

DESIGN

for **interaction** and **multimedia**





This reader was composed for the courses *Interaction Design* and *Multi-media Design Project* as taught at the University of Amsterdam in 2014.

The body text was set in the typeface *Calluna*, with titles in **Museo** and **Museo Slab**. All three are produced by the foundry exlibris (regular and italic variants are available free of charge).

All extracts are under copyright of their respective owners. All other material may be used and reproduced under the Creative Commons CC BY 4.0 License as detailed here:

<http://creativecommons.org/licenses/by/4.0>

Contents

part 1—concept

The sense of style	5
Graphic design thinking: beyond brainstorming	13
Holding off on solutions	36
User stories	37
The little blue reasoning book	41
Getting real	59
Designing for the social web	71

part 2—prototyping

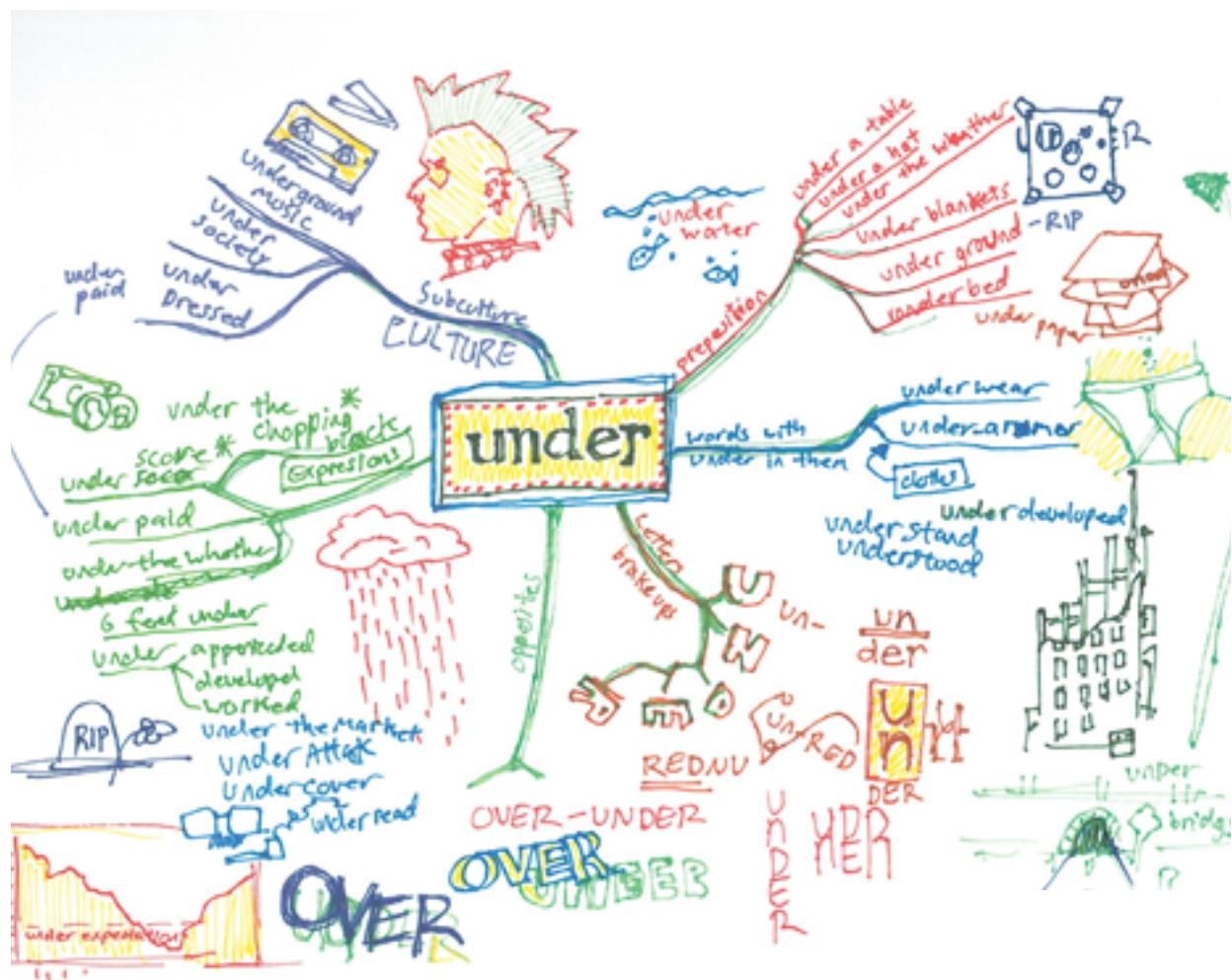
Wireframes	97
Seductive interaction design	99
100 things every designer needs to know about people	121
Don't make me think	139
Web form design: filling in the blanks	179

part 3—finishing

On writing	197
The visual display of quantitative information	233
On web typography	251

appendix—multimedia

Digital multimedia	277
--------------------	-----



A mind map from *Graphic Design Thinking: Beyond Brainstorming*, by Ellen Lupton.

How to read

For this course we are asking you to read a lot, and most of the material will not be discussed in the lectures. It helps to understand that reading in university is very different from reading a book or a magazine at home. Here are some techniques.

Read with a purpose Why are you reading this text? Do you want to understand the main idea? Do you want to be entertained? Or do you want to memorize the important concepts? If so, do you want them memorized for a test tomorrow, or to use over the next month? Each goal comes with its own approach.

Read twice Before you start, think about how much time you have to read the text. Cut that time in two, and read the text twice, skimming through it the first time. This will let you know how much there is to read, and where the important parts are. The second reading lets you learn: there's no need to read from start to finish, you can focus on the important concepts, and dip in and out where you know they are discussed.

Find the structure The human brain is not made for remembering long lists of facts. We learn by finding structure. Look at the chapters and sections. Find the throughline of the author's argument. Draw mind-maps and trees and graphs. Once you understand the skeleton of the text, you'll start remembering how all the facts fit into the main picture, and once you understand that, you'll start remembering them.

Skim first, then revise Your first reading is a skim read. Get from start to finish as fast as you can. It helps you get an idea of the structure of the text, its length and the important facts. While you skim, make a list of things that are likely to come up in the test. This list will give you a way to test yourself once you've finished the skim. Pick out a random subject, and ask yourself whether you know what it's about. If not, you'll know where to look it up, and you can reread just the relevant paragraphs.

Plan around sleep During sleep, our brains process and categorize the day's information for long term storage. The better you sleep, the better you will remember. Plan for this. Let's say it's monday morning, and you have a test on wednesday, but you can only free up 8 hours of reading time. In that case, you're better off spending 4 hours on monday, and four on tuesday, instead of doing everything in one day. You use the same amount of time, but you get extra sleep in between.

“[A] leader is needed who plays a role quite different from that of the members. [...]

This type of approach to group processes places the leader in a particular role in which he must *cease to contribute*, avoid evaluation, and refrain from thinking about solutions or group *products*. Instead he must concentrate on the group *process*, listen in order to understand rather than to appraise or refute, assume responsibility for accurate communication between members, be sensitive to unexpressed feelings, protect minority points of view, keep the discussion moving, and develop skills in summarizing.”

—Norman R.F. Maier. *Assets and Liabilities in Group Problem Solving*

part 1—concept

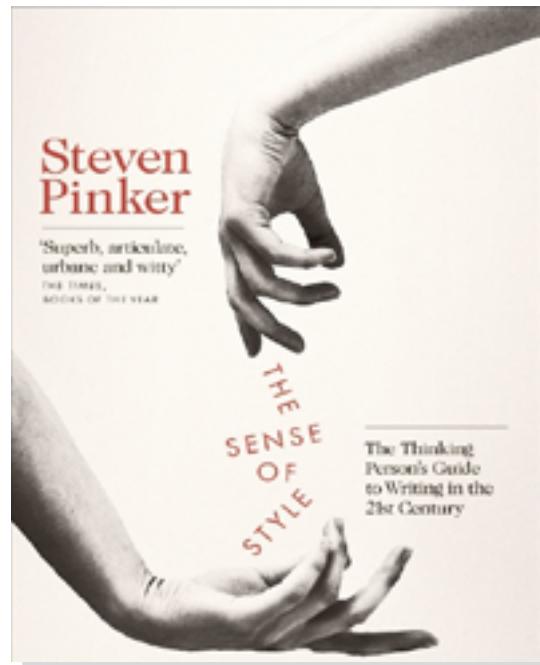


- a. Get the obvious out of the way
- b. Know your audience
- c. Kill your darlings

The sense of style

Steven Pinker

The sense of style is not a book about interaction design or multimedia design. It is a book about writing. Nevertheless, Steven Pinker manages to explain in four short pages, better than any design manual, the effect that is responsible for almost every badly designed product out there: **the curse of knowledge**. It's not that people don't want to design well, it's not that they don't put the effort in, it's quite simply that once you understand something, it becomes impossible to take the perspective of someone who doesn't.



Chapter 3

THE CURSE OF KNOWLEDGE

THE MAIN CAUSE OF INCOMPREHENSIBLE PROSE IS THE DIFFICULTY OF IMAGINING WHAT IT'S LIKE FOR SOMEONE ELSE NOT TO KNOW SOMETHING THAT YOU KNOW

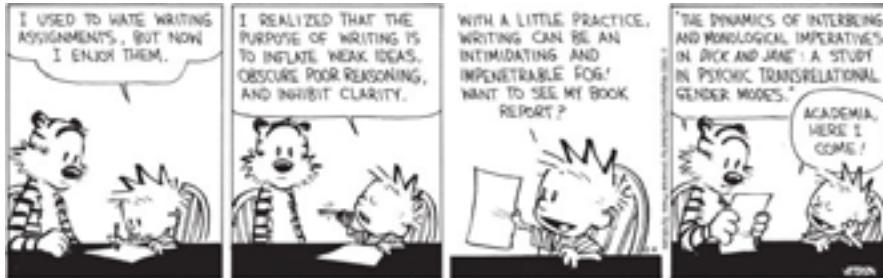
Why is so much writing so hard to understand? Why must a typical reader struggle to follow an academic article, the fine print on a tax return, or the instructions for setting up a wireless home network?

The most popular explanation I hear is the one captured in this cartoon:



Good start. Needs more gibberish.

According to this theory, opaque prose is a deliberate choice. Bureaucrats and business managers insist on gibberish to cover their anatomy. Plaid-clad tech writers get their revenge on the jocks who kicked sand in their faces and the girls who turned them down for dates. Pseudo-intellectuals spout obscure verbiage to hide the fact that they have nothing to say. Academics in the softer fields dress up the trivial and obvious with the trappings of scientific sophistication, hoping to bamboozle their audiences with highfalutin gobbledegook. Here is Calvin explaining the principle to Hobbes:



Calvin and Hobbes © 1993 Watterson. Reprinted with permission of Universal Uclick. All rights reserved.

I have long been skeptical of the bamboozlement theory, because in my experience it does not ring true. I know many scholars who have nothing to hide and no need to impress. They do groundbreaking work on important subjects, reason well about clear ideas, and are honest, down-to-earth people, the kind you'd enjoy having a beer with. Still, their writing stinks.

People often tell me that academics have no choice but to write badly because the gatekeepers of journals and university presses insist on ponderous language as proof of one's seriousness. This has not been my experience, and it turns out to be a myth. In *Stylish Academic Writing* (no, it is not one of the world's thinnest books), Helen Sword masochistically analyzed the literary style in a sample of five hundred articles in academic journals, and found that a healthy minority in every field were written with grace and verve.¹

In explaining any human shortcoming, the first tool I reach for is Hanlon's Razor: Never attribute to malice that which is adequately explained by stupidity.² The kind of stupidity I have in mind has nothing to do with ignorance or low IQ; in fact, it's often the brightest and best informed who suffer the most from it. I once attended a lecture on biology addressed to a large general audience at a conference on technology, entertainment, and design. The lecture was also being filmed for distribution over the Internet to millions of other laypeople. The speaker was an eminent biologist who had been invited to explain his recent breakthrough in the structure of DNA. He launched into a jargon-packed technical presentation that was geared to his fellow molecular biologists, and it was immediately apparent to everyone in the room that none of them understood a word. Apparent to everyone, that is, except the eminent biologist. When the host interrupted and asked him to explain the work more clearly, he seemed genuinely surprised and not a little annoyed. This is the kind of stupidity I am talking about.

Call it the Curse of Knowledge: a difficulty in imagining what it is like for someone else not to know something that you know. The term was invented by economists to help explain why people are not as shrewd in bargaining as they could be, in theory, when they possess information that their opposite



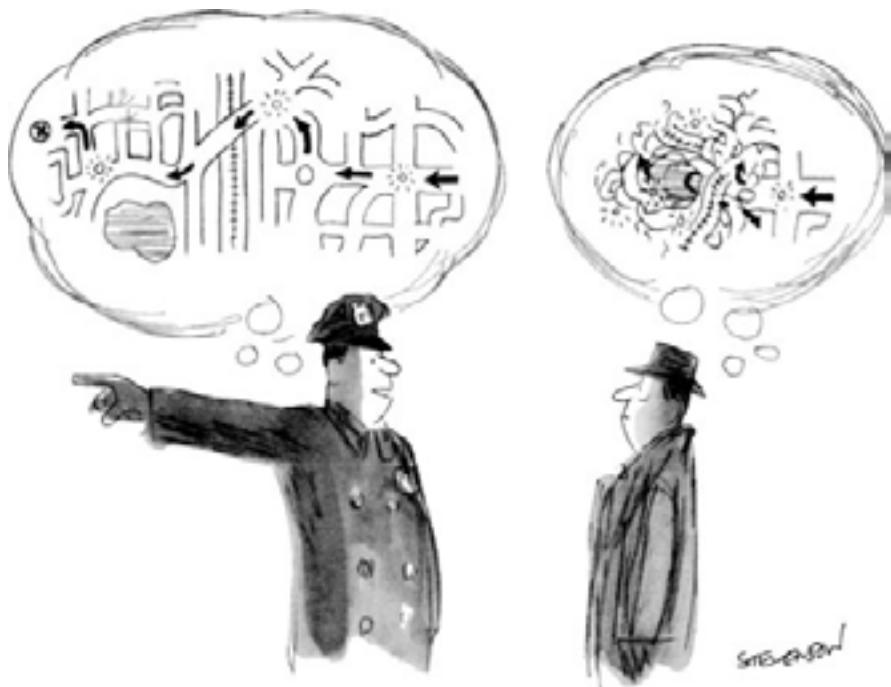
number does not.³ A used-car dealer, for example, should price a lemon at the same value as a creampuff of the same make and model, because customers have no way to tell the difference. (In this kind of analysis, economists imagine that everyone is an amoral profit-maximizer, so no one does anything just for honesty's sake.) But at least in experimental markets, sellers don't take full advantage of their private knowledge. They price their assets as if their customers knew as much about their quality as they do.

The curse of knowledge is far more than a curiosity in economic theory. The inability to set aside something that you know but that someone else does not know is such a pervasive affliction of the human mind that psychologists keep discovering related versions of it and giving it new names. There is egocentrism, the inability of children to imagine a simple scene, such as three toy mountains on a tabletop, from another person's vantage point.⁴ There's hindsight bias, the tendency of people to think that an outcome they happen to know, such as the confirmation of a disease diagnosis or the outcome of a war, should have been obvious to someone who had to make a prediction about it before the fact.⁵ There's false consensus, in which people who make a touchy personal decision (like agreeing to help an experimenter by wearing a sandwich board around campus with the word REPENT) assume that everyone else would make the same decision.⁶ There's illusory transparency, in which observers who privately know the backstory to a conversation and thus can tell that a speaker is being sarcastic assume that the speaker's naïve listeners can somehow detect the sarcasm, too.⁷ And there's mindblindness, a failure to mentalize, or a lack of a theory of mind, in which a three-year-old who sees a toy being hidden while a second child is out of the room assumes that the other child will look for it in its actual location rather than where she last saw it.⁸ (In a related demonstration, a child comes into the lab, opens a candy box, and is surprised to find pencils in it. Not only does the child think that another child entering the lab will know it contains pencils, but the child will say that he himself knew it contained pencils all along!) Children mostly outgrow the inability to separate their own knowledge from someone else's, but not entirely. Even adults *slightly* tilt their guess about where a person will look for a hidden object in the direction of where they themselves know the object to be.⁹

Adults are particularly accursed when they try to estimate other people's knowledge and skills. If a student happens to know the meaning of an uncommon word like *apogee* or *elucidate*, or the answer to a factual question like where Napoleon was born or what the brightest star in the sky is, she assumes that other students know it, too.¹⁰ When experimental volunteers are given a list of anagrams to unscramble, some of which are easier than others because the answers were shown to them beforehand, they rate the ones that

are easier for *them* (because they'd seen the answers) to be magically easier for *everyone*.¹¹ And when experienced cell phone users were asked how long it would take novices to learn to use the phone, they guessed thirteen minutes; in fact, it took thirty-two.¹² Users with less expertise were *more* accurate in predicting the learning curves, though their guess, too, fell short: they predicted twenty minutes. The better you know something, the less you remember about how hard it was to learn.

