. Just as retailers spend a lot of time tuning their environments to the customers passing by, third-generation site designers spend hours and days making their pages enticing to the audience they seek. Third-generation sites form a complete experience – the more you explore, the more the entire picture of the site comes together. *Third-generation design turns a site from a menu into a meal.*

Browser version numbers do not compare! Netscape renamed a small upgrade to Navigator 2.0 as its 3.0 version, while Microsoft's Internet Explorer 3.0 is a giant step forward. Third-generation site *design methods* have nothing to do with third-generation browser versions. Navigator 4.0 will be the next major release from Netscape, promising to match a good portion of the Explorer 3.0 feature set and take the Web in new directions.

Site Structure

Millions of people surf the Web. You don't need all of them in your site. You want to reach a select group and turn window shoppers into customers who will take some action that benefits you both (send e-mail, order a product, give feedback, etc.). Telling people to order your products doesn't work. You must ask them nicely to come in and make them feel at home in your site. Most third-generation sites have an entry, a center area with a core page for exploration, and a well-defined exit **[2.5]**.

Third-generation sites pull visitors through by tantalizing them with something exciting on every page.

Entry

An entry to your site tells people where they are without serving your whole smorgasbord of delights at once **[2.6 A, B]**. More and more sites have *front doors* for just that purpose. A front door – also known as a *splash screen* – loads quickly and tells people what's going on inside. A good front door should be hard to walk away from. Present an image that grabs your audience and pulls them in.

Above all, splash screens should load quickly. Your first screen should take no more than 15 seconds to load at prevailing modem speeds – faster if possible. Present your visitors with a tedious download, and they'll be at

Case Study: Sobek Mountain Travels

Baiting the hook means giving something away. With the commercialization of the Web, fish food is nearly *de rigueur*.

One of the best gimmicks I've seen is the digital postcards of Sobek Mountain Travels. Visitors can send a postcard to a friend, and the site then sends that person an e-mail message saying there is a postcard waiting for them at the site.

They take the special number back to the site's virtual postmaster, who then shows them the picture and text from their friend who visited before. Once there, they are drawn into the rest of the site, where they learn about their tours and send postcards to all of *their* friends. This is a great example of how to give before you receive.

Yahoo! before your access counter can tell you what happened.

Fish Food

As people wander by your site, hold out a basket of goodies to tempt them. Gossip, news, sports scores, weather information, stock quotes, promotional sales, package-tracking services, pictures of Marilyn Monroe, free software, recipes, and sound files routinely lure potential audience members to third-generation sites.

This is what I call *fish food*. If you want to attract investors, put up either current stock prices or some lively, timely investment advice. If you're looking for dog owners, put up "The anatomy of the flea," or have a "name that breed" quiz. You want a gimmick that reaches out to the people you hope will form your community.

The technical term for fish food on the Web is *free stuff* [2.7 A-C]. Give visitors free stuff and a percentage of them will wander into your site. Use your imagination. Think of something your crowd would like to hear about, tell each other about, and go see. When people send your URL to their friends, you know you're serving something they want.

As any advertiser knows, there are no rules for getting people's attention. Use any means at your disposal, even

Java. Put up games, stunts, live video feeds, soap operas, a club for left-handers – anything that generates a buzz. Vandalize your own site, challenge another site to a contest, ask people to vote on something. Things like this work better than filling out forms and asking the search engines to list your site because of its great content.

Entry Tunnels

As visitors enter my sites, I like to give them the option of taking a little ride rather than going straight into the site. I call these rides *entry tunnels*. They help build anticipation **[2.8]** as people approach the heart of the site.

Entry tunnels are most appropriate for consumer sites **[2.9]**. Limit them to about four or five screens and make them entertaining. *Think of the Web as a cozy little town with a half-million restaurants.*

The Core Page

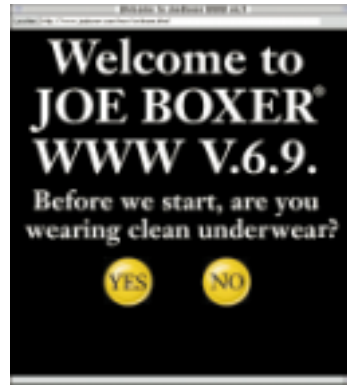The ultimate goal of most web sites

A

B

2.6 A, B A front door targets your audience.

A

B

C

2.7 A-C The goal of free stuff is to generate a buzz on the Web.

2.8 Entry tunnels build anticipation.

2.9 The Joe Boxer entry tunnel.

Barriers to Entry

Third-generation sites lure, seduce, coax. New visitors might not wait for a lengthy download on a high-bandwidth front page. Even the entertaining free stuff will irritate serious, repeat visitors if alternate points of entry or direct links to the core of the site aren't easily accessible. The best sites hook an audience before the audience even knows it.

In entry tunnels, it's no longer practical to ask people to register. Some entry tunnels say "Register here – free!" Who wants to register for free? If you really want people to register, you'll have to give them something major in return. Registration is a barrier to entry – be sure you need it before putting it in.

New approaches to registration will replace today's pleas for marketing information from surfers. The Web is advertising-driven. The advertisers will figure out a way to get what they want to know. In the future, your browser will automatically tell sites much more about you than they do today.

is to create a community. A good site pays off when people return again and again to purchase or participate. Core pages make this process enjoyable.

In contrast to the second-generation concept of a home page, third-generation sites can have either one or several core pages to organize and present the contents. Core pages direct and guide the visitor by providing links to relevant pages and other core pages. Core pages hold content while continuing to entice the visitor through the site **[2.10]**.

Don't be afraid to guide your audience. Give them choices, but also make suggestions. Give them lots of intrasite links and few external links. Put something interesting on every page.

Traditional home pages easily degenerate into an endless vertical list of links. Core pages use content to lure and tantalize. Use details of images and excerpts of text to guide your visitor – otherwise, your work remains buried behind flat, uninformative links.

Take the example of a mail-order site, where the goal is to get your user to call an 800 number or fill out an order form. A direct link to that order form or the 800 number itself should be available on almost every page. Most people won't click the first time they see it, but clicks are a function of exposure. Put the links to these final action pages everywhere, and your audience will get there when they are ready.

Exit

Paradoxically, a well-marked exit entices visitors to stay. Showing visitors the door to an exit tunnel informs them that this is the way out of the site. If they

come to an area that doesn't hold their interest, they shouldn't just type in "www.cirquedusoleil.com" and surf on. They should visit any areas that might be interesting before taking the exit.

Announcing the exit builds a sense of expectation, like announcing the names of the guests on the talk show at the beginning. It's worth your time to make an interesting exit. Cap their visit with a bang, but don't over-advertise it. Links to your exit should be subtle and constant, without enticing people to leave before seeing the rest of the show.

The exit page is a good place to ask for something from your visitors. You might want to have them fill out a form, call your 800 number, make a purchase, sign your mailing list, or take some other action. They are willing to work with you at this point, because you've rewarded their expectations.

The big finish may be as simple as a list of related sites on the Web, or it may be as fancy as an e-mail interface to an electronic billboard in Times Square (www.joeboxer.com).