The curse of knowledge is the single best explanation I know of why good people write bad prose.¹³ It simply doesn't occur to the writer that her readers don't know what she knows—that they haven't mastered the patois of her guild, can't divine the missing steps that seem too obvious to mention, have no way to visualize a scene that to her is as clear as day.* And so she doesn't bother to explain the jargon, or spell out the logic, or supply the necessary detail. The ubiquitous experience shown in this *New Yorker* cartoon is a familiar example:



Anyone who wants to lift the curse of knowledge must first appreciate what a devilish curse it is. Like a drunk who is too impaired to realize that he is too impaired to drive, we do not notice the curse because the curse prevents us from noticing it. This blindness impairs us in every act of communication. Students in a team-taught course save their papers under the name of the professor who assigned it, so I get a dozen email attachments named "pinker.doc." The professors rename the papers, so Lisa Smith gets back a dozen attachments named "smith.doc." I go to a Web site for a trusted-traveler



program and have to decide whether to click on GOES, Nexus, GlobalEntry, Sentri, Flux, or FAST—bureaucratic terms that mean nothing to me. A trail map informs me that a hike to a waterfall takes two hours, without specifying whether that means each way or for a round trip, and it fails to show several unmarked forks along the trail. My apartment is cluttered with gadgets that I can never remember how to use because of inscrutable buttons which may have to be held down for one, two, or four seconds, sometimes two at a time, and which often do different things depending on invisible “modes” toggled by still other buttons. When I’m lucky enough to find the manual, it enlightens me with explanations like “In the state of {alarm and chime setting}. Press the [SET] key and the {alarm ‘hour’ setting}→{alarm ‘minute’ setting}→{time ‘hour’ setting}→{time ‘minute’ setting}→{‘year’ setting}→{‘month’ setting}→ {‘day’ setting} will be completed in turn. And press the [MODE] key to adjust the set items.” I’m sure it was perfectly clear to the engineers who designed it.

Multiply these daily frustrations by a few billion, and you begin to see that the curse of knowledge is a pervasive drag on the strivings of humanity, on a par with corruption, disease, and entropy. Cadres of expensive professionals—lawyers, accountants, computer gurus, help-line responders—drain vast sums of money from the economy to clarify poorly drafted text. There’s an old saying that for the want of a nail the battle was lost, and the same is true for the want of an adjective: the Charge of the Light Brigade during the Crimean War is only the most famous example of a military disaster caused by vague orders. The nuclear meltdown at Three Mile Island in 1979 has been attributed to poor wording (operators misinterpreted the label on a warning light), as has the deadliest plane crash in history, in which the pilot of a 747 at Tenerife Airport radioed he was *at takeoff*, by which he meant “taking off,” but an air traffic controller interpreted it as “at the takeoff position” and failed to stop him before he plowed his plane into another 747 on the runway.¹⁴ The visually confusing “butterfly ballot” given to Palm Beach voters in the 2000 American presidential election led many supporters of Al Gore to vote for the wrong candidate, which may have swung the election to George W. Bush, changing the course of history.

1
OFFICIAL BALLOT, GENERAL ELECTION
PALM BEACH COUNTY, FLORIDA
NOVEMBER 7, 2000

A
OFFICIAL BALLOT, GENERAL ELECTION
PALM BEACH COUNTY, FLORIDA
NOVEMBER 7, 2000

<p>ELECTORS FOR PRESIDENT AND VICE PRESIDENT</p> <p>(A vote for the candidates will actually be a vote for their electors.)</p> <p>(Vote for Group)</p>	(REPUBLICAN)	
	GEORGE W. BUSH - PRESIDENT	3 →
	DICK CHENEY - VICE PRESIDENT	
	(DEMOCRATIC)	
	AL GORE - PRESIDENT	5 →
	JOE LIEBERMAN - VICE PRESIDENT	
	(LIBERTARIAN)	
	HARRY BROWNE - PRESIDENT	7 →
	ART OLIVIER - VICE PRESIDENT	
	(GREEN)	
	RALPH NADER - PRESIDENT	9 →
	WINONA LaDUKE - VICE PRESIDENT	
	(SOCIALIST WORKERS)	
	JAMES HARRIS - PRESIDENT	11 →
	MARGARET TROWE - VICE PRESIDENT	
	(NATURAL LAW)	
	JOHN HAGELIN - PRESIDENT	13 →
	NAT GOLDHABER - VICE PRESIDENT	
		(REFORM)
		PAT BUCHANAN - PRESIDENT
		EZOLA FOSTER - VICE PRESIDENT
		(SOCIALIST)
		DAVID McREYNOLDS - PRESIDENT
		MARY CAL HOLLIS - VICE PRESIDENT
		(CONSTITUTION)
		HOWARD PHILLIPS - PRESIDENT
		J. CURTIS FRAZIER - VICE PRESIDENT
		(WORKERS WORLD)
		MONICA MOOREHEAD - PRESIDENT
		GLORIA La RIVA - VICE PRESIDENT
		WRITE-IN CANDIDATE
		To vote for a write-in candidate, follow the directions on the long stub of your ballot card.

The confusing ballot that changed history.

source https://commons.wikimedia.org/wiki/File:Butterfly_large.jpg

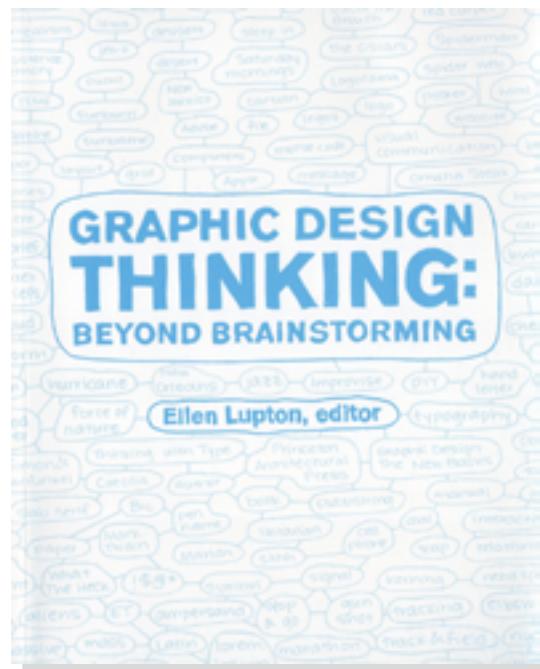
Graphic design thinking: beyond brainstorming

Ellen Lupton & Jennifer Cole Phillips

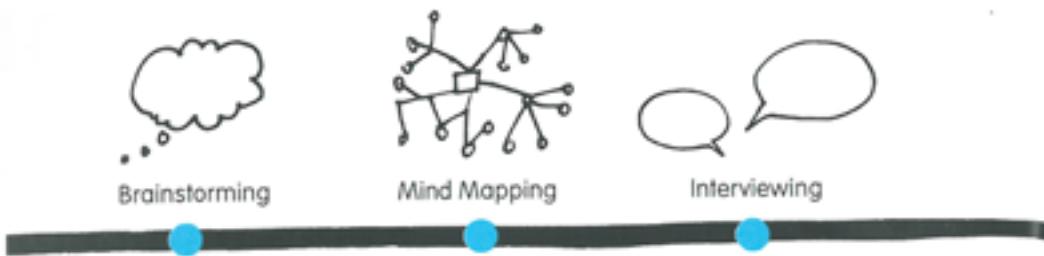
Everybody can draw a perfectly acceptable subscription form in 10 minutes. **What if you were given a month?** Could you spend 20 days designing a single form? How would you start? How different can you make the form from the one you scribbled down in 10 minutes? Could you make it so much better that it would justify a month of work? What would such a super-form even look like?

This is where design thinking comes in: because some subscription forms are worth a month of work. Design exercises will help you get rid of the obvious solution, and ask what the client really wants, and what the user really wants. They'll help you to look beyond the obvious.

Do we even need a subscription form at all?



DEFINING THE PROBLEM



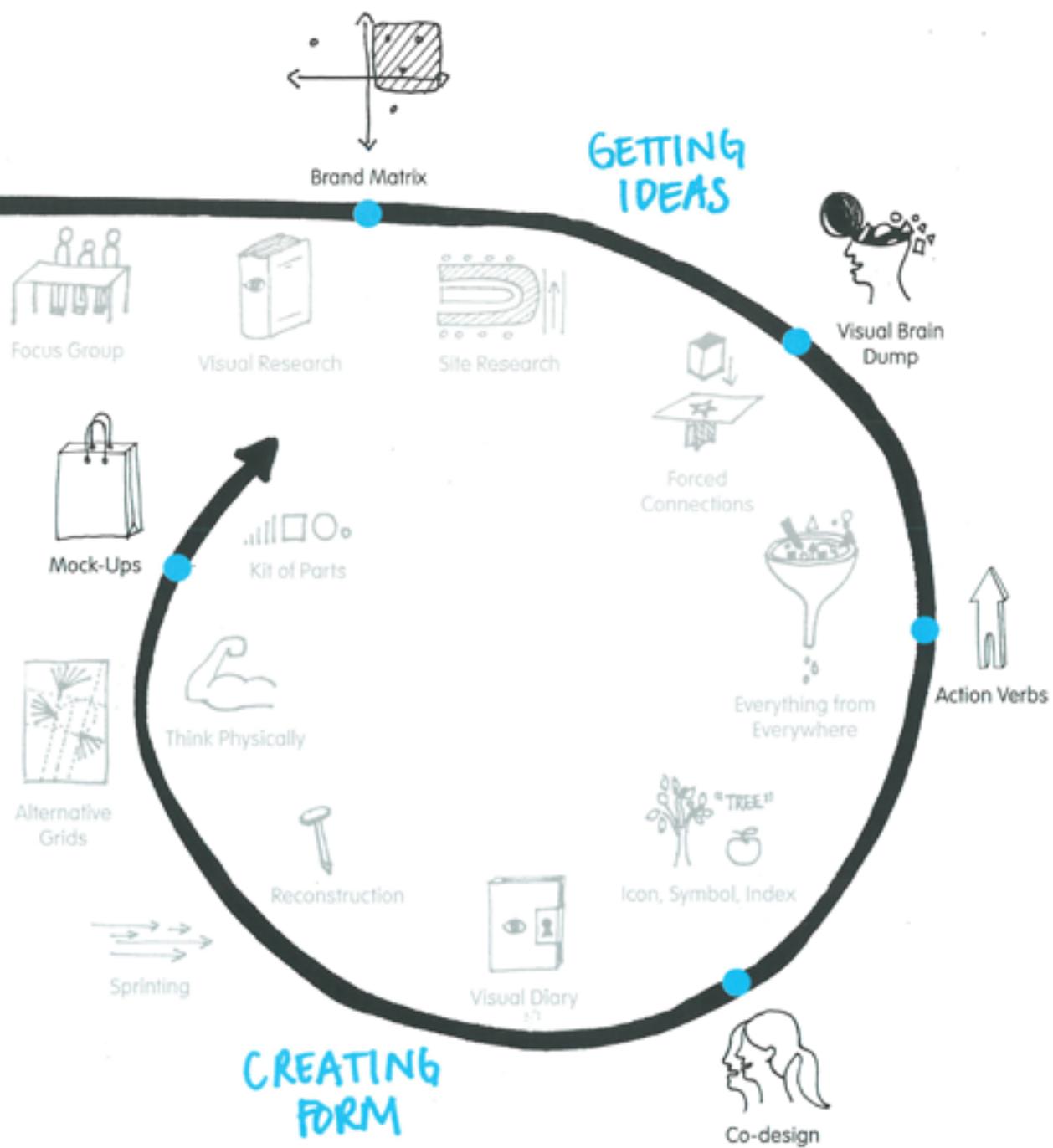
The Design Process

This chapter follows one real-world project through each phase of the design process, from researching the problem to generating ideas to creating form. Along the way, the design team employed various techniques of design thinking that are explored in more detail later in the book. The project was conducted by students in the Graphic Design MFA program at MICA. A team of designers, led by Jennifer Cole Phillips, worked with client Charlie Rubenstein in an effort to raise awareness of homelessness in the local community. Knowing that they could not address all aspects of homelessness in a single project, the team worked to narrow their scope and create a project that could be successfully realized with available resources.

In 2008 Baltimore City documented 3,419 homeless people living within its limits. The team built their campaign around the number 3419, signaling both the scale of the problem and the human specificity of the homeless population. Working in conjunction with the client, the design team conceived and implemented a project that aimed to educate middle school students about homelessness. *Ann Liu*

"The design process, at its best, integrates the aspirations of art, science, and culture"

Jeff Smith



Defining the Problem

16

Interviewing. Designers talk to clients and other stakeholders to learn more about people's perceived wants and needs. Shown here are highlighted excerpts from a videotaped conversation with Charlie Rubenstein, the chief organizer of the 3419 homeless awareness campaign. *See more on Interviewing, page 26.*

Paired with his body language, Charlie's comments showed that he was dissatisfied with the current state of homeless services but also recognized their value.

Here, Charlie started talking more quickly and with more animation in his tone and body language, indicating his passion for treating homeless people like real people instead of just a number.

People often need time to get to the bottom line. After forty-five minutes, we were finally able to hear the core of what the client was trying to achieve with the 3419 campaign.

Interview with Charlie Rubenstein

If we are talking about 3419 as an organization, where do you see it five years from now?

Well, I want to redesign the way we treat homelessness in the city. I don't want to do it from a nonprofit, third party level, I want to do it from the inside out.

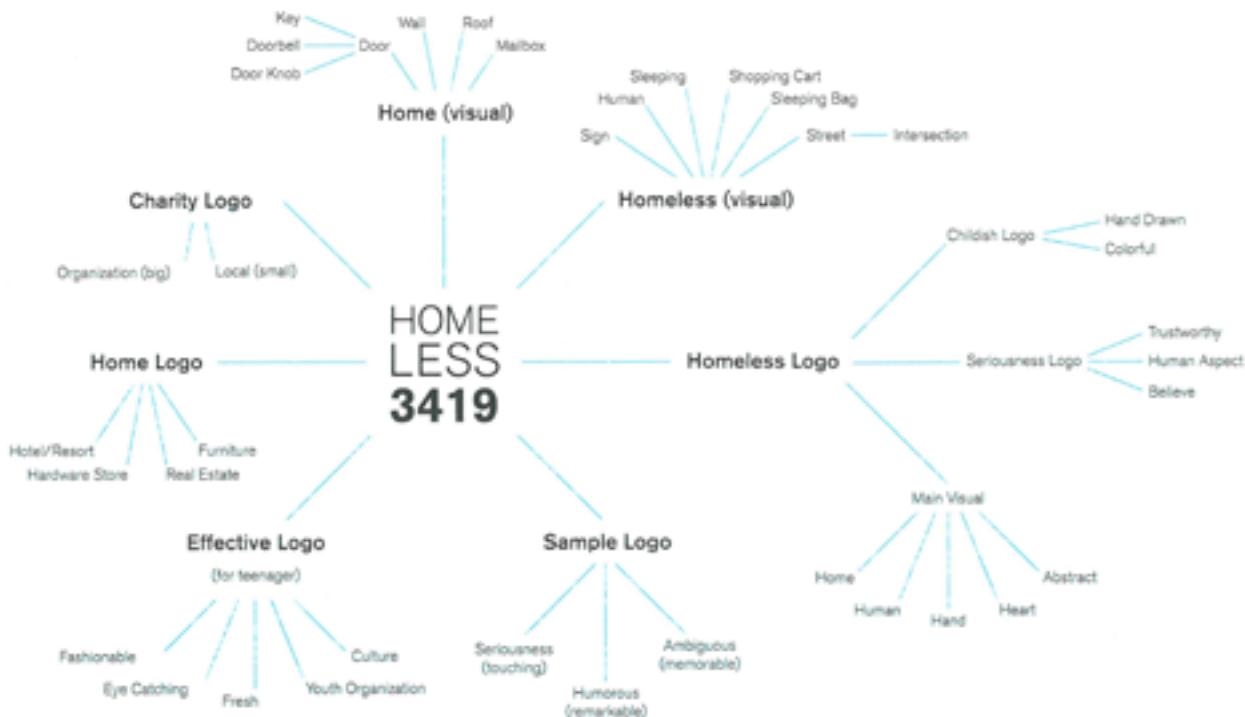
My biggest problem with Baltimore's homeless services, or whatever you want to call it, is that they don't go very deep. There isn't enough reach. For me, it isn't that they are doing it wrong, there just needs to be a new way to do it.

Can you give me a specific example of a new way?

Sure. There needs to be more qualitative research done. There are more quantitative studies around than you could read in a lifetime....So, if you have a policy, its biggest problem is that it's singular and won't work for everybody. The biggest problem is that, even institutionally, we are treating people as numbers. We are treating people as a genre, as if they are faceless, heartless. Like they are just 3419.

I want to create a people-based program.

Because we are talking about people, and there are so many different kinds of them. So, what if we tried to understand who each of these people are? Where they came from and what their names are...I want to do a six-month qualitative research study where we actually go out and interview over five hundred homeless people. And not just one time but over a period of time, so we can understand who these people are.



Mind Mapping. Designers use associative diagrams to quickly organize possible directions for a project. Design: Christina Beard and Supisa Wattanasansanee. [See more on Mind Mapping, page 22.](#)

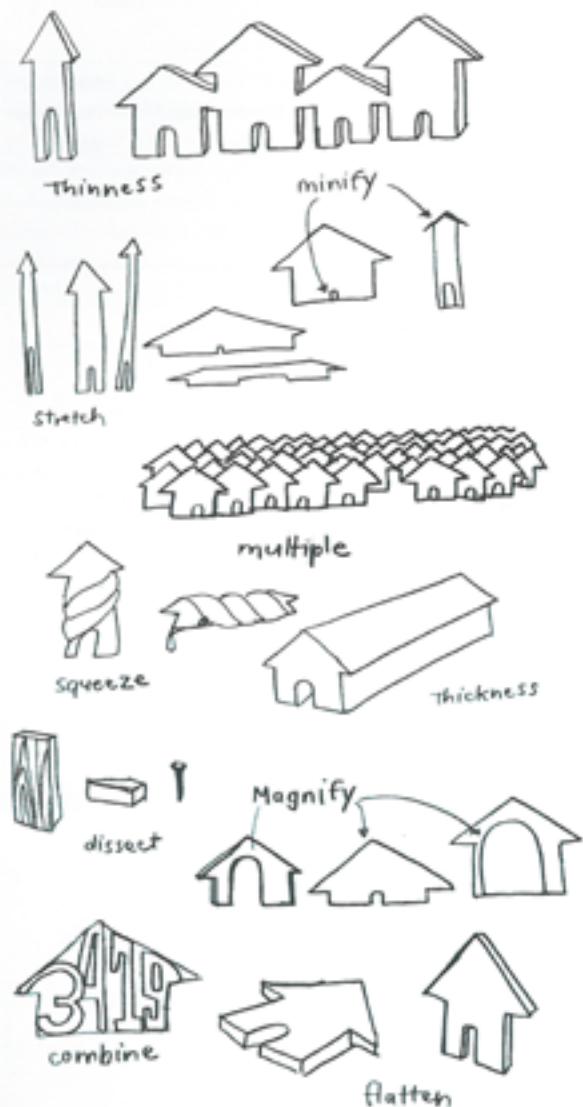


Brand Matrix. This diagram shows relationships among different social change campaigns. Some are single events, while others take place continuously. Some happen online, others, in person. [See more on Brand Matrix, page 42.](#)

Brainstorming. By focusing the campaign on what homeless people have and not what they materially lack, designers chose "can," "want" and "are" as the voice of the project. [See more on Brainstorming, page 16.](#)

Getting Ideas

18



Action Verbs. A fun way to quickly produce visual concepts is to apply action verbs to a basic idea. Starting with an iconic symbol of a house, the designer transformed the image with actions such as magnify, minify, stretch, flatten, and dissect. Design: Supisa Wattanasansanee. [See more on Action Verbs, page 74.](#)

3419 3419
3419 3419
3419 3419
3419 3419
3419 3419
3419 3419
3419 3419
3419 3419

Visual Brain Dumping. Designers created various typographic treatments of 3419 and grouped them together in order to find the best form for the project. Design: Christina Beard, Chris McCampbell, Ryan Shelley, Wesley Stuckey. [See more on Visual Brain Dumping, page 62.](#)

Creating Form



Collaboration. The stencil form was shared with a different team of designers to explore ways that users could transform it. Design: Paige Rommel, Wednesday Trotto, Hannah Mack. [See more on Collaboration, page 92.](#)



Mock-Ups. Making visual mock-ups showing how concepts, like a pillowcase poster, could be applied in real life helps make it concrete for clients and stakeholders. Design: Lauren P. Adams. [See more on Mock-Ups, page 136.](#)

3419

Original DIN Bold

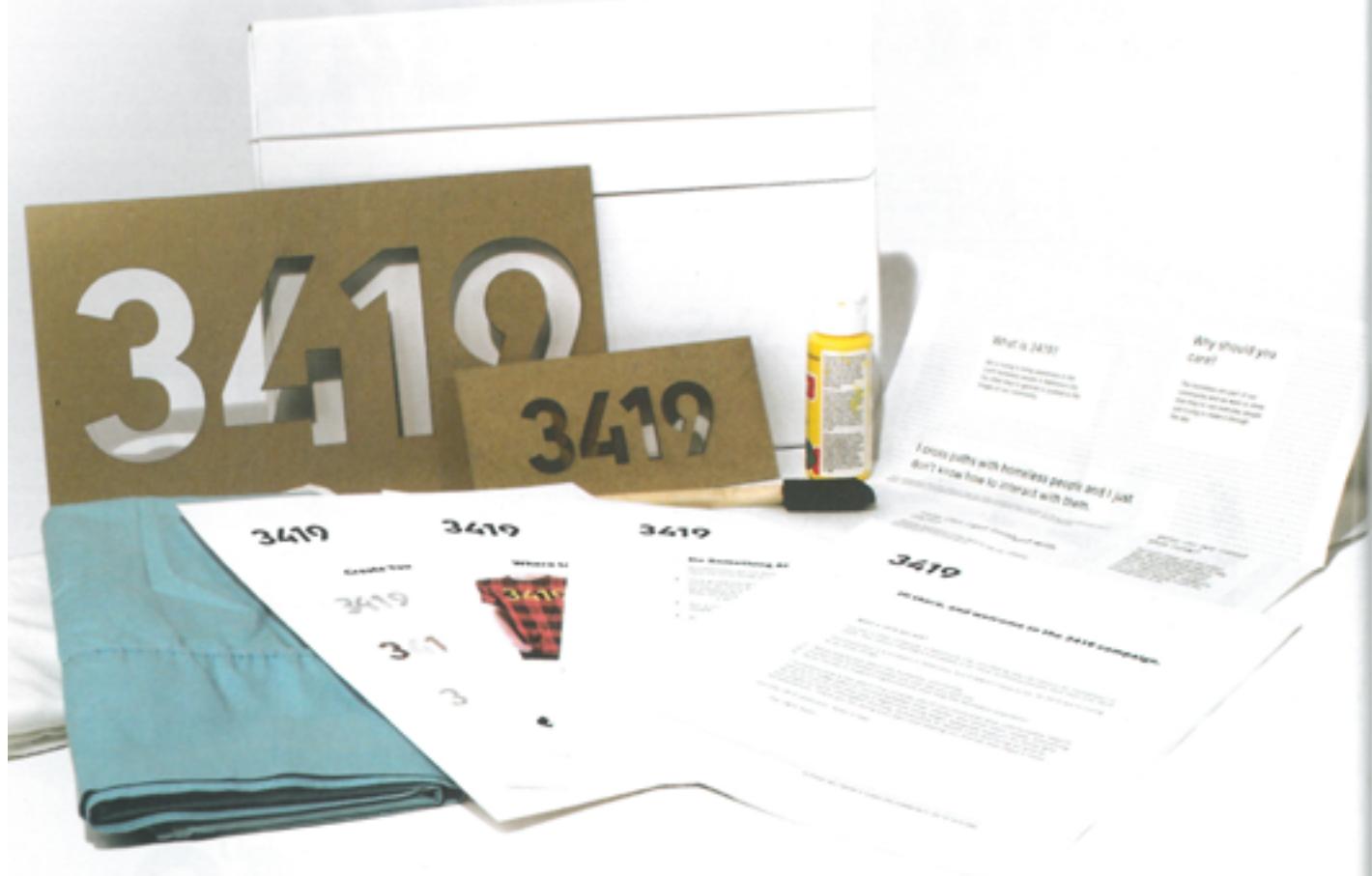
3419

Simplified visual weight

3419

Modified for stencil

Ready for Reproduction. Having decided that a stencil would be part of the 3419 identity, the designer modified letters from the typeface DIN to create a custom mark that could function as a physical stencil. Design: Chris McCampbell.



The Whole Kit and Kaboodle. Designers created a poster and worksheets to teach kids about homelessness in Baltimore and what they can do to help. The kit also includes two stencils, two pillowcases, a bottle of paint, and a brush. The kit invites students

to create their own pillowcase posters, engaging them actively in thinking about the problem and what it means to go to sleep without your own bed. Design: Lauren P. Adams, Ann Liu, Chris McCampbell, Beth Taylor, Krissi Xenakis.

The Cycle Continues

Design is an ongoing process. After a team develops a project, they implement, test, and revise it. For the 3419 homeless awareness campaign, the end result of the initial design phase was a kit for use in middle schools. The kit allowed the project team to interact with their audience, while the users created their own visual contributions with the materials provided and thus expanded the project's language. The design process began all over again.



21



Co-design. The 3419 design team conducted an afternoon workshop with local middle-school students in order to create pillowcases that would be used as posters to hang around their school and city. Co-design involves users in the creative process. *See more on Co-design, page 96.*

02

all i want
TO DO
is write
THINGS
down for
YOU

“Sometimes an idea can be our worst enemy, especially if it blocks our thinking of alternatives.”

Don Koberg and Jim Bagnall

23

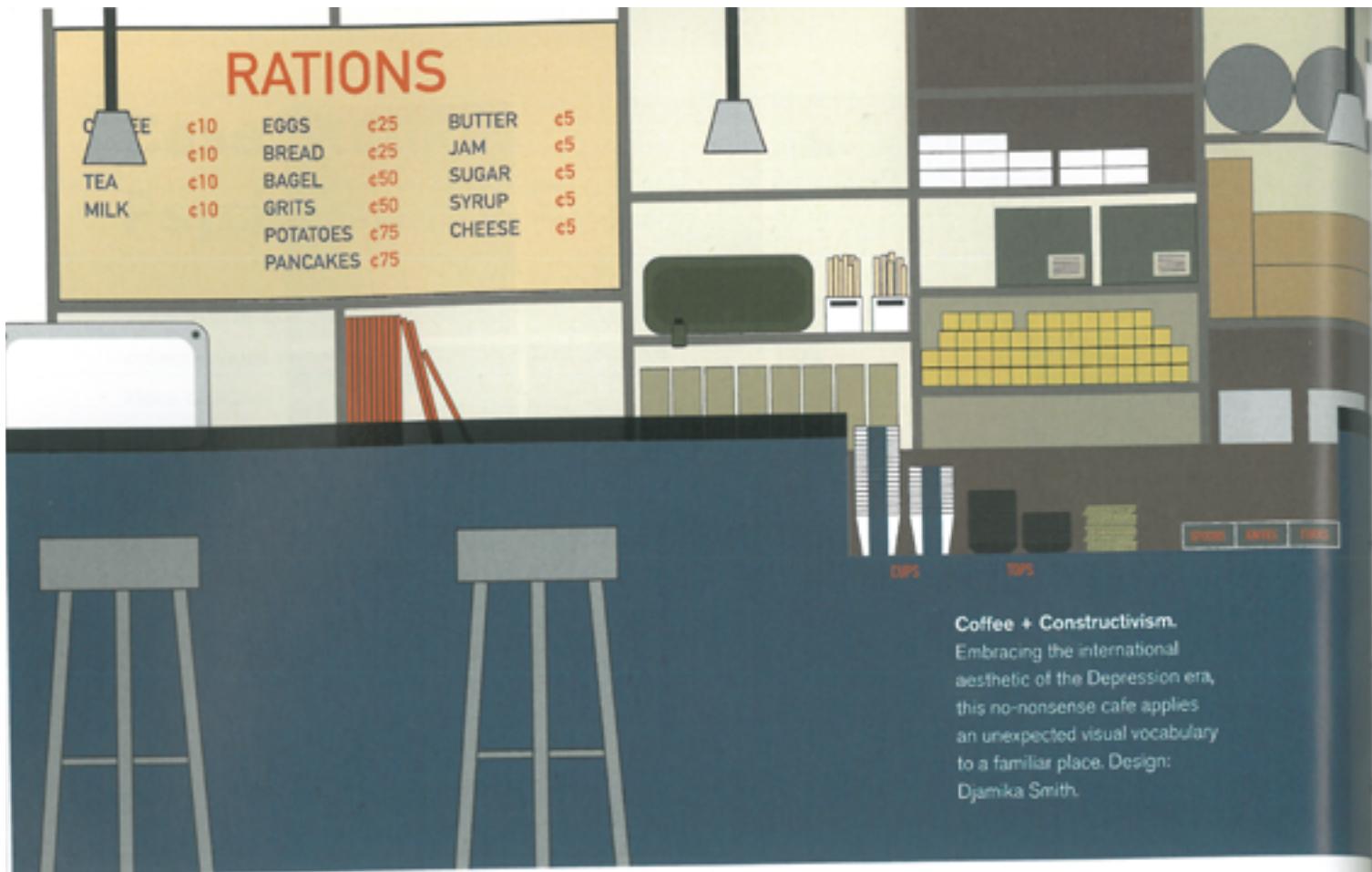
How to Get Ideas

Once you have defined your problem, it's time to devise solutions and develop concepts in greater depth. This often means communicating ideas to yourself and to other designers on your team as well as to clients and potential end users. An intriguing sketch from your notebook or a provocative phrase scribbled on a whiteboard can now become a concept with a concrete shape and a vivid story to tell.

The first phase of the design process involves casting a wide net around your problem; along the way, you may come up with dozens of different concepts, from the obvious to the outlandish. Before devoting time and energy to developing a single solution, designers open

their minds to numerous possibilities and then zero in on a few. The tools explored in this chapter include ways to generate variations on a single concept as well as ways to quickly explore, explain, and expand on a core idea.

With a single-frame project like a book cover, poster, or editorial illustration, the move from ideation to execution is fluid and direct. With complex projects such as websites, publications, and motion graphics, designers tend to work schematically using diagrams, storyboards, and sequential presentations before developing the visual details and appearance of a solution. Physical and digital mock-ups help designers and clients envision a solution in use.



Coffee + Constructivism.
Embracing the international
aesthetic of the Depression era,
this no-nonsense cafe applies
an unexpected visual vocabulary
to a familiar place. Design:
Djamika Smith.

Forced Connections

From cookie dough ice cream to zombie/Jane Austen novels, intriguing ideas often result when unlikely players collide. By brainstorming lists of products, services, or styles, and then drawing links between them, designers can forge concepts imbued with fresh wit and new functions. For example, most java houses today look alike. They feature dark reds and browns, wooden tables and floors, and—if you're lucky—a comfortable couch. But what if a cafe had constructivist decor instead? Or what if your errand to the print shop doubled as your coffee break? Likewise, laundromats get a rap for being dirty and dingy, yet public laundries offer a greener alternative to individually owned appliances. How could you make a trip to the laundromat a more inviting experience? Combining services or applying unexpected styles can change the way we think about predictable categories. Lauren P. Adams and Beth Taylor

Don Koberg and Jim Bagnell discuss the idea of forced connections as a tool for product designers in their book *The Universal Traveler: A Self-Systems Guide to Creativity, Problem-Solving, and the Process of Reaching Goals* (San Francisco: William Kaufmann, 1972).



Espresso + Ink. This concept combines two businesses into one. Just don't spill coffee on your inkjet prints. Design: Kimberly Gim.

How to Force a Connection

01 Choose a connection.

Depending on whether you are designing a business service, a logo, or a piece of furniture, decide what kinds of connections to force. Maybe you want to combine services (gym + laundromat), aesthetics (serious literature + cheap horror), or functions (sofa + work space).

02 Make two lists. Let's say your goal is to design a new kind of coffee shop. Brainstorm lists of functions—tailor, pet grooming,

bicycle repair. Make connections and imagine the results. What would each new business be called? What needs does it address? Who is the audience? Would you want to go there?

03 Combine styles, messages, or functions. Identify conflicting or overlapping ideas embodied in your core problem (museum + nature, school + lunch, coffee + economy). Create lists of images and ideas associated with each element, and draw connections between them.

04 Choose one or more

viable ideas. Make simple graphic images of interiors, products, and other applications to bring your concept to life. Your choices of forms, color, language, and typography can all speak to the core conflicts embodied in your concept. Use your forced connections to uncover the aesthetic and functional possibilities of your idea. Flat, graphic diagrams like the ones shown above quickly flesh out the main features of an idea without getting burdened with specifics.

Case Study

Multipurpose Tools

26

Your house is filled with tools. What happens when you combine two or more of these instruments to make something new? This quick exercise using forced connections yields some ideas that are impractical or absurd but others that could become real products with clever functions. Designer Lauren P. Adams started with verbal lists and then made sketches combining ideas from different lists.



Office Tools	Kitchen Tools	Garage Tools
thumbtack	spatula	wrench
stapler	ladle	hammer
scissors	whisk	nail
masking tape	knife	tape measure
hole puncher	tongs	T-square
pencil	vegetable peeler	trowel
glue	corkscrew	handsaw
ruler	can opener	clamp
marker	drink shaker	screw
compass	measuring cup	screwdriver
paperclip	dish scrubber	level
staple remover	grater	staple gun
	funnel	sledgehammer
	rolling pin	
	sieve	

Handsaw + Ruler. Nearly every saw cut requires measuring first, so why not add a ruler to the saw blade?

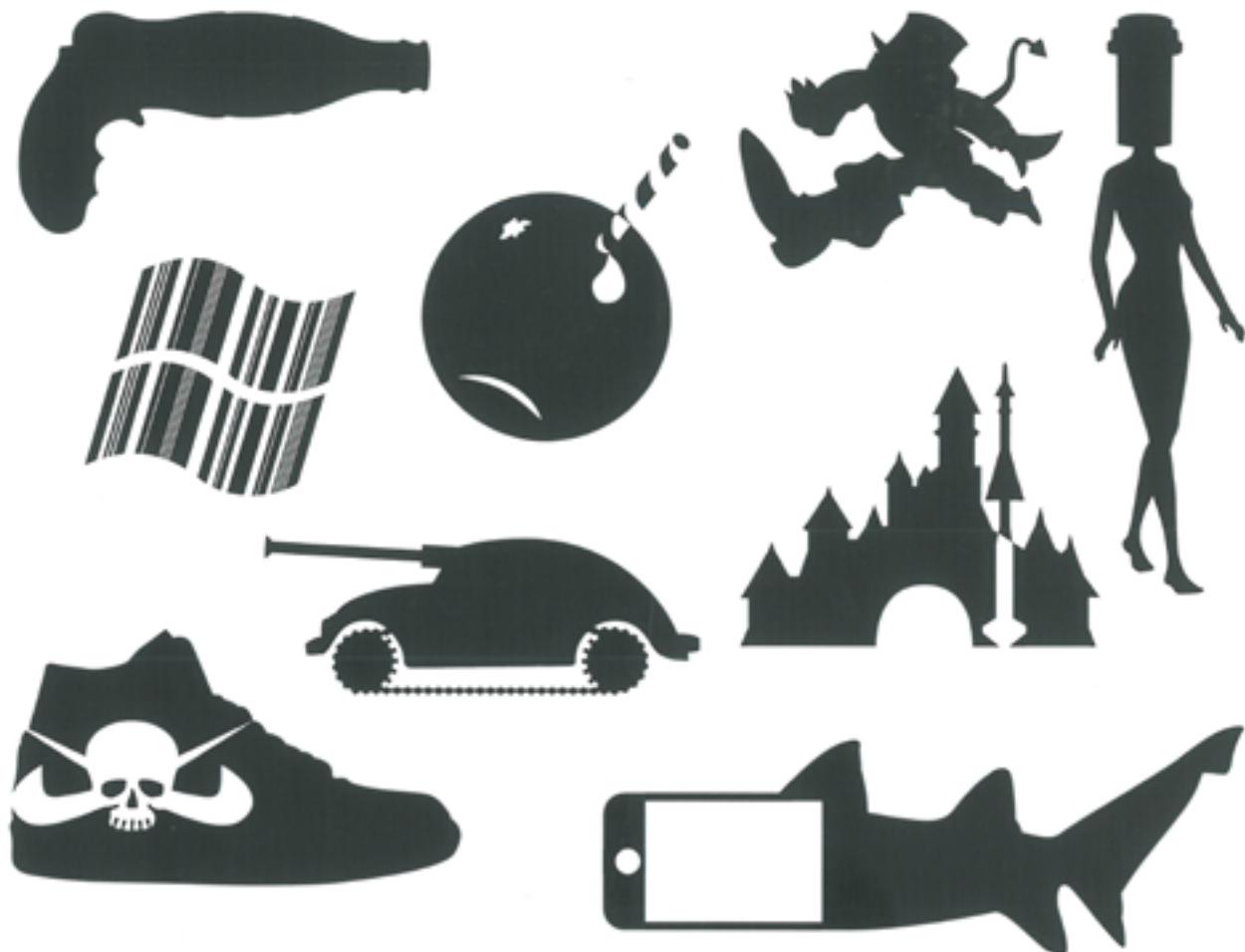
Grater + Trowel. Scoop up your freshly grated cheese, or crumble chunks of hardened dirt before planting.