Exits often include a chance to send comments or feedback, see a special image or movie, solve a puzzle, visit a random link, enter a drawing, etc. Free stuff can work as well at the end of a site as at the beginning.

Net Equity

If people talk about your site, if they come back often, if your metaphor starts a buzz and the front door is enticing, you will be on your way toward building *net equity*. Simply put, net equity is audience mindshare.

Change Is Good

You have a site. You want bookmarks. People don't need to bookmark the entrance to your site – they can probably remember that. But if you have a compelling core page, they just might bookmark it. The free stuff gets them there, but they come back regularly to the core of the site to see what's new.

If your site changes every month, it might as well be static. If it changes weekly, people might bookmark the pages with interesting things going on. If it changes daily, you could be in for

2.10 A core page with content on the left and a magazine metaphor on the right.

Case Study: Klutz Press

John Cassidy, owner of Klutz Press, asked me to create a web site for his company, a leading publisher of children's books. He wanted to put a few of the books' chapters on the Web, to illustrate how much fun they are. My original design for Klutz included a long entry tunnel, a dark room with a light switch, then a four-walled room. Each wall had a book on or near it. When you clicked on a book, you found yourself at the book's title page. Clicking on the title page took you to the book's table of contents, from which you could finally begin to play.

People loved the site, but few even reached the books. Most left after the entry tunnel. The book chapters were 14 clicks from the front door! Our redesign for Klutz is much more direct, without sacrificing fun.

As you arrive at the Klutz treehouse, an animated GIF automatically opens the door and beckons you in. On the second screen, you're in a room that contains a book, with its table of contents below. Selecting a chapter takes you right into the content on page three. We used a Perl script to randomize the books, so you get to a different book every time you enter the treehouse. This eliminated the need for another directory page – another layer of HTML with no content.

The moral of the story: put content on your pages as you guide your visitors.
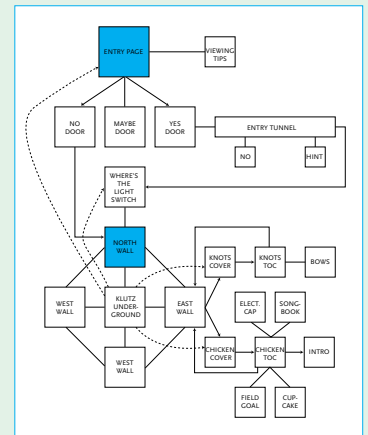
The first sequence (ABOVE) and the revised sequence (BELOW).

some big numbers on your access counters. Make sure to provide links from your active pages to more static areas, especially if you are trying to drive visitors to a particular page.

How many sites have "What's New!" on the front page? We don't need to know how to get to what's new. If it's new, and it's important, it should be in our faces. Put some content on your core page – don't bury it under a "What's New!" link.

Give Before You Receive

Third-generation sites hand out free samples, feed their visitors as they make their way through the site and re-ward them for coming back. The more you give, the more you must keep giv-ing **[2.11]**. If you keep at it, visitors will give back when they are ready.

Attention to the details of making a third-generation site will create a place where people like to hang out, and hanging out translates into net equity. The more net equity you build for your audience, the more you'll be able to ask for later.

Metaphor: Vehicle of Exploration

Third-generation sites often make effi-cient use of metaphor. A strong meta-phor can guide a visitor and glue a site together. Metaphors must be familiar, consistent, and appropriate for the modem speeds of the Web. Metaphors pull in visitors, make them feel at home while giving them features to explore. Examples of metaphors in-clude galleries, comic strips, television
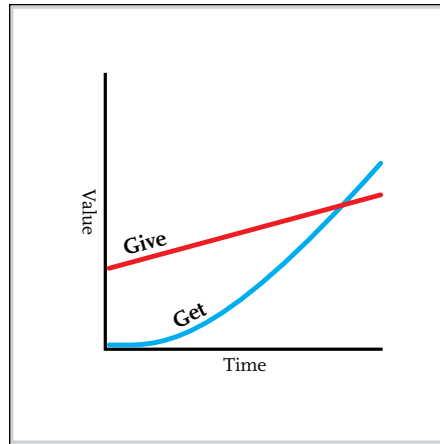
2.11 Only a constant effort to give people what they want can reap rewards later.

2.12 The FAO Schwarz site has a good mix of metaphor and content.

2.13 Some sites get caught up in their metaphors and forget the content.

2.14 A good metaphor is simple and well executed.

channels, magazines, tabloids, store environments, museums, postcard racks, amusement parks, inside things (computers, human body, buildings, ant farm, and so on), safaris, cities, and cupboards. These can be done well [2.12], or they can be overdone [2.13].

Metaphors are vehicles of exploration. Make it simple, consistent, and easy to get around. A good metaphor puts the switch where you expect it. A bad metaphor makes you learn a whole new set of commands to enter. Well-executed metaphors make it difficult to get lost.

Some sites try to present arcade-like interfaces or physical-space metaphors that rely heavily on 3-D graphics. You find yourself wandering down hallways, through doors, up staircases, and so on. These can work at high bandwidth or on CD-ROMs, but not with modems. Keep your metaphors light and effective.

Metaphors come in all shapes and sizes. This book should inspire you to seek new ways of representing your work, to think outside the constraints of HTML. Think of ways to make your visitors feel at home. Make the metaphor appropriate for your audience. Geologists might enjoy a virtual coal mine tour, but canaries won't.

The trick to presenting a successful metaphor on the Web is to couch it in HTML so that it loads quickly and yet doesn't look cheap. Key visual elements of your metaphor will have to be small enough to reinforce the metaphor over and over again.

Graphic designers are a largely untapped resource on the Web. They learn to use metaphor in school and

When is a Metaphor not a Metaphor?

When it's a simile. Simply replacing words with icons does not a metaphor make.

While international menu bars are very helpful to surfers from other countries, if your visitors can't speak English, they probably won't get much out of your site unless you also provide a translated version. Don't turn words into pictures and call it a site.

This is not to say you shouldn't have any symbols or icons on your site, or that you shouldn't be clever in presenting them. Icons play a supporting role in third-generation sites. Mixing metaphor and simile requires balance and restraint.

apply those lessons in making everything from business cards to TV commercials. Take advantage of them! Collaborate with a designer rather than trying to become one. Graphic designers who make sites should apply the visual lessons they've learned on paper to creating effective metaphors on the Web. Designers often check their visu-

al skills at the door when entering the Web. Don't fall into the trap of making second-generation sites at first, just because they are easier to make. Start with third-generation design and the skillful use of metaphor.