Scissors + Wrench. This looks like a clever idea until you consider trying to actually cut something.

Thumbtack + Screw. The thumbtack head would give your hand something to grip while the screw threads make a secure connection.

Sledgehammer + Drink Shaker. The motion of hammering is similar to the motion used to shake a drink. (Sober up before swinging that hammer around.)

Compass + Knife. Cut your cookies to an exact dimension with this gadget for the cook who loves math.



27

Case Study

Visual Puns

Designers often use humor to hook the viewer's interest. Slapping disparate elements together yields unexpected offspring, and when the result is awkward enough to be funny, viewers come away with a laugh. Cleverness often carries a critical edge as well. In the visual puns shown here, designer Ryan Shelley created dark imagery out of recognizable brands, inviting the viewer into a Dr. Seuss-like world where cars, phones, and Barbie dolls take on sinister identities.

Quality Control. Iconic products are combined with unpleasant forms (guns, pills, bombs, sharks), creating a commentary on the grimmer side of capitalism. The designer translated these graphic icons into graffiti stencils.



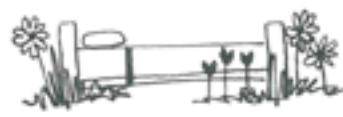
Minify: City Cabin



Magnify: Giant Garage



Rearrange: Sleep In the Kitchen

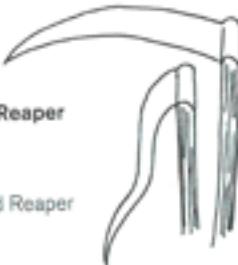


Reverse: Live in the Garden

Rethinking the House. Koberg and Bagnall used action verbs to think about the house in new ways in *The Universal Traveler* (1972). They got the idea from Alex F. Osborn, who presented this technique in his book *Applied Imagination* (1953). Concepts: Don Koberg and Jim Bagnall. Sketches: Lauren P. Adams.

Action Verbs

Alex F. Osborn, who became famous for inventing brainstorming, devised other useful techniques that encourage creativity. One process involves taking an initial idea and applying different verbs to it, such as magnify, rearrange, alter, adapt, modify, substitute, reverse, and combine. These verbs prompt you to take action by manipulating your core concept. Each verb suggests a structural, visible change or transformation. Designers can use this exercise to quickly create fresh and surprising variations on an initial idea. Even a cliché image such as the grim reaper or hitting the bull's-eye can take a surprising turn when you subject it to actions. Designers can apply this technique to objects and systems as well as images. Try reinventing an everyday object such as a house, a book, or a couch by imagining it in a different scale, material, or context. *Lauren P. Adams*



Grim Reaper

Melt:
Flaccid Reaper



Combine:
Beaked Reaper



Flatten:
Chalk-line Reaper

Sketches: Molly Hawthorne



Hang In There
Sketches: Beth Taylor



Flatten

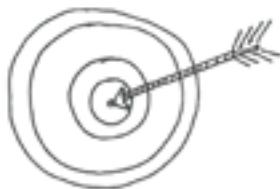


Stretch



Invert

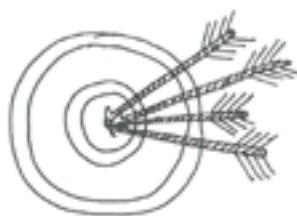
29



Hit the Bull's-eye
Sketches: Chris McCampbell



Magnify



Multiply



Invert

How to Activate an Idea

01 Start with a basic concept.

Maybe it's an obvious idea, such as using a target to represent performance or a struggling kitten to show courage. Like many clichés, these familiar images provide a common ground for communication.

02 Apply a series of actions to

the core image or idea. Create quick sketches. In addition to the words illustrated above, try more unusual ones like melt, dissect, explode, shatter, or squeeze. Don't judge your sketches or spend too much time on one idea; move quickly through your list.

03 Step back and look at what

you did. Have you given a new twist to an old cliché? Have you solved a familiar problem in a fresh way? Have you a new ending to an old story? (What if the kitten falls out of the tree? What if the grim reaper kicks his own bucket?) Find your best ideas and take them farther.



Collaboration

Have you ever seen a collaborative design project fall on its face? (From a thirty-story drop. Onto poison spikes.) Sometimes, designers let their individuality get in the way of teamwork. Effective collaboration yields something new, not a Frankensteinian mash up of parts. In a productive team, each member has ownership over some aspect of the project, bringing a valuable set of perspectives and skills to the group, but each person is willing to merge individual ideas into the bigger structure. The aphorism "two brains are better than one" does not apply to two brains crammed inside one skull. Networks aren't ten hard drives thrown into one box, but rather ten different components that share and communicate.

Working together often involves an element of play. Humor, intelligence, and experimentation are integral to crafting engaging ideas. Sometimes, the best ideas evolve from conversations. Designers pride themselves on interacting with their clients, but designers also need to communicate well with each other. A satisfying collaboration is like building a superfort out of Legos with your friends when everyone shares the bricks. The result will be different from what any one person expected. *Ryan Shelley and Wesley Stuckey*

"The space between people working together is filled with conflict, friction, strife, exhilaration, delight, and vast creative potential."

Bruce Mau



31

Reinvent Mural. These icons for a public mural were designed collaboratively. Design: Lauren P. Adams, Christina Beard, Chris McCampbell. Curator: Cathy Byrd, Maryland Art Place.

How to Collaborate

01 **Sit together.** Work at the same table so that ideas can develop in relation to each other. Skype and iChat don't count.

02 **Hear and be heard.** Nobody has the same experience and background as you; other team members are counting on your eye to help mold a unique outcome. Collaboration involves listening as well as talking, giving as well as taking. A degree of conflict is inevitable in any project—learn to go with the flow.

03 **Identify leaders.** Leadership can be both formal and informal. In corporate settings, groups tend to have an assigned leader. In the looser context of an activist collective or a student collaboration, leadership may emerge organically. Leaders help keep a project on track by distributing duties, representing the team, and prompting decisions when the process stalls. A large team may have several leaders; in a group of just two or three people, everyone could be a leader.

04 **Play.** But play nice. Everyone's goal should be the overall success of the project, regardless of who initiates various ideas along the way. Just like in a game, a little conflict and competition among players can be good for the process, but don't get stuck on protecting your own contribution. Focus on how a team can achieve more ambitious results than an individual working alone.

Alex F. Osborn developed the technique of brainstorming in his book *Applied Imagination: Principles and Procedures of Creative Thinking*. (New York: Scribner's, 1953).

Brainstorming

What picture comes to your mind when you hear the word *brainstorm*? Many of us conjure a dark cloud crackling with lightning and raining down ideas. The original metaphor, however, was military, not meteorological. The term *brainstorming* was coined by Madison-Avenue ad man Alex F. Osborn, whose influential book *Applied Imagination* (1953) launched a revolution in getting people to think creatively. Brainstorming means attacking a problem from many directions at once, bombarding it with rapid-fire questions in order to come up with viable solutions. Osborn believed that even the most stubborn problem would eventually surrender if zapped by enough thought rays. He also believed that even the most rigid, habit-bound people could become imaginative if put in the right situation.

Today, brainstorming is deployed everywhere from kindergarten classrooms to corporate boardrooms. Brainstorming and related techniques help designers define problems and come up with initial concepts at the start of a project. These processes can yield written lists as well as quick sketches and diagrams. They are a handy way to open up your mind and unleash the power of odd-ball notions. *Jennifer Cole Phillips and Beth Taylor*

"The right idea is often the opposite of the obvious."

Alex F. Osborn



Photo: Christian Erickson

How to Brainstorm in a Group

01 Appoint a moderator. Using a whiteboard, big pads of paper, or even a laptop, the moderator writes down any and all ideas. The moderator can group ideas into basic categories along the way. Although the moderator is the leader of the brainstorming process, he or she is not necessarily the team leader. Anyone with patience, energy, and a steady hand can do the job.

02 State the topic. Being specific makes for a more productive session. For example, the topic "new products for the kitchen" is vague, while "problems people have in the

kitchen" encourages participants to think about what they do each day and what they might have trouble with. Breaking the topic down even further (cooking, cleaning, storage) can also stimulate discussion.

03 Write down everything, even the dumb stuff. Everybody in the group should feel free to put out ideas, without censorship. Unexpected ideas often seem silly at first glance. Be sure to record all the boring, familiar ideas too, as these help clear the mind for new thinking. Combine simple concepts to create richer ones.

04 Establish a time limit. People tend to be more productive (and less suspicious of the process) if they know the session won't drag on forever. In addition to setting a time limit, try limiting quantity (a hundred new ways to think about hats). Goals spur people on.

05 Follow up. Rank ideas at the end of the session or assign action steps to members of the group. Ask someone to record the results and distribute them as needed. The results of many brainstorming sessions end up getting forgotten after the thrill of the meeting.

Co-design

Co-design, or co-creation, is a form of design research that engages end users in the process of building a product, platform, publication, or environment. Designers today have learned that users are experts in their own domains. Many designers now view themselves not as controlling an end result but as putting a process into play that actively involves an audience. Co-creation speaks to the rise of do-it-yourself design culture and an empowered consumer base that seeks to use existing products for new purposes.

Whereas interviews (see page 26) and focus groups (see page 30) usually serve to define problems and evaluate results, co-design is a generative technique that involves users and audiences in the creative act of making. Co-design emphasizes user experience as design's ultimate result rather than emphasizing the physical features of an object, website, or other design outcome. Experience is where people find value in goods and services. Given the right tools, nondesigners are well-equipped to envision experiences that will satisfy their needs and desires.

How does it work? In the methodology developed by co-design pioneer Elizabeth B.-N. Sanders, a design team provides a group of potential users with a kit of materials that prompts them to imagine their own solutions to a problem. Whether exploring a car, a phone, a software service, or a hospital room, the co-design process often involves graphic communication. Co-design kits typically include a printed background and a set of materials such as images of generic controls, cut-paper elements, photographs, and tools for making drawings, maps, and collages. These kits often frame open-ended questions, such as what will your school look like in the future? The design team looks for insights and ideas that tap the emotional expectations of users. *Ellen Lupton*

"The new rules call for new tools. People want to express themselves and to participate directly and proactively in the design development process."

Elizabeth B.-N. Sanders



Nokia Open Studios. In the developing world, the adoption of mobile technologies is outpacing that of hard-wired computer and phone systems. Designers from Nokia worked with communities living in informal settlements in Brazil, Ghana, and India. Two hundred twenty co-designers envisioned "dream devices." The participant shown here, a hip-hop dance teacher living in Favela do Jacarezinho in Rio de Janeiro, Brazil, pictured a phone that would diminish violence in her community. Nokia design team: Younhee Jung, Jan Chipchase, Indri Tulusan, Fumiko Ichikawa, and Tiel Attar.

How to Co-design

01 Identify co-designers to collaborate with. If you are creating a product for children, work with kids, teachers, and parents. If you are designing a healthcare solution, work with patients and caregivers. Some researchers suggest collaborating with extreme users: for example, work with people with disabilities (who experience barriers to product use) as well as experts (such as fans, collectors, or repair technicians).

02 Define a question. Your research question should be both concrete and open-ended. Don't predetermine the solution. Instead of asking participants to design a better countertop kitchen mixer, ask them to imagine an ideal kitchen environment.

03 Create a co-design kit. Provide simple tools that invite participants of all skill levels to engage actively and freely.

A co-design kit might include a variety of blank and printed stickers or a set of inspiring words or questions. Sessions can be planned for either individual or group participation.

04 Listen and interpret. Observe how co-designers engage in the process, and study the results of their work. Don't expect picture-perfect products. Instead, learn from people's hopes, desires, and fears.

Holding off on solutions

36

Designing together means solving problems together. With a small group of diverse people—managers, programmers, users, clients—you have to come up with a solution that satisfies everybody. This is often frustrating. For each problem, somebody proposes a solution. Then someone disagrees, or proposes an alternative and the two must face off. Neither wants to back down, because their pride is on the line.

Before long, enough people have given up and group decision making becomes individual decision making. The atmosphere is ruined, and the project is doomed. One of the most effective solutions is **to hold off on proposing solutions**: forbid the mentioning of any solution to any problem. Let people discuss the problem. Who are the stakeholders, what do they want? What have other companies done in this situation? What literature is available? What solutions exist. Is the problem that was tabled the actual problem, or is it limiting our search for solutions? Make sure you understand what everybody wants from a solution and which people have different requirements.

This is the role of the **facilitator**: the leader of the discussion. Usually, the facilitator does not propose solutions or make decisions, he simply listens, and structures the discussion.

Holding off solutions works, because different problems can co-exist peacefully, while different solutions always contradict each other. Consider a team designing a login form for a banking website. The designer wants to make the experience unobtrusive: a password should be entered once, and the session should be kept alive for as long as possible. The security advisor balks at this idea. It's completely insecure. The login should use two-factor authentication: a strong password, entered every session, and a code sent to the user's mobile phone.

Immediately, the two must face off. If they had investigated the problems instead, things would look different: clearly the form must be secure, but the user must also be comfortable. Indeed, if the security measures are too harsh, the user will circumvent them, *reducing* the security. Once the two stakeholders understand each other's aims, they can look for a solution that satisfies both criteria. Perhaps, they can hold off the two-factor authentication until the user wants to transfer money. Perhaps they can show less sensitive information, like the account balance, without even the log-in requirement.

Such elegant solutions, solving both problems, only present themselves when the problem is properly mapped out, and understood by the whole team.

User stories

Holding off on solutions can work very well, but thinking and talking only in terms of solutions is human nature. It takes a strong-willed facilitator to get a group to stick to such a scheme. An easier trick is to tell people to write down solutions, and then forget about them. Make sure that they know they are heard, but then search for other possibilities. This is an essential part of brainstorming, for instance. Write everything down, but don't dwell on anything.

Another method is to translate the solution into the underlying problem that it solves, and to write *that* down: talk about solutions, but use the discussion as a way to map out the problem.

In interaction design, this is commonly done using **user stories**. User stories help you have a constructive discussion, mapping out the exact problem, without making any assumptions about the solution.

Consider the following example: a design team is creating a document editor. The client demands that before closing the editor, a dialog appears, asking the user to save her documents. The designer disagrees: confirmation dialogs are bad practice. The designer suggests saving automatically. The client isn't sure: she may not want to save her changes, she may want to discard them. The programmer notes that the confirmation dialog is the easiest option, and the manager chooses the confirmation dialog. It is only a small problem after all.

There are two problems. First, the team makes a poor choice, and risks an argument if the designer refuses to let the issue go. Second, the decision to implement a confirmation dialog has far reaching implications. For instance, it locks down the decision to build a desktop application, and to allow modal dialogs. This is a recurring problem. No solution affects only the problem under discussion. It has wide ranging implications for every corner of the application.

A user story would help to avoid the problem. It is a simple piece of text that follows these rules:

- It describes a user and a problem that the application should solve for them.
- It makes no assumptions about the application.
- It is no longer than three sentences.

In our example, the user story could be

A user is worried that she'll lose her work by accidentally closing the program.

38

The confirmation dialog is a solution (of sorts) for our problem, but further discussion might produce another user story:

A user is worried that she'll lose her work if the program or the computer crashes.

This user story is not solved by the confirmation dialog at all. The user's objections to the autosave option can also be written down in the form of a user story:

A user would like to go back to a previous version of the document.

The user stories are noted down, and the discussion shifts to collaboration. The client notes that emailing documents back and forth is a messy business. The following user stories are noted:

A manager wants to avoid different versions of the same document, with conflicting texts.

A user would like to review and revert the additions made by another team member.

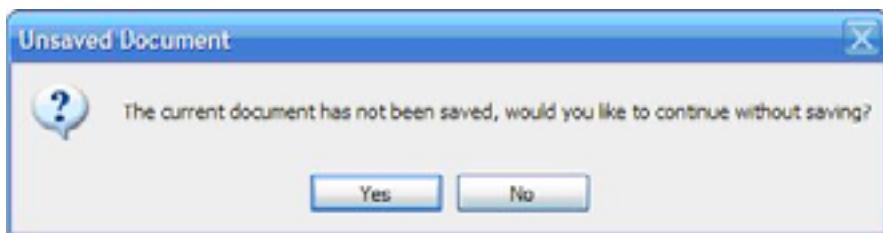
The picture that emerges is that a browser-based editor would allow for simpler collaboration. If such a system would have a good version history, the earlier user stories would be solved as well: documents could be saved automatically, and earlier versions could easily be reviewed.

This solution is much more involved than the simple dialog, of course, but it also solves more of the user's problems and solves them more fundamentally. Suddenly, the programmer and the budget manager can see the value of a version history and an autosave feature.

If the team had locked down the confirmation dialog early on, this picture would not have emerged. The user stories allowed them to have a constructive discussion, without making assumptions about what

their solution would look like.