Once you choose a metaphor, stick with it. It may sound easy, but once you get going, there will be tempta-

A

B
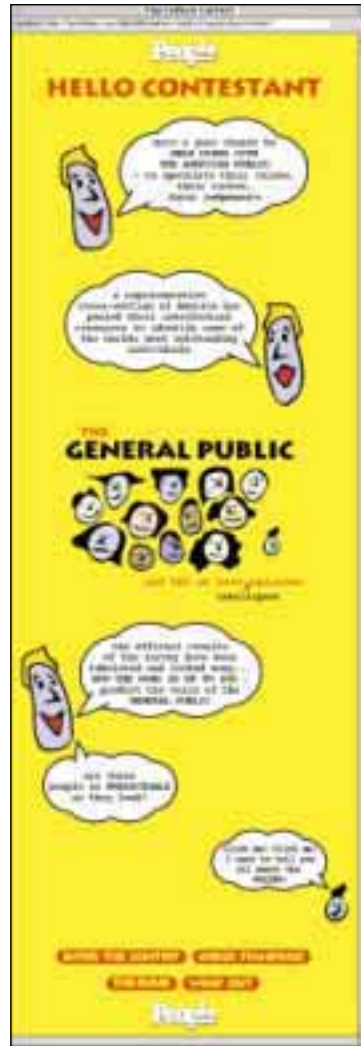
C

2.15 A-C Theme-based sites use consistent imagery and artwork.

tions. Keep it simple. A good metaphor helps select your audience. This proposal **[2.14]** for an entry was meant to be mysterious and inviting, setting up a metaphor for the entire site: outer space. (Compare this image to the previous space-town concept.)

Theme:
Interior Design for the Web

You don't need a metaphor to make a third-generation site. A consistent theme can work just as well. A theme can be visual or conceptual. Examples of themes: painterly, primitive, photographic, juvenile, art deco, typographic, futuristic, and so on. As with metaphor, themes can either enhance or get in the way **[2.15 A-C]**.

A theme can be almost anything. The best are subtle and consistent. Think of storefronts. Retail stores differentiate themselves by presenting a thematic environment. Some – particularly toy stores – use metaphors (castles, playhouses, and more), but most use colors, textures, lighting, and graphics to uniquely identify their stores.

Commercial interior designers know about theme. They make a space functional and interesting, not cartoonish or repetitive. They must create both tangible and intangible value, pleasing the senses while serving the needs of the business.

Thematic sites are more difficult to create than you might think. There is a great temptation to throw in more of everything – sound, animation, fonts, graphics – leading to clutter and confusion. Using quality photography, for

example, can make a difference on a theme-based site. Using the few colors available to site designers is a big challenge. Defining a subset of those colors to create an individual palette that expresses a particular identity is nearly impossible. A good thematic site is an exercise in subtlety and consistency.

Information-Based Sites

Many sites are not geared toward consumers. In the information realm, sites must satisfy impatient, directed visitors. These sites can't afford to put too much glitter in front of the information. Nevertheless, they can be compelling without using a lot of icons and banners.

Most information-based sites present endless pages of text and bulleted lists, with a predictable home page up front (NEWS | ABOUT US | CATALOG | FAQ | HELP). The best have a search engine enabling visitors to find things immediately, but if a customer doesn't know exactly what she wants, she is lost.

Information-based sites come in all shapes and sizes, but they rarely engage. The best way to overcome the information-overload approach is to establish a simple theme and keep opening doors in front of people. Put content, new information, and navigation on every page. Give them small opportunities to take unexpected turns or see new sections. Use a light theme or metaphor to make the experience worth coming back to.

Information-based sites must be balanced. They must pull in new people, while giving the regulars an easy

The Currency of the Web

A **hit** is any file downloaded. A hit can be a page of text, an in-line graphic, or a downloadable movie or sound file. Thus, if you have a page with ten pictures on it, a person coming to that page will generate 11 hits (if her browser can see images). Many people confuse hits with accesses or visitors, which makes 10,000 of them around noontime sound as though their site is packed with people (when it really isn't). Hits are the pennies of the Web.

An **access** is an entire page served. Accesses (also called *pagehits*) are the smallest unit to track, because they can help you determine where people go in your site.

A **visitor** is the true one-dollar bill of the Web. Unfortunately, *unique visitors* are more difficult to define. That's why there's so much incentive to get visitors to register – people are anonymous as they view your site. For the most part, I simply make educated guesses about the number of unique visitors to my sites. Programs are now available to help webmasters track visitors going through their sites.

A **repeat visitor** is the ten-dollar bill of the Web. If a visitor bookmarks your site, it means she is willing to come back. In the mail-order business, such a person is called a *responder*.

If people order things on your site, they turn from visitors into **customers,** the ultimate goal. A customer is the highest level a visitor can attain. A good web site strives to turn a small percentage of random surfers into customers.

Webmasters & Webmistresses

A webmaster or webmistress is the person responsible for keeping the server running properly. Site designers are not webmasters any more than ship designers are first mates.

2.16 StockCenter entices people to see stock indexes and learn about their product while showing stock prices.

2.17 InfoSeek uses the left margin to give suggestions.

way to get exactly what they came for.

One example is the site I designed for StockCenter **[2.16]**. People go right to the information they want. From the left margin, they can choose from a number of possibilities.

InfoSeek **[2.17]**, a popular search engine, uses tables to present the search results on the main area of the page, along with a selection of recommendations in the left margin. This added value to an information-based site keeps surfers coming back.

Information-based sites must offer both browse and search capabilities. Regular customers need a page they can bookmark, preferably listing the features of the site and providing the shortest path to any given page. There should be a search window, or at least a button to a search page, on every page in the site.

One way to present a lot of continually changing data is to use Netscape's frames feature. Frames can help, espe-

cially for presenting large sets of pages that users should compare. Information-based sites do not need frames, but they are candidates for frames. For now, a manufacturer who wants to present hundreds of products side by side might consider frames. I don't think frames are the only solution (at this writing, no web-wide search engine uses them), but designers of sites with large, frequently changing data sets should weigh the advantages and disadvantages of frames. Internet Explorer's *floating frames* might also help on sites like this. *(See "Frames and Framesets" in Chapter 4 for more information on frames.)*

Dynamic sites will be the norm in the information realm. Rather than bookmarking a static page, frequent users will fill out a form telling the site about their needs. The site will go to work for them, sending them e-mail messages when new items of interest arise, providing a custom, made-

on-demand page just for them when they log in, and generally keeping their interests in mind as they cruise the site. A good dynamic site presents opportunities to learn new things and see new offerings while trying to meet 90 percent of the frequent surfer's needs on the first two pages.

Design makes an even bigger difference on information-based sites. Lessons from information design, interactive design, architecture, and user interface design will help third-generation site designers create new paradigms for navigating large data sets. While the rest of this book discusses more consumer-based models of design, many of the principles of third-generation site design apply to information-based sites as well.

Summary

People tend to surf with their shortest attention spans turned on. Think of the center of your site as the kitchen, where you're serving meals all day. Once visitors have been lured in by the aroma of good food, they start prowling around, opening cupboards, and raiding your cookie jar. A commitment to fresh daily content is often the best way to attract and keep a crowd. Invite the critics over for a taste. Keep handing out samples and don't present the bill until after the meal. Cook up a good site, and you'll always have a line of hungry customers waiting to get in your front door.

As Joseph Squier, author of The Place, states: "Independent of medium or tool or technique, there are timeless aspects of art that endure. Artists communicate."

Client/Server Computing

Servers are computers connected to the Internet around the clock, serving information. *Clients* are programs, like Netscape (a browser) and Eudora (a mail reader). When I use the word *client*, I always mean a *program* that resides on your computer as you use the Internet.