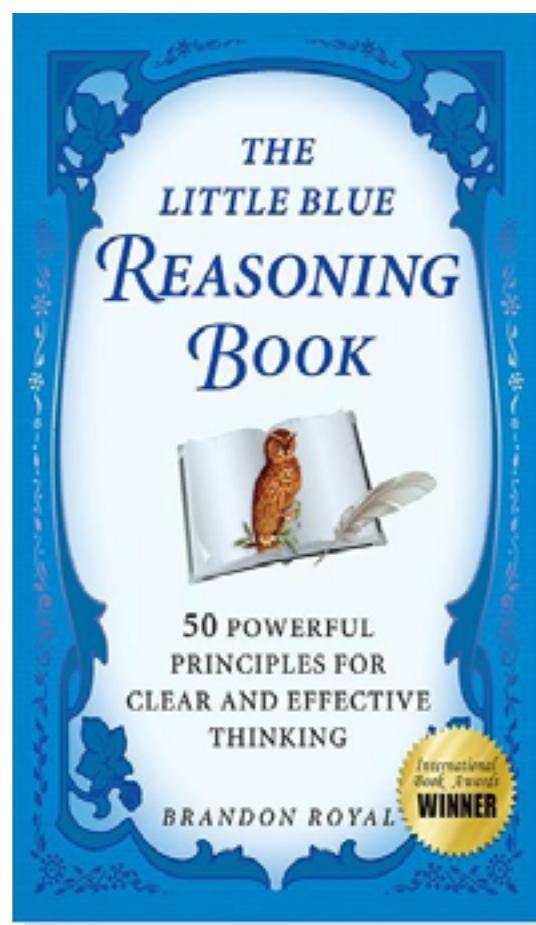
When the problem was fully mapped out, the solution presented itself. User stories let people think in their natural mode, proposing solutions, but stop the discussion from devolving into details. User stories help a team to reach a consensus on the app they're building, and make sure that everybody is working towards the same goal.



The little blue reasoning book

Brandon Royal

“Both **divergence** and **convergence** are necessary for effective problem solving. Divergence opens the mind to creative alternatives; convergence winnows out the weak alternatives, focusing on and choosing among the strong alternatives. Without divergence, we could not analyze a problem creatively or objectively; without convergence, we would just keep on analyzing, never coming to closure.”



The concept of creative thinking is somewhat more difficult to describe than to illustrate. The following story suffices as a noteworthy example.

Many years ago, a hapless merchant owed a substantial sum of money to a wealthy moneylender. Unable to pay back his debt, the merchant knew that the moneylender could see to it that he was put in jail.

The moneylender was old, ugly, and ill-tempered, but he couldn't help but notice how beautiful the merchant's teenage daughter was. He proposed a deal to relinquish the debt and ensure that the daughter would not starve as a result of her father's going to jail.

The moneylender said he would place two small stones, one white and one black, into an empty money-bag and permit the splendid young lady to choose her fate. If she reached into the bag and chose the white stone, her father's debt would be canceled and she would be free from any obligation to marry him. If the stone was black, the father's debt would be canceled, but the young girl would have to marry the moneylender. If she refused to choose, the father would immediately go to jail.

Horrified by their present predicament, the father and daughter knew they could not refuse the moneylender's proposal for debt relief.

Soon the moment of truth had arrived. The three of them met on the garden path of the moneylender's large home. The moneylender bent over to pick up two small rocks. The girl grimaced with fright upon noticing that the moneylender had picked up two small rocks, both of them black, which he now placed in the money bag.

What would you have done if you had been the unfortunate girl? If you had been asked to advise her, what would you have told her to do? You may believe that careful, logical analysis would solve the problem, if there were a solution. This type of traditional, straightforward thinking is not much help to the girl in this story. In this respect, there are but two possibilities:

1. The girl should take a black pebble and sacrifice herself in order to save her father from prison.
2. The girl should refuse to choose a pebble, show that there are

two black pebbles in the bag, expose the moneylender as a cheat, and demand a fair retrial.

This story shows the difference between traditional thinking and creative thinking. Traditional thinkers are concerned with the fact that the girl has to take the pebble and on how the parameters of the "game" are fixed. Creative thinkers are concerned with changing the focus or parameters of the game. Traditional thinkers take the reasonable view of a situation and then proceed logically and carefully to work it out. Creative thinkers tend to explore all the different ways of looking at something, rather than accepting the most promising and proceeding from that.

43

Here's how the story ends:

"Please choose, my fair maiden," the moneylender said. The young woman reached into the money bag and pulled out a rock, which she purposefully let fall to the ground and disappear within the camouflage of the stone path beneath her. "How clumsy of me," she said, while looking toward the money-lender. "No matter though. If you look at the stone held in the bag, we'll be able to tell which color I must have chosen."

With a sense of shock, and with no intention of admitting his dishonesty, the moneylender allowed the girl to reach back into the bag and reveal a black stone. "I chose the white stone!" the girl shouted with joy.

In this way, by using creative thinking, the girl changes what seems an impossible situation into an extremely advantageous one. The girl is actually better off than if the moneylender had been honest and had put one black and one white pebble into the bag, for then she would have had only an even chance of being saved. As it is, she is sure of remaining with her father and at the same time having his debt canceled. Creative thinking might result in two possibilities:

1. Before choosing, the girl should demand the opportunity to change colors (so in choosing a black colored stone, the debt is cancelled and she is set free), even offering to let the moneylender

reach into the bag and choose a stone for her.

- 
2. The girl should choose a stone, fumble it to the ground to conceal its color, and ask to see the color of the remaining stone in the money bag (just as she did in this story).

What types of things most hinder our ability to unleash creative thinking? The answer lies in distinguishing between programmed versus non-programmed responses.

Programmed responses are essential in everyday life, saving us from having to engage in deep thought in order to do routine tasks, e.g., going to the store, driving a car, or saying “hello” while anticipating a familiar response. However, programmed responses form barriers when we encounter novel situations, requiring non-programmed responses.

DIVERGENT VS. CONVERGENT THINKING



Tip #4: Convergent thinking focuses the mind; divergent thinking opens the mind.

At any point in the analytic process, from the very beginning to the very end, we are engaged in one of two thinking modes: convergence or divergence.

Convergence means bringing together and moving toward one point. Whenever we take a narrower view of a problem, focusing our mind on a single aspect of the puzzle, we are in a convergent mode. Whenever we take a broader view of a problem, whether by examining evidence more thoroughly, gathering new evidence, or entertaining alternative solutions, we are in a divergent mode.

While divergent thinking opens the mind to new ideas and thoughts, convergent thinking closes the mind by viewing a problem ever more narrowly until it focuses on — and produces

— a single solution. An apt simile is a camera lens that can zoom in until the subject fills the aperture (convergence) or adjust to broaden the field of view around the subject (divergence). An even more dramatic contrast occurs when using a microscope or telescope.

45

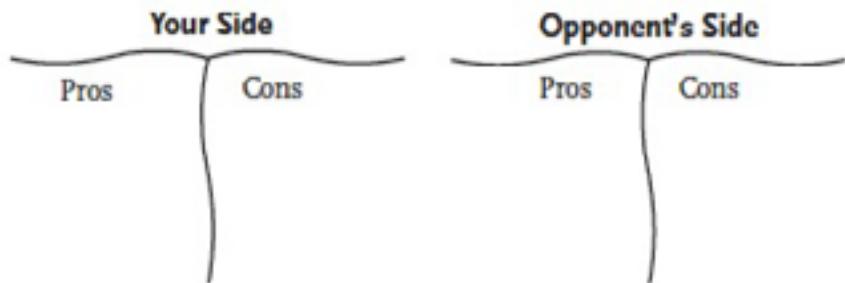
Both divergence and convergence are necessary for effective problem solving. Divergence opens the mind to creative alternatives; convergence winnows out the weak alternatives, focusing on and choosing among the strong alternatives. Without divergence, we could not analyze a problem creatively or objectively; without convergence, we would just keep on analyzing, never coming to closure. It is therefore vital to effective problem solving that an individual be prepared to shift back and forth between divergent and convergent approaches easily and at will, using each mode to its best effect as the problem solving process dictates.

Unfortunately, it is extremely difficult for individuals to shift back and forth between these two ever-opposite, ever-warring approaches. Most of us are inherently better convergers than divergers. Divergence is not as instinctual as is convergence. Indeed, most of us habitually resist divergence — sometimes passionately, even angrily.

DEVIL'S ADVOCATE TECHNIQUE



Tip #5: The devil's advocate technique imposes objectivity and compels divergent thinking.



Definition: Devil's advocate – a person who advocates an opposing or unpopular cause for the sake of argument or to expose it to a thorough examination.

The devil's advocate technique forces us to consider the merits of the “other side” of an issue or topic. What we actually do is act as if we believe our opponent's side is right! In this way, we can gain a greater degree of objectivity. This technique is excellent for use in negotiations because it forces us to understand the other party's position and leads to a more realistic, effective bargaining process.

Consider that you are an analyst working at a leading market research firm. Your objective is to write a report on the market economics for product A. You have a strong idea that the market for product A is becoming more price-sensitive, and that the greater variations in price observed in the market are due to widening differences in product quality and branding (or perceptions of that product's quality or brand strength). You need to confirm your suspicion, and you're off to an interview with the marketing director of a major company who is responsible for marketing product A. But wait! If you and other analysts hold a

similar view then you risk simply confirming something you already believe. Play the devil's advocate. Go into the meeting and ask questions designed to "disconfirm" what you think is true. You might ask, "So, is it true that the market for product A is becoming less price-sensitive?" The responses elicited may be essential to gaining a more complete understanding of the situation.

47

IDEA KILLERS AND IDEA GROWERS



Tip #6: Not challenging the obvious, evaluating ideas too quickly, and fear of looking the fool — these are the three greatest creativity inhibitors.

Not challenging the obvious

Creativity may suffer whenever we, as individuals, accept the status quo. We have to challenge the obvious. "Does one plus one really equal two?" It could indeed equal two. But it might equal eleven, as in "1 + 1 = 11." Or it could equate to "T," the result of placing one bar on top of the other.

Management consultants are constantly faced with the need to challenge the obvious. For instance, a client calls the consultant in and says, "Profitability is down because product costs are too high. Can you help me find a way to reduce them?" The consultant will instinctively challenge the obvious, asking whether it is the case that costs are too high. Perhaps it is another factor in the profitability mix (i.e., price or unit sales) that is really to blame.

Evaluating ideas too quickly

One way of confronting this barrier is to look at your hands. Think of your right hand as representing "idea production" and your left hand as representing "idea evaluation." Often an idea produced is

immediately evaluated and possibly killed, e.g., by the phrase, “That won’t work.” Success in creative thinking demands that the two hands should be separated, and that the left hand (idea evaluator) should be put to one side for the moment.

All ideas are acceptable in a creative situation, regardless of the quality. They may be good, bad, useful, useless, legal, illegal — it doesn’t matter. Subsequently, the evaluation hand is brought back, and at that stage, a strange thing happens. Some of the ideas that originally would have been dismissed are looked at afresh, possibly with the comment: “Wait a minute, there may be something in that idea after all.”

Fear of looking the fool

Failing to challenge the obvious and evaluating ideas too quickly may well be the by-products of being afraid to look like a fool. We learn to fear ridicule from an early age and it follows us into later life. Many excellent examples are found in the world of management. In a hierarchical organization, junior team members are less likely to put forth wild, wacky ideas for fear that more senior team members will see them as silly. The junior does not want to destroy his or her chances of promotion, and therefore sticks to well-tried, analytical routines. At the other end of the scale, the most senior manager seeks to protect his image — one that has been built up over many years. That senior manager doesn’t want to confirm to his or her underlings that he or she is a silly old fool. As a consequence, he or she does not propose any wild ideas either.

In short, we must fight apathy, hastiness, and insecurity. History abounds with instances of people who haven’t been proactive enough in evaluating new ideas or who have been overly dismissive of new inventions or artistic or literary styles. This is particularly true where individuals are deemed authorities in their fields and err on the side of protecting their reputations. Here are several examples taken from the domain of science and art.

- Walt Disney was fired by a newspaper editor of the Kansas City Star newspaper because “he lacked imagination and had no good ideas.” Years later, the Disney company bought ABC which owned the Kansas City Star.

49

- Although Vincent van Gogh produced some 800 paintings, he was able to sell only one painting during his lifetime. The *Red Vineyard at Aries* was sold to the sister of one of his friends for 400 francs (approximately \$50).
- In 1921, Newton Baker, U.S. Secretary of War, reacted to Brigadier General Billy Mitchell’s claim that airplanes could sink battleships by dropping bombs on them: “That idea is so damned nonsensical and impossible that I’m willing to stand on the bridge of a battleship while that nitwit tries to hit it from the air.”
- “Can’t act. Can’t sing. Can dance a little.” MGM summary of a screen test of some guy named Fred Astaire, 1928.
- A Paris art dealer refused Pablo Picasso shelter when he asked if he could bring in his paintings from out of the rain.
- “I have traveled the length and breadth of this country and talked with the best people, and I can assure you that data processing is a fad that won’t last out the year.” The editor in charge of business books for Prentice Hall, 1957.
- “We don’t like their sound and guitar music is on the way out.” Decca Recording Co. on rejecting the Beatles, 1962.
- “But what … is it good for?” Engineer at IBM’s Advanced Computing Systems Division, 1968, commenting on the microchip.
- Madonna, the best-selling female rock artist of the 20th century, was rejected by several music labels in the early 1980s. One talent agent is reputed to have said that her voice wasn’t unique enough to stand out in a crowded marketplace.

- 
- In the early 1990s, J.K. Rowling's *Harry Potter and the Philosopher's Stone* was rejected by more than a dozen UK publishers, the majority of which believed that the story wasn't mainstream enough.

The process of making mistakes in judgment and/or missing opportunities can be further illustrated within the framework of Type I and Type II errors. These two types of errors also are discussed within the topic of Hypothesis Testing in *Chapter 3*.

Type I errors are really errors of commission, while Type II errors are errors of omission. Type I errors are the result of projects that we should have rejected but instead adopted. Type II errors are the result of projects that we should have adopted but instead neglected. Type I errors result in observable failures. Type II errors result in missed opportunities.

A Type I error occurs when we take an action and it turns out to be a mistake. For example, whenever a top movie executive "green-lights" (okays) a movie project that turns out to be a failure, a Type I error is committed. The executive's career could suffer in a very public way, as these kinds of errors are very visible.

A Type II error occurs when we don't take an action, and the mistake comes from missing an opportunity. If one movie executive passes on a decision to make a movie, and another movie house later produces it, turning it into a blockbuster, a Type II error is committed.

Type II errors are often hard to see, even if they are common. The problem is that most Type II errors are never discovered. This is because many opportunities never immediately resurface. Projects or ideas, once killed or shelved, seldom get a second opinion. They are stopped without being shown to other people (or organizations) to see if someone else wants to take on the risk to pursue them.

Because Type II errors are mostly invisible, they come at less initial cost to people and organizations than do Type I errors. It's often easier to say no to something that might be a huge success than it is to say yes, because most of the time, no one will ever know what the outcome might have been. As long as most individuals (and the departments or organizations they work for) are evaluated based on the outcomes of their decisions, and not on what opportunities they might have missed, Type II errors will never be fully monetized.



Tip #7: Keep a mental list of idea "killers" and idea "growers."

Idea Killers

It would cost too much.

We tried it before.

That's not my job.

That's not your job.

That's not how we do it.

Why don't you put that in writing?

It's impossible.

That sounds crazy to me.

You may be right, but ...

Maybe next year.

If it ain't broke, don't fix it.

It would take too much time.

Our customer would never go for that.

My mind is definitely made up.

I don't think that's important.

Our company is too small.

Our company is too big.

It's good enough.

We don't have time right now.

That's a stupid idea.

I don't need any more information.

You can't do that here.

Idea Growers

Before we make a final decision, let's review all the options.

Are there any questions?

Where else can we go for additional information on that?

May I ask a question?

What would happen if ... ?

In light of the new information, I've changed my mind.

53

How could we improve ... ?

I'd like to get your help with an idea I'm working on.

Let me ask you for some ideas on ...

Is this what you meant?

Who else would be affected?

What have we missed?

Who else has a suggestion?

I don't know much about that. How about you?

Why do we always do it like that?

Wouldn't it be fun if ... ?

What ideas have you come up with?

How many ways could we ... ?

Thank you!

BRAINSTORMING



Tip #8: Brainstorming has rules: quantity of ideas is preferred, wacky ideas are welcomed, delayed evaluation is mandatory, and "hitchhiking" is encouraged.

Ideas are the lifeblood of creativity, and brainstorming is a method to generate ideas. Brainstorming sessions are usually conducted in a group of between six and fifteen people. The setting is a room equipped with a whiteboard (or flip chart) so that ideas can be written down. The goal of brainstorming is to produce "novel but appropriate" ideas — the very heart of creativity. To achieve this goal, one must adhere to the "rules" of brainstorming.

First, quantity of ideas is the primary objective. Ideas should flow right from participants' tongues to the whiteboard. Second, to get people to come up with truly novel ideas, we say "wackier is better." Let the ideas flow by themselves. No one should fear looking the fool. All ideas, however wild or silly, are accepted. Third, delayed evaluation is mandatory. It is contradictory to try to create ideas and evaluate them at the same time. Any such attempt will curtail the creative process.

Fourth, as the session progresses, people will naturally "hitchhike" on ideas. "Oh that idea reminds of this" and "If that is so, then how about ..." Hitchhiking means that one person is able to use another person's idea to go further and supply another idea. Toward the end of the brainstorming session, ideas will be scattered haphazardly from one end of the whiteboard to the other. This is perfectly natural. This may cause some participants to giggle or burst out laughing because very rarely does anyone experience this kind of free-flowing activity, especially in an office environment. Once ideas are regrouped and summarized, the results may be truly surprising. Managers, for example, who are unfamiliar with the power of brainstorming sessions are typically amazed at how many commercially viable ideas, that have never been previously uncovered, exist in the "collective mind" of their staff members and employees.

business? (At this stage, let your imagination run wild: wacky, impractical ideas are as welcome as practical ones. Again, quantity over quality: the list can be whittled down later.)

55

REFRAMING PROBLEMS



Tip #9: Consider whether a problem is really the problem. Think in terms of redefining the problem.