An Internet service provider (ISP) gives you access to the Internet. A server sits somewhere and sends files to everyone. You don't need to locate your site on the server of your service provider. Your site can be *hosted* on a server in Zurich, but your ISP should be a local company that gives you good access to the Internet for a good price.

The View-Info Command

Netscape's browser has a little-used feature that gives you information about the pages you see.

While most people are used to the View Source command that shows the HTML of any file you see on the Web, Netscape Navigator will also give you quantitative statistics for any given image. This includes the size of each image, both when compressed (content length), and also in memory (decoded size). Add up all the numbers to see how large your page really is.
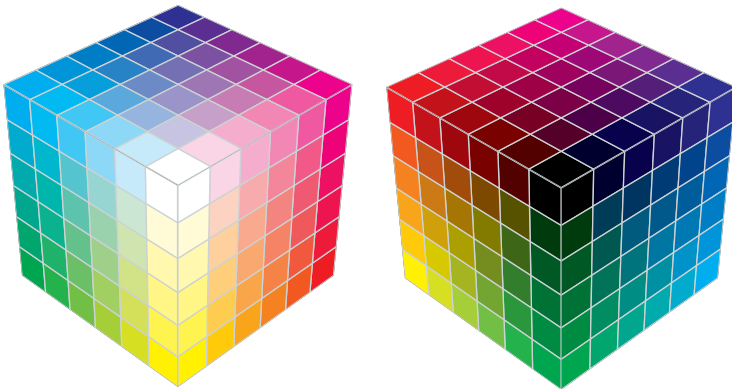
3.4 A, B The color cube. See this image on the Book Site.

The internal colors of the cube are the four evenly spaced colors between each corner, for a total of 216. This simple approach results in a *dithering palette.* Other palettes might have been more useful – this one has only four internal gray values, for example – but the color cube is easy to program. Most browsers now use Netscape's color cube.

On a monitor set for 8 bits, the browser automatically dithers the image using this palette. Setting your monitor to 256 colors will show you what your images will look like to surfers with low-end systems.

The Color Cube

The Netscape browser has its own palette, called *the color cube,* for displaying images on 256-color monitors on all platforms **[3.4 A, B]**.

Netscape's six-sided color cube comprises 216 colors. Why not the full 256? Windows needs 20 colors for itself, other programs (and wallpaper) use another 20, and the balance – 216 colors – is available to the browser.

At the corners of the cube are all eight possible combinations of 255 and 0 (full on and full off) in RGB space **[3.5 A-C]**.

Pick from the Cube

Load the color cube's palette into Photoshop's color picker. To create the palette, bring the color cube graphic into Photoshop, export the color table as "Color Cube Palette," and use the Load Swatches feature. (See the Book Site to pick up the color cube graphic.)
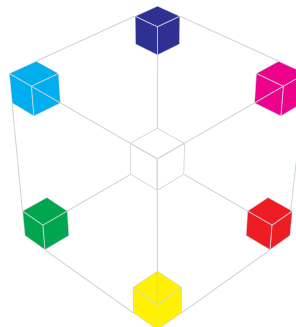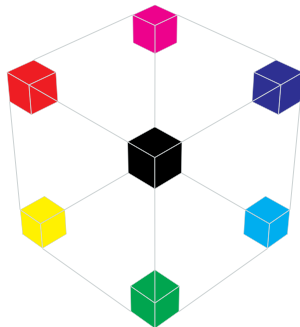
H	E	X		R	G	B
FF	FF	FF		255	255	255
FF	FF	0		255	255	0
0	FF	FF		0	255	255
0	FF	0		0	255	0
FF	0	0		255	0	0
FF	0	FF		255	0	255
0	0	FF		0	0	255
0	0	0		0	0	0

3.5 A-C The corners of the color cube.

The bottom line: Use these 216 colors in your images or they will be used for you.

Creating Images with the Color Cube

When using Painter or Photoshop to generate images, I try to use colors from the color cube. The general rule is to make sure large areas use colors from the cube, to prevent noticeable dithering.

As you work in layers, Photoshop or Painter will automatically anti-alias your image, creating lots of intermediate colors that aren't in the cube (see the next section, "Anti-Aliasing"). For now, use colors from the cube for large areas – until the browser companies change the rules again [3.6].

Mapping Images to the Color Cube

When you have no control over how an image is created, you'll have to convert it to the color cube's palette. To ensure consistency when displayed, convert images early in the process to keep yourself focused on the end result.

There are several different methods for pouring your images into the cube. A program like DeBabelizer offers the most choices. In Photoshop, use the Custom palette feature and load the color cube as the custom palette. (As of this writing, there is no way to save the color cube in Photoshop permanently. I keep copies of the cube in various places on my hard disk and load them as necessary. In DeBabelizer, you can either give it a name or use it as the super palette.)

The big choice here is *to dither or not to dither*. It's always best not to dither if possible. Dithering makes many

Dithering

Dithering is the process of using colors from the available palette to make up for colors not in the cube. Main colors are sprinkled throughout to simulate intermediate colors. If you need a color one-third of the way from color A to color B, use 33 percent pixels of color B sprinkled on top of color A. In practice, it works fairly well for photographic images, but not for most other types. Try to avoid dithering as much as possible; it makes your files much bigger.

A   B   C
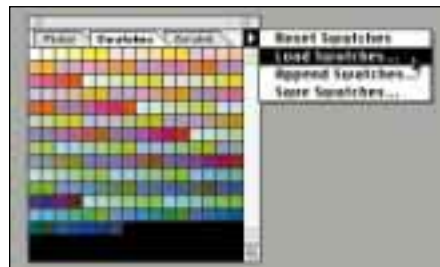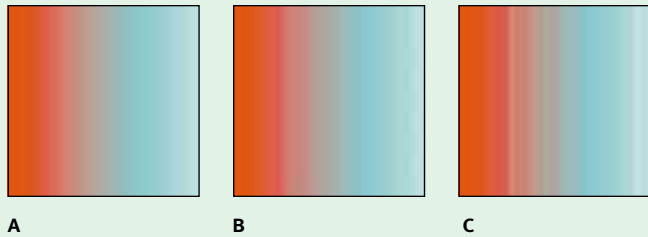
3.6 Loading the color cube into the picker.

Hex Values

In HTML, the <HEAD> tag requires hex values for specifying colors for background color, text color, and so on. Hex numbers are a way of specifying values from 0-15 with a single character (from 0-F). Use any 2-digit combination (for red, green, and blue) of "HHH" – where H is in the set {00, 33, 66, 99, CC, FF}. A light blue is "#CCCCFF" and a middle orange is "#996600". As long as you use only these values, any combination will produce a color-cube color. If you're aiming for a broad audience, you should use only these combinations.

47

The JPEG Format

JPEG (the common name for the raster image format defined by the Joint Photographic Experts Group) is the best way to compress photographic images. Because of the overhead involved, it doesn't work well for small images or line art. To reduce file size, JPEG separates the brightness information from the color hues. It essentially keeps a good copy of the black-and-white version of an image, to which your eyes are sensitive, and throws away most of the subtle color differences that you can't distinguish. Rather than doing it line by line, as GIF does, JPEG breaks the image into zones.

Compression

JPEG is a *lossy* process – information is always lost in the compression. Once compressed with JPEG, even using the highest quality settings, the image will not be the same as the original RGB version. It might look the same, because your eyes can't tell the difference, but the file is smaller. At higher compression ratios (lower quality settings), the image is noticeably different, whereas high-quality JPEGs are used as substitutes for the original. Use low-quality JPEGs on the Web, because they compress so well.

Compression ratios for JPEG range typically from 10:1 to 100:1, depending

A1 3-bit GIF, dither 2,985 bytes

A2 4-bit GIF, dither 3,728 bytes

A3 5-bit GIF, dither 4,624 bytes

A4 6-bit GIF, dither 5,486 bytes

B1 3-bit GIF, no dither 2,331 bytes

B2 4-bit GIF, no dither 3,181 bytes

B3 5-bit GIF, no dither 3969 bytes

B4 6-bit GIF, no dither 4,964 bytes

C1 JPEG 25% 2,351 bytes

C2 JPEG 50% 3,267 bytes

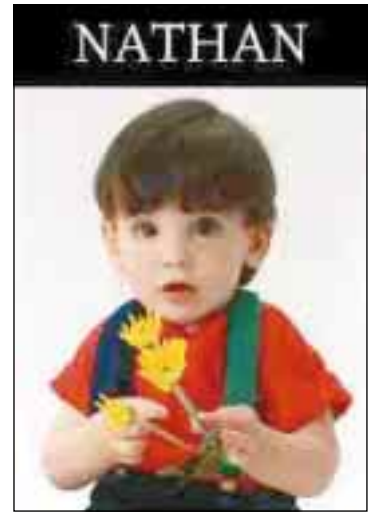C3 JPEG 75% 4,566 bytes

C4 JPEG 100% 15,817 bytes

3.18 A-F Images A-C are 92 × 132 pixels; D-F (facing page) are 166 × 238. JPEG files have several kilobytes of extra information, called *overhead*, that make them unsuitable for small pictures.

D1 3-bit GIF; dither; 7,446 bytes

E1 3-bit GIF; no dither; 5,097 bytes

F1 JPEG 25%; 4,264 bytes

D2 5-bit GIF; dither; 11,498 bytes

E2 5-bit GIF; no dither; 9,692 bytes

F2 JPEG 50%; 6,107 bytes

on the quality setting. The greater the compression, the smaller (and lossier) the file will be. In general, sharper images make larger files. Blurry pictures download quickly. Because of the overhead in the JPEG format, GIFs are a better choice for small pictures. I usually don't go to JPEG until my images are larger than about 100 × 100 pixels [**3.18 A-F**].

Always start with the lowest-quality JPEG setting you can, throwing away the most information and reducing file size as much as possible. If your program says: Worst, Okay, and Best, choose the Worst option. You'll be surprised at how acceptable low-quality JPEGS are.

4.4 A, B Absolute widths keep columns from collapsing, while relative widths move with the browser window.

```
<TABLE BORDER="2">
 <TR>
  <TD WIDTH="100">
   Width=100
  </TD>
  <TD VALIGN=MIDDLE WIDTH="200">
   Width=200 (pixels)
  </TD>
 </TR>
</TABLE>
```

```
<TABLE BORDER="2">
 <TR>
  <TD WIDTH="25%">
   Width=25%
  </TD>
  <TD VALIGN=MIDDLE WIDTH="75%">
   Width=75%
  </TD>
 </TR>
</TABLE>
```

4.5 A, B HTML for absolute (LEFT) and relative (RIGHT) column widths.


4.6 A, B Using cellspacing for white space assures a noncollapsible gutter. (Visit the page yourself and view the Document Source to see the HTML.)

70

infinite bottoms. You can specify the height, and you can use `<TD VALIGN= bottom>`, but then everything goes to the bottom. In general, tables grow and shrink to accommodate their contents.

To control the *width* of your table cells, use either relative or absolute widths [**4.4 A, B**]. To specify *relative cell widths,* use `<TD WIDTH=x%>`, where *x* is the percentage of the width of the table containing the cell. Relative cell widths give you flexible tables that reflow when the window size changes. *Absolute cell widths* (use `<TD WIDTH= x>`) specify the width in pixels. This cell does not change size when the window does. Make the following table with both relative and absolute numbers [**4.5 A, B**], then adjust the browser window width. *(You can see this on the Book Site, but it's more instructive to make your own.)*

Use `CELLSPACING` and `CELLPADDING` to guarantee white space [4.6 A, B]. As far as the third-generation site designer is concerned, cellpadding and cellspacing are identical. *Cellpadding* describes the distance between the contents of a cell and its walls. *Cellspacing* describes the amount of space surrounding each cell. As implemented by Netscape, these tags apply to horizontal and vertical distances simultaneously. I often use cellspacing to create a *gutter,* where I want to separate two columns by a given amount. This is especially useful when designing forms *(See Chapter 8, "A Storefront")*.

Use table cells to mortice images horizontally and vertically [4.7]. If you put two images next to each other, there
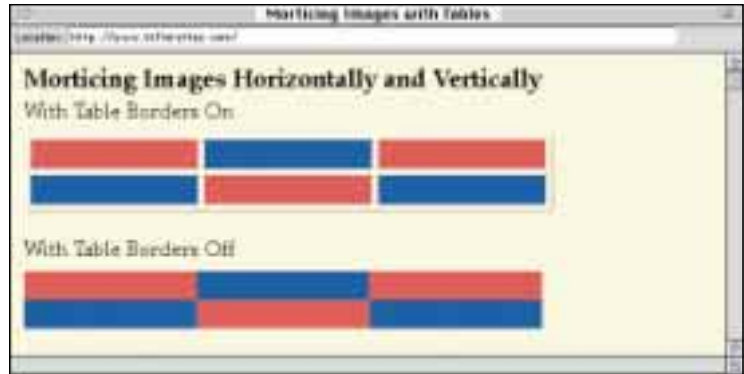
4.7 Table rows butt right on top of each other.

```
<TABLE BORDER="0" CELLPADDING="0"
 CELLSPACING="0" WIDTH="450">
 <TR>
  <TD><IMG SRC="resources/dot_red.gif"
   ALIGN=LEFT WIDTH="150" HEIGHT="50"
   BORDER="0"></TD>
  <TD><IMG SRC="resources/dot_blue.gif"
   ALIGN=LEFT WIDTH="150" HEIGHT="50"
   BORDER="0"></TD>
  <TD><IMG SRC="resources/dot_red.gif"
   ALIGN=LEFT WIDTH="150" HEIGHT="50"
   BORDER="0"></TD>
 </TR>
 <TR>
  <TD><IMG SRC="resources/dot_blue.gif"
   ALIGN=LEFT WIDTH="150" HEIGHT="50"
   BORDER="0"></TD>
  <TD><IMG SRC="resources/dot_red.gif"
   ALIGN=LEFT WIDTH="150" HEIGHT="50"
   BORDER="0"></TD>
  <TD><IMG SRC="resources/dot_blue.gif"
   ALIGN=LEFT WIDTH="150" HEIGHT="50"
   BORDER="0"></TD>
 </TR>
</TABLE>
```

4.8 This code generates the table at the bottom of illustration 4.7.

71

Layer Management

Always leave the background layer of a Photoshop document a single color. Everything else goes into separate layers for easy anti-aliasing and flexibility.