Ponder the following problem: “A restaurant is losing customers because customers are annoyed at how long it takes to line up outside in order to get a seat inside the restaurant.”

If you were hired as a consultant, reporting to the headquarters of the restaurant chain, what would you suggest?

Typical solutions to be anticipated include:



Enlarging the restaurant facilities in order to serve more customers



Streamlining the menu in order to make ordering and delivery of food faster



Refusing to let customers occupy tables if not ordering food; no “drinking-only” tables

These are all potential solutions. Nevertheless, they address only one of a number of possible general objectives: to speed up the process of getting customers through the dining process. An alternative goal is to find ways to keep people from getting annoyed at lining up. This suggests a host of potential strategies,

such as installing televisions that customers could watch while they wait for a table, giving them free snacks while they wait in line, conducting market research while they wait in line, or having live or videotaped entertainment (e.g., magicians) to amuse persons in the line.

Still another objective is to keep the restaurant from having too many customers at one particular time of day. One idea/strategy would be to get more of the regular restaurant customers to come at non-peak hours. This might be accomplished by giving special dinner or drink discounts during certain hours of the day or holding special promotional events, such as corporate cocktail parties, speaking engagements, book signings, and guitar solos.

It is rare for people to step back and try to define alternative goals. Instead, most people read or hear of a problem and almost immediately begin generating strategies. One way to become more creative is by explicitly defining a minimum of two or three different goals for each problem situation.

Here's another example: An agricultural importer's association was attempting to seek a way to reduce the number of bruised pears which occurred when these fruits were transported.

The importers initially defined their goal as "decreasing the rate with which pears became bruised or damaged when shipped." This led to various strategies for modifying distribution systems and packing procedures, such as including more padding around the pears and using smaller packing boxes. Although all of these strategies provided partial solutions, none was considered a breakthrough.

Reframing the problem led to a new goal: "creating a pear that is less likely to be bruised!"

This entailed hiring individuals to look into the process of breeding pears. By exploring strategies to modify the pear, a portion of the problem was eventually solved. An "apple-pear"

was born — a fruit with some of a pear's taste but with an apple's sturdiness. Now grocery stores could be supplied with large quantities of unblemished pear hybrids.

57

Get into the habit of asking if the problem really is the problem. Is the goal really the goal?

Getting real

37signals

37signals is an angry little company. They hate the way things are done. They hate functional documents, use cases, information architecting, UML diagramming, brainstorming, design meetings. Even wireframes and user testing. And they're not entirely wrong.

Too much process, too many fixed methods, with too little understanding will suck the joy out of any project. If you get a small team of people who understand what they're doing, and who they're designing for, there will be no need for any of these methods. Ultimately, you have to make up your own mind about which methods work for you. *Getting Real* is just one of the voices in the discussion, albeit a particularly clear and inspiring one.

The full book is available for free at
gettingreal.37signals.com



Have an Enemy

60

Pick a fight

Sometimes the best way to know what your app should be is to know what it shouldn't be. Figure out your app's enemy and you'll shine a light on where you need to go.

When we decided to create project management software, we knew Microsoft Project was the gorilla in the room. Instead of fearing the gorilla, we used it as a motivator. We decided Basecamp would be something completely different, the anti-Project.

We realized project management isn't about charts, graphs, reports and statistics – it's about communication. It also isn't about a project manager sitting up high and broadcasting a project plan. It's about everyone taking responsibility together to make the project work.

Our enemy was the Project Management Dictators and the tools they used to crack the whip. We wanted to democratize project management – make it something everyone was a part of (including the client). Projects turn out better when everyone takes collective ownership of the process.

When it came to Writeboard, we knew there were competitors out there with lots of whizbang features. So we decided to emphasize a “no fuss” angle instead. We created an app that let people share and collaborate on ideas simply, without bogging them down with non-essential features. If it wasn't essential, we left it out. And in just three months after launch, over 100,000 Writeboards have been created.

When we started on Backpack our enemy was structure and rigid rules. People should be able to organize their information their own way – not based on a series of preformatted screens or a plethora of required form fields.

One bonus you get from having an enemy is a very clear marketing message. People are stoked by conflict. And they also understand a product by comparing it to others. With a chosen enemy, you're feeding people a story they want to hear. Not only will they understand your product better and faster, they'll take sides. And that's a sure-fire way to get attention and ignite passion.

Now with all that said, it's also important to not get too obsessed with the competition. Overanalyze other products and you'll start to limit the way you think. Take a look and then move on to your own vision and your own ideas.

Don't follow the leader

Marketers (and all human beings) are well trained to follow the leader. The natural instinct is to figure out what's working for the competition and then try to outdo it – to be cheaper than your competitor who competes on price, or faster than the competitor who competes on speed. The problem is that once a consumer has bought someone else's story and believes that lie, persuading the consumer to switch is the same as persuading him to admit he was wrong. And people hate admitting that they're wrong.

Instead, you must tell a different story and persuade listeners that your story is more important than the story they currently believe. If your competition is faster, you must be cheaper. If they sell the story of health, you must sell the story of convenience. Not just the positioning x/y axis sort of "We are cheaper" claim, but a real story that is completely different from the story that's already being told.

–Seth Godin, author/entrepreneur (from Be a Better Liar)

Make Opinionated Software

62

Your app should take sides

Some people argue software should be agnostic. They say it's arrogant for developers to limit features or ignore feature requests. They say software should always be as flexible as possible.

We think that's bullshit. The best software has a vision. The best software takes sides. When someone uses software, they're not just looking for features, they're looking for an approach. They're looking for a vision. Decide what your vision is and run with it.

And remember, if they don't like your vision there are plenty of other visions out there for people. Don't go chasing people you'll never make happy.

A great example is the original wiki design. Ward Cunningham and friends deliberately stripped the wiki of many features that were considered integral to document collaboration in the past. Instead of attributing each change of the document to a certain person, they removed much of the visual representation of ownership. They made the content ego-less and time-less. They decided it wasn't important who wrote the content or when it was written. And that has made all the difference. This decision fostered a shared sense of community and was a key ingredient in the success of Wikipedia.

Our apps have followed a similar path. They don't try to be all things to all people. They have an attitude. They seek out customers who are actually partners. They speak to people who share our vision. You're either on the bus or off the bus.

Hold the Mayo

63

Ask people what they *don't* want

Most software surveys and research questions are centered around what people want in a product. “What feature do you think is missing?” “If you could add just one thing, what would it be?” “What would make this product more useful for you?”

What about the other side of the coin? Why not ask people what they don't want? “If you could remove one feature, what would it be?” “What don't you use?” “What gets in your way the most?”

More isn't the answer. Sometimes the biggest favor you can do for customers is to leave something out.

Innovation Comes From Saying No

[Innovation] comes from saying no to 1,000 things to make sure we don't get on the wrong track or try to do too much. We're always thinking about new markets we could enter, but it's only by saying no that you can concentrate on the things that are really important.

-Steve Jobs, CEO, Apple (from The Seed of Apple's Innovation)

Wordsmiths

Hire good writers

If you are trying to decide between a few people to fill a position, always hire the better writer. It doesn't matter if that person is a designer, programmer, marketer, salesperson, or whatever, the writing skills will pay off. Effective, concise writing and editing leads to effective, concise code, design, emails, instant messages, and more.

That's because being a good writer is about more than words. Good writers know how to communicate. They make things easy to understand. They can put themselves in someone else's shoes. They know what to omit. They think clearly. And those are the qualities you need.

An Organized Mind

Good writing skills are an indicator of an organized mind which is capable of arranging information and argument in a systematic fashion and also helping (not making) other people understand things. It spills over into code, personal communications, instant messaging (for those long-distance collaborations), and even such esoteric concepts as professionalism and reliability.

-Dustin J. Mitchell, developer

Clear Writing Leads To Clear Thinking

Clear writing leads to clear thinking. You don't know what you know until you try to express it. Good writing is partly a matter of character. Instead of doing what's easy for you, do what's easy for your reader.

*-Michael A. Covington, Professor of Computer Science at The University of Georgia
(from *How to Write More Clearly, Think More Clearly, and Learn Complex Material More Easily*)*

Interface First

65

Design the interface before you start programming

Too many apps start with a program-first mentality. That's a bad idea. Programming is the heaviest component of building an app, meaning it's the most expensive and hardest to change. Instead, start by designing first.

Design is relatively light. A paper sketch is cheap and easy to change. HTML designs are still relatively simple to modify (or throw out). That's not true of programming. Designing first keeps you flexible. Programming first fences you in and sets you up for additional costs.

Another reason to design first is that **the interface is your product**. What people see is what you're selling. If you just slap an interface on at the end, the gaps will show.

We start with the interface so we can see how the app looks and feels from the beginning. It's constantly being revised throughout the process. Does it make sense? Is it easy to use? Does it solve the problem at hand? These are questions you can only truly answer when you're dealing with real screens. Designing first keeps you flexible and gets you to those answers sooner in the process rather than later.

The Orange Pen That Started Blinksale

As soon as I realized my frustration with off-the-shelf invoicing software, I decided to draw out how I would prefer my invoicing solution to work. I pulled out an orange pen, because it was the only thing handy that evening, and had about 75 percent of the UI drawn out within a few hours. I showed it to my wife, Rachel, who was ironing at the time, and asked, “What do you think?” And she replied with a smile, “You need to do this. For real.”

66

Over the next two weeks I refined the designs, and completely mocked-up static HTML pages for almost the entire first version of what would become Blinksale. We never did any wireframes beyond those orange-pen sketches, and getting straight into the HTML design helped us stay excited about how “real” the project was becoming, even though at the time we really didn’t know what we were getting into.

Once the HTML mockups were completed, we approached our developer, Scott, with the idea for Blinksale. Having most of the UI designed up front was extremely beneficial on several levels. First, it gave Scott a real vision and excitement for where we were going. It was much more than just an idea, it was real. Second, it helped us accurately gauge how much of Scott’s effort and time it would require to turn the design into a functioning application. When you’re financially bootstrapping a project, the earlier you can predict budget requirements, the better. The UI design became our benchmark for the initial project scope. Finally, the UI design served as a guide to remind us what the application was about as we progressed further into development. As we were tempted to add new features, we couldn’t simply say, “Sure, let’s add that!” We had to go back to the design and ask ourselves where that new feature would go, and if it didn’t have a place, it wouldn’t get added.

-Josh Williams, founder, Blinksale

Copywriting is Interface Design

67

Every letter matters

Copywriting is interface design. Great interfaces are written. If you think every pixel, every icon, every typeface matters, then you also need to believe every letter matters. When you're writing your interface, always put yourself in the shoes of the person who's reading your interface. What do they need to know? How you can explain it succinctly and clearly?

Do you label a button **Submit** or **Save** or **Update** or **New** or **Create**? That's copywriting. Do you write three sentences or five? Do you explain with general examples or with details? Do you label content **New** or **Updated** or **Recently Updated** or **Modified**? Is it **There are new messages: 5** or **There are 5 new messages** or is it **5** or **five** or **messages** or **posts**? All of this matters.

You need to speak the same language as your audience too. Just because you're writing a web app doesn't mean you can get away with technical jargon. Think about your customers and think about what those buttons and words mean to them. Don't use acronyms or words that most people don't understand. Don't use internal lingo. Don't sound like an engineer talking to another engineer. Keep it short and sweet. Say what you need to and no more.

Good writing is good design. It's a rare exception where words don't accompany design. Icons with names, form fields with examples, buttons with labels, step by step instructions in a process, a clear explanation of your refund policy. These are all interface design.

Tell Me a Quick Story

68

Write stories, not details

If you do find yourself requiring words to explain a new feature or concept, write a brief story about it. Don't get into the technical or design details, just tell a quick story. Do it in a human way, like you would in normal conversation.

It doesn't need to be an essay. Just give the flow of what happens. And if you can include the brief story in context with screens you are developing, all the better.

Stick to the experience instead of getting hung up on the details. Think strategy, not tactics. The tactics will fall into place once you begin building that part of your app. Right now you just want to get a story going that will initiate conversation and get you on the right track.

Designing for the social web

Joshua Porter

“During [the twentieth] century we have for the first time been dominated by non-interactive forms of entertainment: cinema, radio, recorded music and television. Before they came along all entertainment was interactive: theatre, music, sport—the performers and audience were there together, and even a respectfully silent audience exerted a powerful shaping presence on the unfolding of whatever drama they were there for. We didn’t need a special word for interactivity in the same way that we don’t (yet) need a special word for people with only one head.

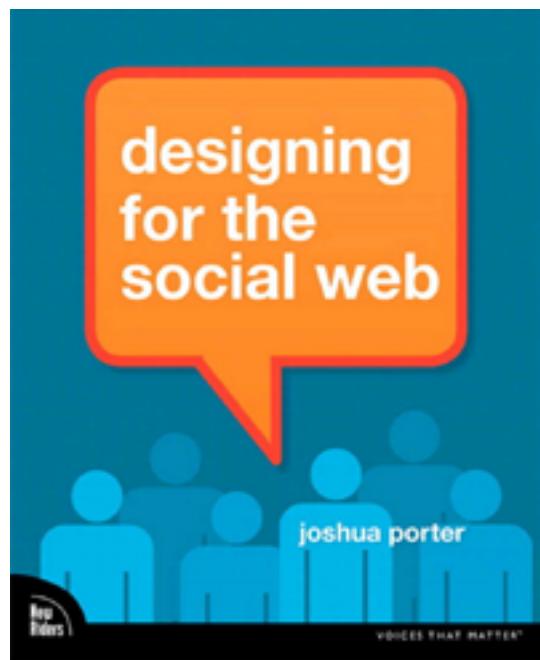
I expect that history will show ‘normal’ mainstream twentieth century media to be the aberration in all this. ‘Please, miss, you mean they could only just sit there and watch? They couldn’t do anything? Didn’t everybody feel terribly isolated or alienated or ignored?’

‘Yes, child, that’s why they all went mad. Before the Restoration.’

‘What was the Restoration again, please, miss?’

‘The end of the twentieth century, child. When we started to get interactivity back.’ ”

—Douglas Adams, *How to Stop Worrying and Love the Internet* (1999)



Ten Ways Flickr Builds Communities¹⁰

1. **Engage.** Don't just listen to your community
2. **Enforce.** Let the community help set standards and policies for appropriate behavior—then enforce them
3. **Take Responsibility.** Fess up immediately when you make mistakes
4. **Step Back.** Don't be afraid to step back and let your customers take over
5. **Give Freely.** Never underestimate the allure of a free T-shirt (or sticker, or button...)
6. **Be Patient.** Take knee-jerk reactions with a grain of salt
7. **Hire Fans.** Make sure your employees are as passionate about your product as your community's most die-hard fans
8. **Stay Calm.** Develop a thick skin
9. **Focus.** Be flexible but don't lose sight of your priorities
10. **Be Visible.** Stay human



73

The Rise of the Social Web

A social and economic change that has barely begun

“The Web is more a social creation than a technical one. I designed it for a social effect—to help people work together—and not as a technical toy. The ultimate goal of the Web is to support and improve our weblike existence in the world. We clump into families, associations, and companies. We develop trust across the miles and distrust around the corner. What we believe, endorse, agree with, and depend on is representable and, increasingly, represented on the Web. We all have to ensure that the society we build with the Web is of the sort we intend.”

—TIM BERNERS-LEE, WEAVING THE WEB¹

¹ <http://www.w3.org/People/Berners-Lee/Weaving/>

The Amazon Effect

If you've ever watched someone shop at Amazon.com, you may have witnessed the Amazon Effect.

I first saw the Amazon Effect during a usability study several years ago. I was observing a person shopping for a digital camera recommended to her by a friend. As part of the testing procedure, I asked the shopper to go to CircuitCity.com and try to buy the camera. She started typing the URL, then stopped.

Shopper: *Can I go to Amazon first?*

Me: *No.*

Shopper (frowning): *Well, I always go to Amazon first. I love Amazon.*

Unfortunately, our testing methodology didn't allow for that. We couldn't let people shop just anywhere. We were testing very specific sites at the request of our client. Though we were testing Amazon in the study, we weren't testing Amazon with this particular shopper.

Me: *I'm sorry. I can't let you go there just now. But let me ask: why do you want to go to Amazon?*

Up to that point, we'd had a couple of people ask to visit Amazon in the test and had assumed they kept asking because they had accounts there. We figured they had previously shopped at Amazon and had a history with the company, had created wish lists and purchase histories there, and were generally more comfortable shopping in a familiar environment. We assumed the familiarity of Amazon was what kept them coming back.

But as with so many assumptions, it was wrong.

Shopper: *I go to Amazon to do research on a product I'm shopping for, even when I plan to buy it on another site.*

Me: *Even when you plan to buy it on another site?*

Shopper: *Yes, of course.*

Wow! This wasn't what we had expected. People wanted to go to Amazon so badly to do *product research*, not because they had an account there. The magnetic pull of Amazon, what I like to call the *Amazon Effect*, was entirely different from what we had assumed.

People-Powered Research

So why the pull of Amazon versus, say, another online electronics retailer? Didn't Amazon have the same information as other sites? Weren't they basically all selling the same cameras? What does Amazon do that others don't?

The answer becomes clear almost immediately when watching someone shopping: *customer reviews*.

At Amazon, customer reviews act like a magnet, pulling people down the page. That's the content people want. The page loads, the viewer starts to scroll. They keep scrolling until they hit the reviews, which in some cases are up to 6000 pixels down from the top of the page! Nobody seems to mind. They simply scroll through screens and screens of content until they find what they're looking for.

During a test a few days later, another shopper exhibited a distinctive behavior. He went to the reviews and immediately sorted them to bring the 1-star reviews to the top of the list. This meant they wanted to see the negative reviews first.

Me: *Why did you do that?*

Shopper: *Well, I want to make sure I'm not buying a lemon.*

Another shopper, who exhibited the same behavior of going directly for the reviews, told me why they rarely look at the other content on the page — the wealth of content like the manufacturer's description and other product information.