Many tools anti-alias by default, which can either be beneficial or horrific. For instance, ellipses should be anti-aliased, while rectangles should not.

In Photoshop, I like to use the HSB rather than the RGB slider, because it's easier to make colors that go together.

I make sure I'm in RGB mode, because CMYK isn't for display screens, and I always use pixels, rather than inches, as my standard unit of measurement.

6.2 The blue area on the left tiles down to become the vertical bar. Make the file 1200 pixels wide overall, so that it doesn't tile again to the right.
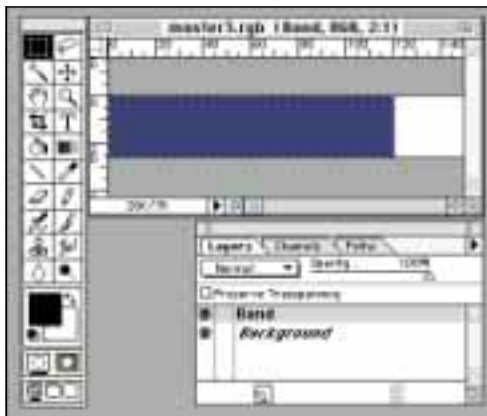
6.3 The tiling background as it looks in the browser.
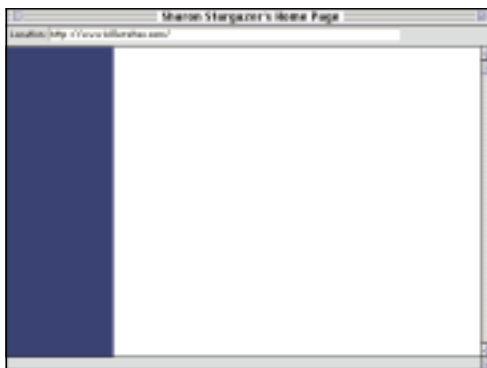
following code into a new file, open it with Netscape, and – voilà! **[6.3]**

```
<HTML>
<HEAD><TITLE>
Sharon's Home Page
</TITLE></HEAD>
<BODY BGCOLOR="#FFFFFF"
 BACKGROUND="spine.gif">
</BODY></HTML>
```

A Third-Generation Banner

I'd like to put a red circle in the foreground, centered on the border between blue and white. This breaks the plane of the blue bar, causing it to stand out above all other elements. Visually, it will become the "frontmost" image, drawing the eye like a lit match. I am going to put Sharon's name over it, so you know right away whose page it is. (To me, this is better than `<H1>`, but that's just my opinion.) I'll start with a circle 80 pixels in diameter, two-thirds the width of the blue band. This should give me a balanced look – not too scrawny, yet not overpowering. I open a new Photoshop document, 120 3 120 pixels – a bit larger than the circle. On a new layer, I draw a red circle using the ellipse tool, which is anti-aliased by default. Anti-aliasing makes the edges of the circle look smooth **[6.4 A, B]**.

I export the circle as a GIF, using the GIF89a Export plug-in's eyedropper to designate pure white as the transparent color. I view this GIF at various positions, using the image's **HSPACE** and **VSPACE** arguments to space them over, rather than bothering with extra single-pixel GIFs. **[6.5, 6.6]**

What's going on? Because GIF supports only one level of transparen-

106
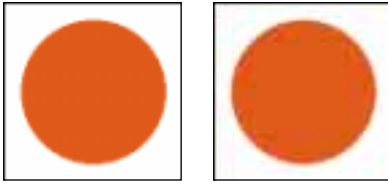
6.4 A, B A circle, aliased (LEFT) and anti-aliased (RIGHT).
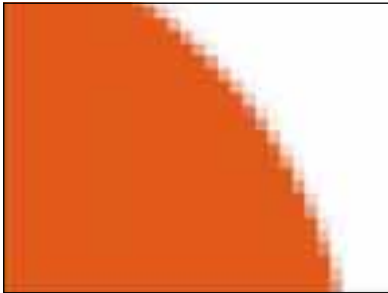
6.7 This image contains many colors to make the edge look smooth.
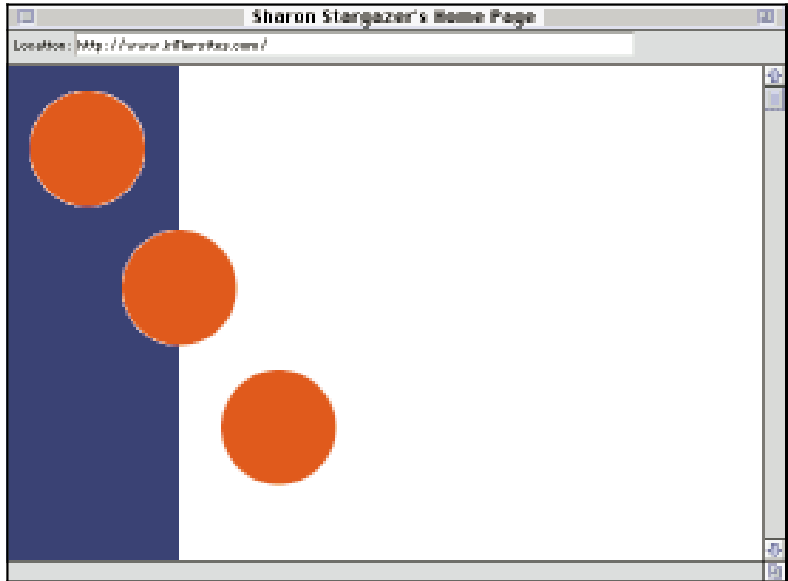

6.6 The use of anti-aliased foreground images requires precise alignment over backgrounds with major color changes.

cy, all the near-white pixels (which are actually shades of pink) make the shape look good against white, but not against blue. I anti-aliased the circle against a background of solid white, but half the background is blue [6.7]. I'm in trouble.

To anti-alias the gif properly, I add a new layer, draw a blue rectangle, and position the circle against it [6.8].

Note: These anti-aliasing problems wouldn't happen if we had 256 levels (8 bits) of transparency, but with GIFs, we can designate only one color (1 level) as transparent.

```
<HTML>
<HEAD><TITLE>Sharon Stargazer's Home Page</TITLE></HEAD>
<!-- Page Parameters -->
<BODY BGCOLOR="#FFFFFF" BACKGROUND="spine.gif">

<!-- Begin Red Circle Placement -->
<IMG VSPACE=15 HSPACE=20 WIDTH=80 HEIGHT=80
  SRC="redcircle.gif"><BR>
<IMG VSPACE=15 HSPACE=80 WIDTH=80 HEIGHT=80
  SRC="redcircle.gif"><BR>
<IMG VSPACE=15 HSPACE=140 WIDTH=80 HEIGHT=80
  SRC="redcircle.gif">
<!-- End Red Circle Placement -->

</BODY></HTML>
```

6.5 Note the use of HSPACE to control the horizontal position of the circle.

7.15 A nicely designed business card can make a good entry to your site.

I get so much positive mail from people who see it. The blueprint metaphor takes you out of the Web and into a different, yet familiar world. I achieved it by splitting images and locking them into tables. The page loads quickly yet feels expansive. Splitting images can help get around the limitations of tables when mixing text and image.

Notice I don't have a "NEWS" section. Because the table takes some seconds to load, I usually put in some items of interest right at the top of the page, so people have something to read as it loads. Although I haven't shown it, this is an effective way to tell people what's new while they are waiting and you have their attention. As the saying goes: "content is king!"

7.14 [LEFT] Now that I'm pleased with the way the site looks, I can turn off my borders and upload the files to my server.

Application

One of Matthew Butterick's designs for his on-line 'zine, *Dex,* is a great application of the main technique explained in this chapter: how to cut a large image up to make a killer home page **[7.16]**. Matthew gives the scrolling visitor a visual treat by creating the illusion of a seamless display of playing cards. Visit the site and you'll see how effective it is.

7.16 Table borders turned on: *Dex* magic secrets – revealed at last!

Deadly Sin Number Three

Background Images That Interfere

Backgrounds on the Web have reached epidemic proportions. Wallpaper is nice, but reading the handwriting on the wall can cause damage to the retina.

People use background images because they add a "theme" to the page, or because they "fill up all that unused space." It's the kitchen-sink school of page design, which often escalates to homicidal (not killer) site design. Backgrounds do more damage to web pages than almost anything else.

Thoughtless designers get carried away, the pixels fly, and surfers get hurt. The only good background is a solid or nearly solid color: gift-wrap makes bad stationery.

Site diagram

Diagram labels:

- INDEX PAGE
- TITLE PAGE / NAV BAR
- PREAMBLE PAGE / NAV BAR
- PERL SCRIPT CHOOSES FROM 3 DIFFERENT PHOTO SETS
- 1 POSITIVE IMAGE
- 1 NEG. IMAGE
- 36 POS.
- 36 NEG.
- EACH PHOTO SET CONTAINS 36 PHOTOS AND THEIR NEGATIVE IMAGES.
- NAV BAR / FEEDBACK PAGE
- EVERY PAGE WITH A NAVIGATION BAR LINKS TO ALL OTHER PAGES WITH NAVIGATION BARS.
- NAV BAR / CLIENTS
- NAV BAR / BIOGRAPHY
- NAV BAR / COMMENTS PAGE
- PHOTO LINKS PAGE

DOUG MENUEZ is one of the country's most versatile and interesting photographers. He was a photojournalist for *Time* and *Newsweek* during the '70s and '80s. He's done editorial and documentary work, as well as fashion and advertising photography. For ten years he's been shooting life on the digital frontier, documenting people in high-tech companies across the globe.

Doug came to my studio in search of a cutting-edge, provocative site, combining words and images to bring his view of the digital revolution to surfers interested in photography and photojournalism. He wanted to show the disparity and similarities between the digital *haves* and *have-nots* – the people who use whiz-bang technological innovations and the people who make them in far-away places.

Doug's primary concern was image quality. We decided to optimize this site for visitors with 28.8 kbps modems and systems that display thousands of colors. We specifically decided not to worry about the color cube and web surfers who could see only 256 colors; that would limit black-and-white images to very few shades of gray, and the photo quality would degrade.

"Content on every page!" as the famous magazine designer Roger Black would say. As Doug and I talked, it became clear that the photographs themselves should be the site, rather than an entry door followed by an antechamber, then several gallery spaces. We wanted a metaphor that was understated, not architectural or literal.

We end up agreeing on a loose interpretation of a contact sheet, enhanced by animated GIFs at the entrance and a

different group of photos each time you visit the site.

This fairly technical chapter is much more effective if seen on the Web before continuing – it uses animated GIFs *and a Perl script running on the server to present random images (www.menuez.com).*

Strategy

Digital Moments is a linear journey with a dramatic entrance, a simple intro page, 36 images, a behind-the-scenes parallel universe, and a closing page that gives visitors a chance to express their comments on the exhibition and its meaning.

I choose the number 36 as a theme, to take visitors through 36 images as though they were in Doug's private gallery, looking at a contact sheet. In fact, there are 56 images total, but I want visitors to see a "roll" of 36 at a time. This site will use a random script to present 36 different images per visit. There's no reason for them to choose which photograph to see next. I want to show the photos in the order I choose.

Advertising the number 36 at the beginning tells visitors how long the journey is, setting expectations for the trip. There is no way to see all 56 total images. They can go through again if they want to see another selection of 36 taken from the total.

The impact of this site relies on randomness, on giving the feeling of a random walk. To set the tone up front, I take an idea from Ray Guillette's animated-GIF title page for his groundbreaking site, "Sound Traffic Control."

To give the feeling of the contact sheet, I use an abstraction of the frame

markers on film as a navigational element. As an added feature, visitors can click on the large images, taking them to the other side: a flipped, negative version of the image with the photographer's notes on the circumstances under which the shot was taken. This parallel universe enriches the site and creates an unusual behind-the-scenes view. I got this through-the-picture idea from Suza Scalora's enchanting "Mythopœia" site *(all sites mentioned are listed at the Book Site)*.

Entry, Exploration, Exit

This site has a *greeting page* that tells you a bit about the gallery, warns you of the upcoming download, requests that you turn off underlining, and entices you to enter. The next page, with the animated GIFs, is the *title page*. I want it to make everyone who sees it send the URL to her friends. More than just a nice entry, this page also serves as a strong image for magazines and other media to use when reporting on the site. It also allows me to put up a short cut, for those who want to miss the title page and get right into the exhibit. This is helpful for visitors who've already seen the title page but want to go through the photographs again.

The next page is the *preamble*, where I put up a short statement from a curator's point of view. The *exploration* section is the journey through the 36 photos. There is no random access – no way to take a shortcut to the end page – you must go through the entire sequence to get to the end.

The *feedback page* should be worth reaching. I want the equivalent of the

13.3 Too much chrome: the Acrobat and browser interfaces don't leave enough room for the document.

Making a **PDF** the Right Way

It's best to build a PDF for on-screen reading from the ground up. The ground, in this case, is page size and geometry.

Virtually all computer screens have width-to-height dimensions in a 4:3 ratio, so I favor this proportion for on-screen PDFs. Operating systems, browsers, and Acrobat itself will crowd in unpredictably on this 4:3 area with their own user interfaces (especially in height) but monitor proportions themselves are constant, and the simple harmony of 4:3 makes it worth using. Similarly, the minimum screen resolution in common use is 640 × 480 pixels, so that's my target size for PDFs. Surfers with small monitors will see

pages at somewhat less than actual size, or view them in full-screen mode, without the browser and all the chrome it takes up on the screen. I'll design for either case – this means sizing the type generously and including a navigation system.

I return to my QuarkXPress document and begin reformatting. With some documents, I can begin by changing the page size in the Document Setup dialog box. In this case XPress won't let me: the page elements won't fit on the pasteboard. Instead, I create a new document, 640 by 480 points large (printer's points = pixels). I begin dragging the elements of the old document over into the new one, adjusting their sizes as necessary **[13.4]**. This is not an automatic process. I am literally redesigning the Klutz catalog, but I'm doing it in XPress, not with single-pixel GIFs.