Shopper: *I already know what it's going to say, it's going to tell me how great their product is. Why would I need to read that? If I want to know the truth, I have to read what other people like me thought about it.*

There it was: a crystallization of the value of customer reviews. Customer reviews allow people to learn about a product from the experience of others without any potentially biased seller information. No wonder



Figure 1.1 Amazon's product pages are extremely long, but that doesn't keep people from scrolling almost the entire length of them to find the customer reviews.

everyone wanted to shop at Amazon. They had information that no other site had: they had *the Truth*.

And that truth, interestingly enough, arose from simply aggregating the conversation of normal people like you and me.

Counter-Intuitive Economics

Let's take a bird's-eye view of what's happening at Amazon. Consider these peculiarities:

- ▶ **Amazon doesn't always provide the most valuable information on their site.** Instead, the people writing reviews contribute valuable information others are looking for. Amazon simply provides the tool with which to write the reviews.
- ▶ **People write reviews without getting paid.** There is no monetary reward for writing reviews. Yet dozens of reviewers have written over a thousand reviews each! These folks know they aren't going to get paid, but do it anyway.
- ▶ **People are not being managed in any tangible way.** This incredible outpouring of reviews is not being managed. Individuals are acting independently of one another and together provide an amazing resource.
- ▶ **People pay attention to strangers they'll never meet.** Yet, they still take the time to help out these strangers by describing their experience with a product.
- ▶ **People police each other.** In addition to taking the time to write reviews, people also help judge whether they found a given review helpful, thereby weeding out the bad (by pushing them to the bottom).
- ▶ **People openly identify themselves.** Even in this most public of places, where anybody could see what they're doing, most people freely identify themselves.

Given our common conception of how to get people to do work, many of these points are counter-intuitive. We've been taught that hard work is rewarded by an honest wage, yet people at Amazon are working for free. People aren't supposed to work for free. The value of customer reviews flies in the face of how economics is supposed to work!

The models that economists have created assume there must be an incentive for production, in plain terms *money*. So how could Amazon create such a large, stable, *valuable* system without paying any of their contributors even a penny for their efforts?

The conclusion we must reach is staring us in the face:

Amazon's reviews are about much more than money.

Indeed, the overwhelming success of Amazon's reviews is evidence of a way in which the web has produced a dramatic change in the world's economy. In traditional economic terms the mere existence of reviews just doesn't compute. Few existing economic models can accurately describe the value being given (or received) on Amazon.

Yochai Benkler, author of *Wealth of Networks*, a wonderful book describing these new economic changes in detail, notes:

A new model of production has taken root; one that should not be there, at least according to our most widely held beliefs about economic behavior.

It should not, the intuitions of the late-twentieth-century American would say, be the case that thousands of volunteers will come together to collaborate...

It certainly should not be that these volunteers will beat the largest and best-financed business enterprises in the world at their own game.

And yet, this is precisely what is happening...²

The Social Web

Of course Amazon isn't the only one designing for and supporting the activity of its audience in this way: it is merely one of countless examples of social design on the web. For the purposes of this book, we define social design in the following way:

Definition: Social design is the conception, planning, and production of web sites and applications that support social interaction

² Yochai Benkler, *The Wealth of Networks*, Yale University Press, 2006.

We've barely seen the tip of the iceberg when it comes to designing social software. I'm confident we'll be discussing social software (and how to design it) for decades to come. It is the future of the web. Here are several reasons why:

1. **Humans are innately social.** Since humans are social, it makes sense that our software will be social, too.
2. **Social software is a forced move.** The sheer amount of information and choice we're faced with forces us toward authentic conversations (and tools to help us find and have them).
3. **Social software is accelerating.** Social software is trending upward: it is already the fastest growing and most widely used software on the web. The future suggests more of the same.

Let's take a look at each of these reasons in depth to get a clearer picture of the rise of the social web.

Humans Are Innately Social

Humans are innately social creatures. We exhibit *social behavior*. If we did not, if we weren't social from the day we are born, then social software would be incongruous: it just wouldn't make sense. Instead of garnering our attention and energy, Amazon, eBay, and MySpace would be worthless.

While most of us would agree that we are social by nature, what exactly does it mean to be social? Well, social is a fuzzy term, and most dictionaries define it as something to do with "group formation" or "living together."³ But those terms don't illustrate the richness of our social lives. Being social is more than merely forming groups: it's all the interactions, decisions, and conversations that happen in and around those groups!

It includes, but certainly isn't limited to:

Sharing, caring, feeding, loving, fighting, conversing, friendship, sex, envy, shouting, arguing, betrayal, rumor mongering, gossiping, laughing, crying, providing support, whining, advocating for others, recommending, swearing off.

³ For example, the dictionary on my Mac says: "of or relating to the aggregate of people living together in a more or less ordered community" (this is not very helpful).

Key Aspects of Social Behavior

1. Humans are complex social animals who interact with each other for almost every need: food and water, shelter, technology, friendship, learning, fun, sex, ritual, sport
2. Humans organize themselves into groups, often belonging to multiple groups at the same time
3. Groups can be as small as two people or as large as a religion, and can be for any purpose
4. Groups can be made up of family, friends, acquaintances, or any set of people with something in common
5. Humans act as both group members and individuals at the same time
6. Humans behave differently in groups than they do individually, and vice-versa
7. Humans play different roles in different parts and periods of their lives
8. When humans are uncertain, they rely on social connections to help them out
9. People usually compare themselves to those in their social group, not to society at large
10. The people we know greatly influence how we act
11. Sometimes being self-interested means to support the group, sometimes it means to diverge from the group and focus on oneself
12. Humans aren't always rational, but usually behave in a self-interested manner
13. Unpredictable behavior emerges within groups over time
14. People derive enormous value from social interaction that cannot be accounted for in monetary terms

Lewin's Equation

The mere fact that we as humans organize ourselves into groups isn't all that special. After all, other animals form groups. But as this list shows, being in groups, and being around groups, and *not being in groups* really changes the way we behave.

We didn't always think this way. In 1933, German behavioral psychologist Kurt Lewin, escaping Hitler's rise to power, emigrated to America in order to continue his studies on group behavior. At that time, the commonly held notion about human behavior was that we act according to our

personality. Sigmund Freud and his theories on the unconscious mind were in vogue. Most of the prevailing research assumed in one way or another that our inborn tendencies dictated our behavior.

But Lewin's research said different. He challenged the prevailing wisdom by formulating a simple yet profound statement to describe human behavior. The statement, which was expressed as an equation, of all things, thrust Lewin to the forefront of an emerging field. Indeed, Lewin is often called "the father of social psychology."

This is Lewin's equation:

$$B = f(P, E)$$

The equation says that an individual's behavior is a function of both their personality *and* their environment. While the classic nature vs. nurture debate asks you to take sides, Lewin's equation does not: it invitingly allows for both the person and their environment to affect what happens in a complex, yet profound, way.

From Environment to Interface Design

Lewin's equation highlights the tension between the individual and the environment. The environment, of course, is basically made up of everything that isn't us. That's an awfully big set of things to think about! However, we easily recognize several types of environments. One is the *physical environment*, which has a tremendous effect on what we do. When it's cold outside, we must put clothes on or suffer the consequences.

Other people and groups make up our *social environment*. And, perhaps even as much as the weather dictates how we dress, the actions of others affect how we behave. Imagine how many of our decisions are strongly influenced by what other people say or do. Just as the friend who made a product recommendation to our shopper on Amazon influenced her behavior, so we are profoundly influenced by the people we know and the groups we join.

In the software world there is even another kind of environment: the *software interface*.

The interface is the environment in which people work and play on the web. It is the arbiter of all the communication and interaction that takes place there. If there is an action available in an interface, then you can

perform the action. If an action is not available in an interface, then you're out of luck. While we are intuitively aware of this, just as we are aware of the weather, we rarely reflect on how much our behavior is determined by the interfaces we use. Almost all of it!

This sounds like the designers of the interface are in control! Not so fast. Designing an interface that evokes the desired behavior is a huge challenge.

If the interface is too confining, people won't use it.

If the interface is too flexible, people won't know how to use it.

In the middle, the sweet spot, interface designers can create powerful social software that supports the person and their personality, as well as the social environment and the groups they are a part of.

The Challenge of Social Software

Thus the challenge of social software is to design interfaces that support the current and desired social behavior of the people who use them.

Designing an effective interface has always been tough, even when we were merely designing interfaces for one person to interact with content we controlled. But when we add the social aspect, things get even more difficult. Though we can see glimpses, we have little understanding of the overall effect of social software going forward. In 1985, Howard Rheingold, writing about the nascent personal computer revolution, foresaw social software's massive challenge and potential for change:

Nobody knows whether this will turn out to be the best or the worst thing the human race has done for itself, because the outcome of this empowerment will depend in large part on how we react to it and what we choose to do with it. The human mind is not going to be replaced by a machine, at least not in the foreseeable future, but there is little doubt that the worldwide availability of fantasy amplifiers, intellectual toolkits, and interactive electronic communities will change the way people think, learn, and communicate.⁴

Just as humans are social, so our software must be as well.

⁴ Howard Rheingold's books are wonderful: *Tools for Thought* (<http://www.rheingold.com/texts/tft/>) and *Virtual Communities* (<http://www.rheingold.com/vc/book/>). Though they were written in 1985 and 1993, respectively, they were *at least* a decade ahead of their time. Probably two.

Social Software is a Forced Move

The person shopping at Amazon in the opening of this chapter was relying on social connections to help her make a shopping decision.

She did this in two ways:

First, she asked a friend to recommend a digital camera. That friend, knowing her and her lifestyle, would recommend a camera based on his knowledge of her. Maybe the friend recommended a camera he had experience with. Or, perhaps a different model based on some difference he recognized between them.

Second, the person relied on an informal social network of people at Amazon who wrote reviews. She didn't know these people, yet she relied on them anyway, trusting them to deliver quality information. The trust in this case is present not because they are friends, as was true for the original recommendation, but because they represent the shared experience of shopping for a camera.

This study was merely the first time this phenomenon became clear to me. Since then, I have noted it in nearly all aspects of life. Voting, shopping, eating, reading, computing, driving... in these and all activities we ask others for help in making decisions. Relying on social networks is how the vast majority of decisions are made!

A Forced Move

This reliance on our social network is increasingly a *forced move*. Living in the *Information Age*, for all its benefits and wonders, is like drinking from a fire-hose. We have more information than we know what to do with, more than we could ever digest, and probably more than we can even imagine.

And a previous age, the *Industrial Age*, still has a strong effect as well. The ease of manufacturing at a large scale has caused a situation where we simply have far too many things to choose from. So now we not only have too much information, we have too many products as well. Often we don't have two or three options to choose from: we have *dozens*. And then there is a seemingly infinite amount of information about those products! There is simply not enough time to consider each option thoroughly.

To fight this deluge of information, we're turning more and more to trusted sources, whether they be in our own household or in other

social circles. Instead of trying to sort, filter, and weed through endless sources of information, we're focusing our attention on those we already trust, or those we have reason to believe might be trusted. We don't have much choice.

The Paradox of Choice

Barry Schwartz notes an interesting side effect of this problem: the Paradox of Choice.⁵ He has found that when faced with such an overload we not only fail to make the right choice in many situations, but we often actually get paralyzed and make no choice at all! I remember a friend of mine was shopping for a digital camera several years ago, and decided to utilize several online price trackers to help him find the best model at the best price. He became paralyzed by the options. The paradox was realized: he ended up not getting a camera! He had to rationalize this by citing another reason (a change in financial situation) because on the surface, like any paradox, not choosing due to too much information seems irrational. It's not. It's human.

Ads, Ads, and more Ads

Another continuing effect of the Industrial Age is advertising, which is necessitated as the distance between the person with the message (often a business owner) and the person receiving the message (often a customer) grows. If you have a relationship with the person you're doing business with, your conversation with them (and their ability to help you) is all the advertising they need. But in an age where there is no personal relationship, no face-to-face contact, business owners need to get their message to customers in some other way, and that way is advertising.

Advertisers are always working harder to get our attention. It is said the average person sees anywhere from 500 to 3000 ads each day⁶ and an average twenty-year-old has watched 30,000 hours of television.⁷ It's hard to go anywhere and not see a plethora of advertisements: a few hours casual use of the web and TV per day and you'll easily see hundreds of advertisements.

⁵ Barry Schwartz, *The Paradox of Choice*. Harper Perennial, 2005.

⁶ There is considerable debate about how many ads people see per day, with the key issue being how many we notice vs. how many come into our peripheral vision. See more: <http://answers.google.com/answers/threadview?id=56750>

⁷ http://www.firstmonday.org/issues/issue2_4/goldhaber/index.html

Bias, Bias, and more Bias

The problem with advertisements isn't just that they're distracting, it's that they're also *biased*: they don't represent a truthful view of the world. They're all about sell, sell, sell. When we see an advertisement, we're seeing an idealistic vision of the world that simply doesn't exist.

As the shopper on Amazon said in reference to the camera manufacturer: "I already know what they're going to say." This bias is simply unacceptable. To retain our sanity in a world of too many biased messages, we're being forced to rely on our social circles to give us sorely needed unbiased perspective. We'll go out of our way for an authentic conversation with someone we can trust. We don't want to know how excited someone is to tell us about *their* great new thing, we want to hear what *people like us* have to say. Just like the Amazon shopper.

The Attention Economy

Combine the increased number of items to choose from, the blitz of advertising, and the explosive growth of the web, and it's easy to see why we are swimming in information. Humans have never had to deal with such a situation.

In 1971, seeing the writing on the wall (and everywhere else), the insightful Herbert Simon described the inevitable outcome of this information onslaught:

In an information-rich world, the wealth of information means a dearth of something else: a scarcity of whatever it is that information consumes. What information consumes is rather obvious: it consumes the attention of its recipients. Hence a wealth of information creates a poverty of attention and a need to allocate that attention efficiently among the overabundance of information sources that might consume it⁸

Simon points to the real need here: we need to allocate our attention efficiently. In other words, we need to pay attention to what matters, and try to ignore what doesn't.

The *Attention Economy*, as it has come to be called, is all about the exchange of attention in a world where it is increasingly scarce. Much of what we do on the web is about this exchange of attention. To circle back to the reviews at Amazon, it is definitely about more than money: it's about attention.

⁸ http://en.wikipedia.org/wiki/Attention_economy

At its very core, social software is about connecting people virtually who already have relationships in the physical world. That's why MySpace and Facebook are so popular. What do most people do on those sites when they sign up? They immediately connect with friends they already have!⁹ Or, to put it another way, they maintain their current attention streams. These applications are helping people manage their attention in an economy where it is increasingly hard to do so.

When we join social network sites and focus our attention mostly on the people we know there or give our attention to people like us on Amazon, we're filtering information and being parsimonious with our most precious asset. We're effectively saying "No" to the vast majority of information out there, and we're being forced to do this by the sheer amount of information we face.

Social Software is Accelerating

Social software has always been successful. Email, which dates from the early 1960s and is arguably the most successful software ever, was actually used to help build the Internet.¹⁰ Email is social, as it allows you to send messages to one or more people at a time. In the late 1970s, Ward Christensen invented the first public bulletin board system (BBS), which allowed people to post messages that others could read and respond to. One BBS, the WELL, gained tremendous popularity in the late 1980s and early 1990s as a well-known online community. Much of the early social psychology research done on online properties was focused on the WELL. Usenet, a system similar to BBSs, also found tremendous popularity in the 1980s as people posted articles and news to categories (called newsgroups). All of these social technologies predate the World Wide Web, which was invented by Sir Tim Berners-Lee in 1989.¹¹

The web is incomparable. Now, nearly two decades after its invention, the world has completely and permanently changed. It's hard to imagine what life must have been like before we had web sites and applications.

Starting with the social software precursors mentioned above, the web has evolved toward more mature social software. What follows is a very abridged history of the web from a social software point of

⁹ For more insight into the reasons why people use MySpace, read Danah Boyd's: Identity Production in a Networked Culture: Why Youth Heart MySpace <http://www.danah.org/papers/AAAS2006.html>

¹⁰ <http://en.wikipedia.org/wiki/Email>

¹¹ Super cool link: Tim Berners-Lee announcing the World Wide Web on Usenet:<http://groups.google.com/group/alt.hypertext/msg/395f282a67a1916c>

view. This is important because our audiences, except the youngest ones, have lived through and experienced this history and it shapes their expectations.

86

A One-Way Conversation (Read Only)

In 1995, back when Amazon was just a fledgling start-up, the web was quite a different place than it is now. It had just turned five years old. By one estimate it contained 18,000 web sites, total.¹² (Now there are hundreds of millions.) Most of those 18,000 web sites shared a common property: they were read-only. In other words, all you could do was read them. It was a one-way conversation. The information flowed from the person/organization who ran the site to the person viewing it. Sure, you could click on a link and be shown another page, but that was the extent of the interaction. Click, read, click, read. If you were lucky, the site might have listed a phone number that you could call.

That's not to say that people didn't use it socially. One person would write something on their web page, and a while later another would respond on their own web page. This made the conversation difficult, but possible. It's kind of like only being able to talk at your own house. When you want to say something, you and your friend go to your house. To get your friend's reply, you go to theirs.

A Two-Way Conversation (Read/Write)

Amazon and other pioneers then made a big leap forward: they figured out how to attach a database to the web site so they could store information in addition to simply displaying it. This capability, combined with cookies to save state information, as well as forms for inputting information, turned web sites into web applications. They were no longer read-only. They were read/write. Thus *two-way conversation* emerged on the web, a conversation between the person using the site and the person/organization who ran it.

12 <http://www.cnn.com/2006/TECH/internet/11/01/100millionwebsites/>

A Many-Way Conversation (Social)

Next, as web applications became more sophisticated, designers tried new feature sets. As people got comfortable interacting with them, and as bandwidth increased and access became more pervasive, designers started to enable *many-to-many* conversations. Feature sets evolved based on which features survived in the new environment. Instead of just talking to the people who published a site, you could talk to all the other people who visited it as well.