In general, all elements involving fine detail need to be a bit larger for an on-screen PDF than would be appropriate for print. This is especially true of text, which I generally increase between 10 and 30 percent. Because the type is anti-aliased, I can get away with remarkably small sizes – some faces are legible even at 7 point, without magnification – but the idea is not to see what I can get away with, it's to get the best results.

Fortunately, this document was constructed with *style sheets*. Changing the type specs for the whole document is a simple matter of redefining existing styles. The more rationally a document is built in the first place, the easier it is to reformat. I choose 16-point for the running text, set in Chel-
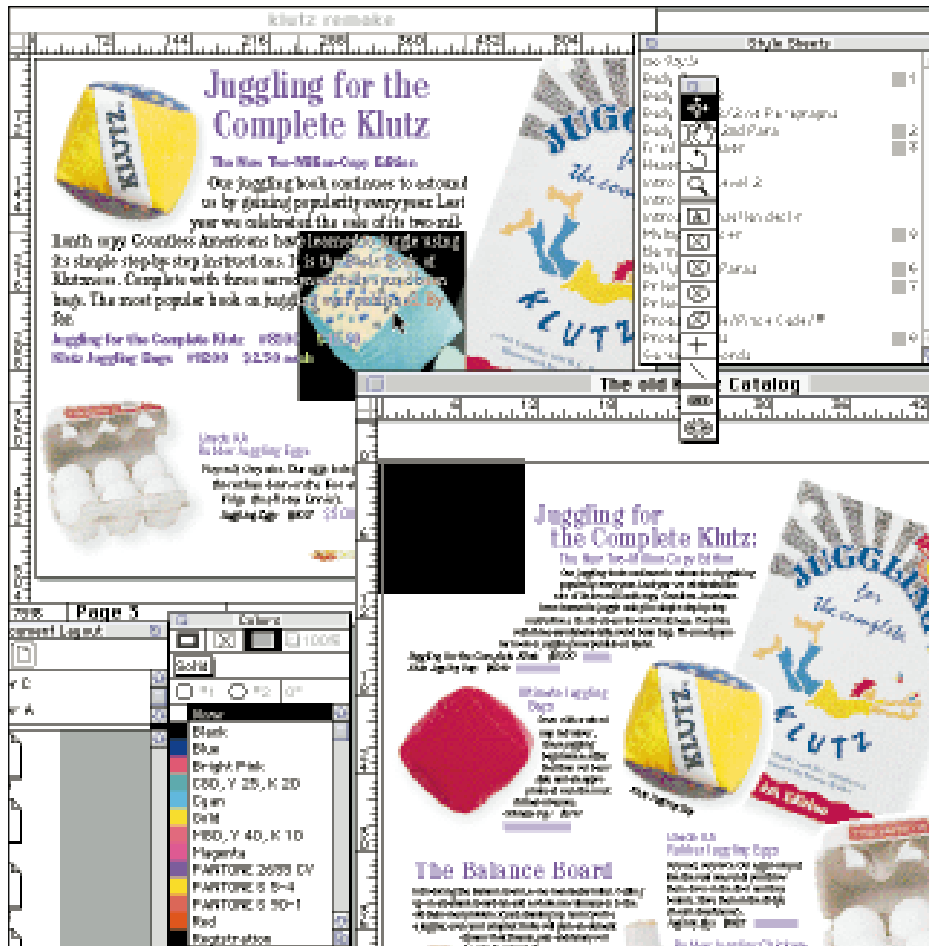
13.4 Adapting a paper-format design to a screen format.

tenham Condensed. Acrobat's serif substitution font mimics the Cheltenham family especially well, so I will choose not to embed Cheltenham when I distill the PostScript, for a savings of about 100K in the final PDF.

With larger type and less total page area, there isn't room for a one-to-one page translation. I could downscale all of the graphical elements, but I'd lose too much detail. Instead I include images selectively, and cover only one item per page [13.5].

In general, the fewer elements you arrange on a page, the fewer lines of code will be required to describe them in PostScript, and the smaller your PDF will be. As in HTML, clean construction and a considered use of images translates to a better browsing experience.

Before I go through the same process for the rest of the document, I want to address a structural problem. The printed Klutz catalog has no table of contents; it's meant to be flipped

Deadly Sin Number Seven

Paralysis

Possibly one of the most difficult things to do on the Web is to make a single page as good as it can possibly be. You can always do something to make it better. If you have a site, you know there are places you haven't touched in a long time, and two months is a very long time on the Web.

We all start with horizontal rules and blank-line typography. It's part of the learning curve. As we gain control of our pages, we raise the bar on ourselves, striving for better pages as we add tools to our toolkit. It doesn't get easier, the results just get better.

Possibly the best piece of advice I can give any designer is to roll up your sleeves and dig in. Start pushing pixels and tables around to see what works and what doesn't. I never get a page right on the first, or second, or third try. I'm always re-thinking my pages, realizing in the middle of the night how I could have done something more simply or cleanly.

A web site is an adventure. It's like surfing. You pick a goal, you start the journey, you end up somewhere else, but it turns out to be more interesting than where you thought you were going. While I advocate control over your pages, I hope

this book has expanded your creativity by freeing you from the narrow, linear thought process imposed by HTML programming.

HTML is not for dummies. It can't be learned in a week. Making great pages is exacting work, and you can always do better. Third-generation site designers work their way up the hard way, sweating the details and using whatever tools are at hand to make balanced, beautiful, communicative pages. I hope that after you make a third-generation site, people you have never met will come visit, enjoy, send you mail, and connect in ways you never expected. Then you will know why it's all worth the effort.

Dave's Guide to Better Surfing

Because I insisted, my publisher let me squeeze in a few notes on surfing. Though it is beyond the scope of this book, I would like to make a few recommendations to increase your surfing pleasure.

Link Colors: Brain Damage on the Infobahn

The Framers of the Web decided to denote links with underlining and special colors. They decided to make unvisited links blue and visited links purple. They avoided using red, probably because they thought red universally meant "STOP!" Also, they might have figured that some people can't see red, and many people (at the time) had black-and-white monitors, so underlining links was a good idea.

At user interface school, they teach us that the color red has a prominent place in the evolution of our visual system. In the jungle, the color red serves as a strong warning of danger. Red is a "flag" color: It shouts. It does not mean "STOP!"; it means "HEY!"

Good designers take advantage of this attention mechanism without overusing it and watering it down. Have you ever seen a blue speedometer dial? The right choice is to turn the "hot" unvisited links dark red and leave the visited links distinguishable, but "cool," receding into the black text [A1.1]. Red says, "Over here! Hit me!" Blue says, "Been there, done that." Red jumps out. Blue recedes into the background. Go to a magazine stand and test this hypothesis for yourself.

I want my visited links to become associated with (but distinct from) the text and leave the active (unvisited) links standing out on the page. Hot red and cool blue links should be standard on all browsers, but they aren't. To compensate, I start almost all my pages with the following header:

```
<BODY BGCOLOR="#ffffff"
TEXT="#000000" LINK="#CC0000"
ALINK="#FF3300"
VLINK="#330099">
```