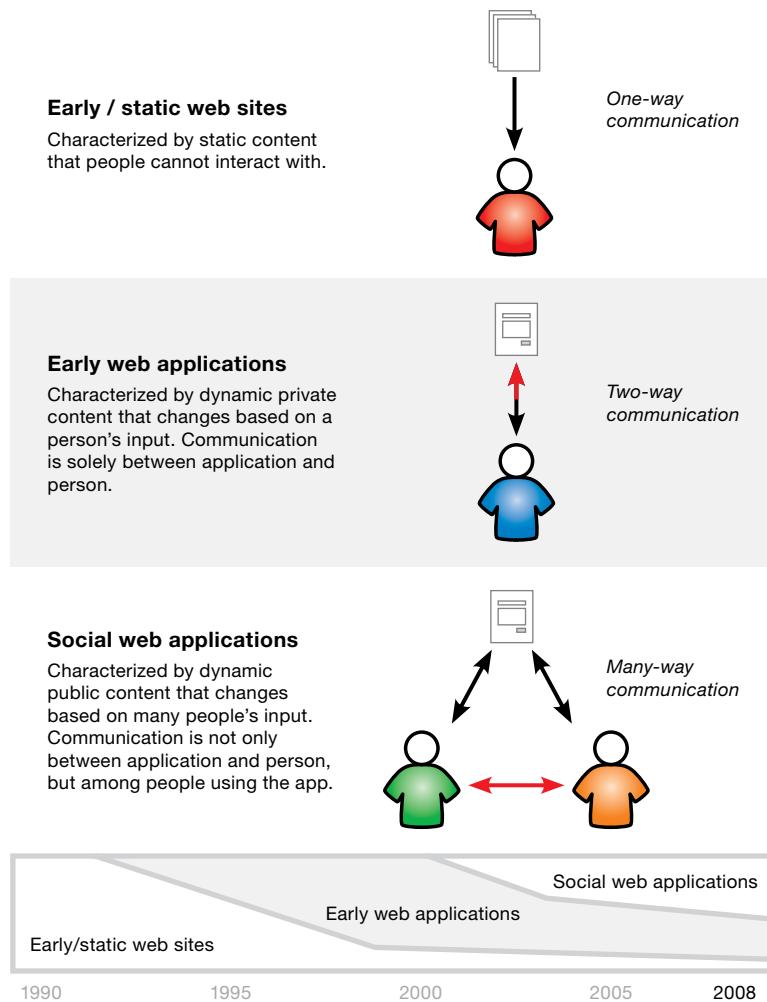


Figure 1.2 The evolution of communication from one-way to many-way on the web.

As the power and reach of the web became evident in the last part of the 1990s, designers started to refashion bulletin board systems for the web, taking advantage of the knowledge gained from those earlier attempts. One casualty of this porting was that the original BBSs largely faded away.

These many-to-many conversations were a small step technologically but a huge step socially. When you go from talking to one party (the site owner) to talking to many parties (other visitors) you enable, for the first time, *group interaction*. Group interaction is what separates a web application from a social web application.

Another recent step that has brought this change into clearer focus is *ego-centric software*. The rise of social network sites like Friendster, MySpace, and Facebook has put the person at the center of the software. While there has always been talk about *community* on the web, web software makes a much deeper set of social interactions available to us. You can friend people. You can follow them. You can even send people a kiss.

The biggest web properties are social

Social web applications are now everywhere. Consider the following list of names you know and love, all of which are in the top 30 most-trafficked web properties in the U.S.:¹³

- ▶ YouTube grew faster than any web app in history as **millions of people uploaded homemade videos**
- ▶ Wikipedia is a **collaborative encyclopedia** written by tens of thousands of contributors around the world
- ▶ MySpace is by far **the most visited social network property**, with 65 million people a month visiting in December 2007¹⁴
- ▶ eBay is an amazing ecosystem where **perfect strangers exchange billions of dollars** a year in auctions without meeting face-to-face
- ▶ The photo sharing site Flickr allows millions of people to **share photos** with friends and loved ones
- ▶ Craigslist provides a simple interface where people can interact easily and do things, such as post **classifieds**, that they used to do in newspapers

¹³ According to Alexa, a useful tool for finding trends (but like all traffic measurement sites, any specific numbers from the site should be taken with a grain of salt).

¹⁴ <http://siteanalytics.compete.com/myspace.com?metric=uv>

- ▶ Facebook started on the Harvard campus by emulating an actual book handed out to freshmen (The Facebook) and grew into a behemoth of **social networking**
- ▶ IMDb aggregates the **movie ratings** of thousands of people to provide a helpful answer to the question, should I see this movie?
- ▶ Thousands of people on Digg, a social news site, **submit and rate stories** in an attempt to make it to the home page
- ▶ Google Search works by placing relevance on the **collective linking behavior** of the entire population on the web
- ▶ Yahoo's web-based Mail application is used by **hundreds of millions of people**

But those are just the biggest ones. Lots and lots of smaller social web applications are sprouting up as people get more comfortable with the idea of interacting socially. Here are some interesting ones:

- ▶ **Sermo.** A social network site that connects professional doctors in order to speed up information sharing and dissemination
- ▶ **PatientsLikeMe.** A social network site that provides support for people living with HIV, ALS, and others
- ▶ **Kiva.** A social network site that lets people in developed countries loan money to entrepreneurs in the developing world
- ▶ **Nike+.** An app for runners who can upload their personal exercise information and share with others
- ▶ **LibraryThing.** An app that allows you to upload and share your personal library and book ratings with others
- ▶ **RateMyProfessors.** A hilarious site that allows students to rate professors in a public forum for all to see

The Fastest Growing Web Properties Are Social

Social web applications are the fastest growing properties on the web. It's no wonder. Good social sites have social features that enable them to be shared easily. Their entire purpose is to connect people, and when they do that efficiently, they grow very quickly as a result.

YouTube, for example, streams over 100 million videos *per day*. One of its co-founders, Jawed Karim, notes very few people dispute that YouTube is the fastest growing web site in Internet history.¹⁵



Figure 1.3 Social sites/applications/platforms are the fastest growing properties on the web.

Where Do You Spend Your Time?

Here's an amazing statistic:

In August 2007, over ten percent of the time Americans spent online was on a single social web app: MySpace.com.

With all the choices we have for where to spend our time, nearly twelve percent of all people's time is spent on a single site! In addition, a mere twenty web domains account for thirty-nine percent of our time online. Many of them are social web applications.

These numbers are startling for several reasons.

We are deeply attached. The average time per visit on MySpace is the length of a sitcom: twenty-six minutes.¹⁶ And, since many people visit MySpace, Facebook, and other social network sites at least once per day, this lengthy stay is habitual. In other words, the social web is becoming a way of life.

We follow our friends. One of the more egalitarian promises of the web is that "every web site is equal." Any given site has just as much opportunity as the next one. But these numbers show that while this may be true in principle, in practice people strongly congregate where their social circles and their friends are.

¹⁵ <http://www.youtube.com/watch?v=nssfmTo7SZg>

¹⁶ <http://blog.compete.com/2007/09/11/facebook-third-biggest-site-page-views-myspace-down/>

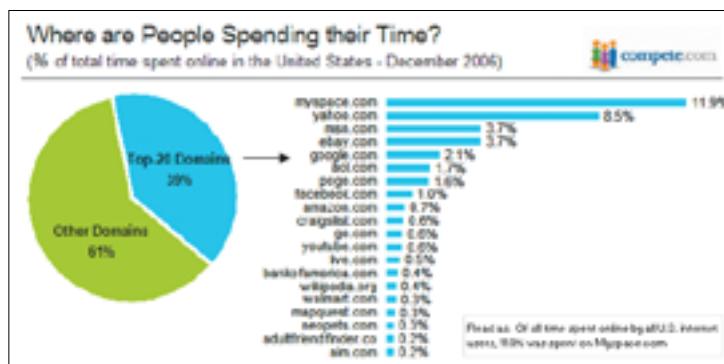


Figure 1.4 This graphic from Compete, an analytics company, shows how mad people are about MySpace. 11.9% of all online time in the U.S.? That's insane!

Blogs!

In addition to the big name sites above, there are an estimated 100 million blogs on the web. According to the blog-tracking site Technorati, in March 2007 there were approximately 70 million blogs, with 120,000 blogs being added every day!¹⁷ By the time this book is published, the number of blogs on the web will be over 100 million.

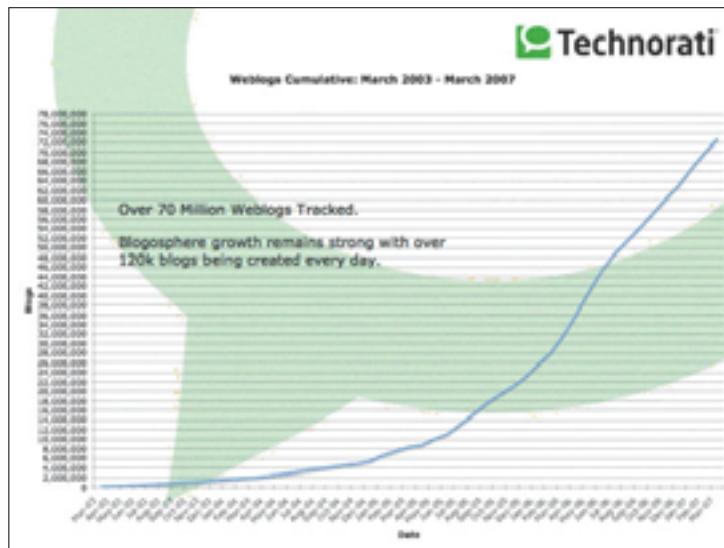


Figure 1.5 The number of blogs on the web is growing at an amazing rate, with no signs of stopping.

¹⁷ <http://technorati.com/weblog/2007/04/328.html>

Conclusion

Less Than 20% So Far

The growth of the social web is mind-boggling. Even more remarkable, however, is that this growth is *unlikely to slow down anytime soon*. According to InternetWorldStats, which aggregates statistics from sources like Nielsen/NetRatings:

Only 1.2 of the 6.5 billion people on Earth use the Internet. That's less than 20%.¹⁸

Despite the rich history of social software and the rich interactions happening already on sites like Amazon, we are still only at the beginning of the social web. As more and more people from around the world get access to the Internet and grow comfortable interacting socially online, we'll see a continued growth and maturation of social web applications. The successes of the moment (the Amazons, MySpaces, and Facebooks) will grow and change, and new applications will come to join them or take their place. That kids tend to intuitively grasp and embrace the social nature of the experience is a strong predictor of this future.

¹⁸ <http://www.internetworldstats.com/stats.htm>

	Action	Display	Feedback
Digg	Submitting a news story	Upcoming, popular, homepage	Digg, share, and bury stories
Amazon	Writing a product review	Mostful, most recent	Is this Helpful? Report this, Comment
Netflix	Rating a movie	Recommended movies	Add to queue, Rate movie, Not interested
Google	Writing a web page	Results based on relevancy	Link between web pages, Click on search results
Wikipedia	Starting an article	Article page	Edit articles over time
Del.icio.us	Saving & tagging a bookmark	Most popular, Related tags, all tags	Copy bookmarks
Flickr	Uploading and tagging a picture	Interestingness, popularity, clusters	Tagging, setting Favorites
YouTube	Uploading a video	YouTube interface, embedded in blogs	Favorite it, Report it, Embed it

“Remember: when people tell you something’s wrong or doesn’t work for them, they are almost always **right**. When they tell you exactly what they think is wrong and how to fix it, they are almost always **wrong**.”

—Neil Gaiman

part 2—prototyping



- a. Test early, test often
- b. Design together
- c. Make it look discardable



Wireframes

Early in the design process, you have two priorities:

- Test your assumptions
- Communicate with the stakeholders

97

The first lets you know if you're headed in the wrong direction, the second makes sure that people aren't surprised by the finished project. The underlying idea is the same in both cases: make sure you don't spend too much time on something you'll have to discard. You'll need a **quick-and-dirty prototype**. The first and most important prototype you'll make in any project is a simple paper sketch of what your app might look like: a wireframe.

Wireframes let you communicate with the stakeholders. Why describe your idea, when you can show it? They also let you test your design on potential users. You can ask them what they think certain elements mean, or how they would execute a certain task. No need to build anything, a pen and paper is all you need.

Of course, once you start changing your design based on this feedback, you'll want to move to the computer. As you can see on the right, even wire-frames drawn with computer software often look simplistic and hand-drawn. This is an important detail: it communicates very effectively to both your test users and your stakeholders what they are looking at. It will show them that they are not looking at the design, they are looking at an explanation of a concept. Something that will be handed off to a designer later, to be made to look slick.

The simple look has another benefit: it looks **discardable**. Even though good wireframes can take days to draw, they look like there's no investment behind them. This means that stakeholders will not be scared to suggest changes and offer criticism. Show them a slick, well-polished design, and they won't have the heart to suggest that it's all wrong.

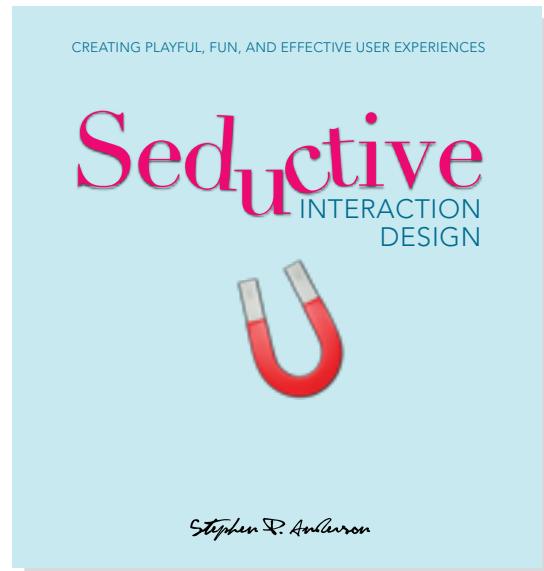
Finally, hand-drawn wireframes are a two-way language. Show an accountant a photoshopped design and the most he can say to it is yes or no. Show him a wireframe, and he can draw his own alternative, to show what he means.

Seductive interaction design

Stephen P. Anderson

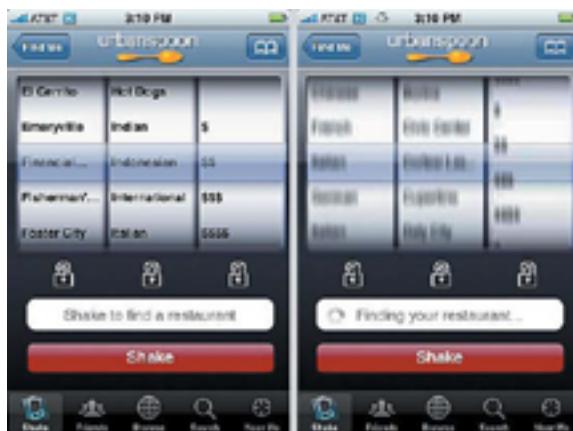
“In the context of dating, we use playful behaviors to connect with others. We tease each other with little tidbits of information about ourselves. We serve up flirtatious smiles at just the right moment. We ask silly questions or make humorous propositions. We ‘accidentally’ brush a hand against the other person’s skin. Assuming there’s some mutual interest, these signals say, ‘Hang out with me and you’ll have a good time.

So what about Web sites?”



Are You Unpredictable?

URBANSPOON CREATED an interesting twist on choosing a restaurant. Taking a cue from slot machines, it turns deciding where to eat into a playful experience. You simply shake your phone (the app was one of the first to take advantage of the iPhone's accelerometer) and three slots for cuisine, price, and location are randomly selected. Don't like your choice? Shake again. While you can "lock" any of these slots, and the service does offer restaurant reviews, it's this simple playfulness that sets Urbanspoon apart from other restaurant review apps. And of course, this is the kind of thing people like to share with one another. The app is fun and (when it was released) it showed off a cool new feature of the phone.



Surprise taken to the extreme

Chatroulette offers a similar unpredictable experience. It's a live video chat site, but not one involving a list of friends or any intentional conversations. Instead, you're randomly paired with a stranger, at least until one of you clicks Next and is paired with someone else. The completely random nature of this system attracted a lot of curious—mostly younger—users and created an opportunity for some rather odd forms of expression. Impersonators, performers, curious onlookers, and yes, exhibitionists have all flocked to the site. Advertisers used the site to promote a remake of *The Exorcist* (users were paired with a teenage girl who suddenly appeared to become a demon possessed). Musician Ben Folds used the site to do piano improv live onstage. There have even been interesting social experiments, like being paired with a handwritten message stating, "Tilt your head and I win." (Most people tend to comply, at which point a hand appears to add another tally mark to the sheet of paper.)

Unfortunately, the complete lack of identity—you simply click Play to join in—encourages voyeurism and exhibitionism. Despite efforts to clean up Chatroulette, it has become associated with men flashing their

privates. Still, there's something more than novelty going on here. In a world where most connections are based on a shared interest or history, there's something refreshing about a service based on serendipity.

Shervin Pishevar, an advisor to Chatroulette, notes, "Most of us have connected with the vast majority of our old friends, colleagues, and acquaintances by now," thanks to social networks like Facebook. "The venues to meet people *randomly* are increasingly limited." In the wake of Chatroulette, dozens of new services with business models based on serendipity have begun to emerge, including a dating site, a site for designers to get feedback on their work, and numerous Chatroulette clones that are trying to avoid the trolls by requiring authentication methods, such as Facebook Connect.

Our brains are aroused by the unexpected

While stability and a sense of control are no doubt critical user interface principles, there's something exciting about the unexpected. Not knowing what to expect heightens our anxiety, and our curiosity.

Our brains are aroused by new and unexpected discoveries within our normal routines.

From a neuroscience perspective, being surprised releases a cascade of dopamine, a reward chemical. We actually get a brief high from this momentary surprise. In the case of Chatroulette, this surprise (and the anxiety preceding each new turn) is heightened by the social aspect of the experience. This is the antithesis of predictable and boring.

Note, in both of these experiences, control is never taken away from the users. I can pass to the next person in Chatroulette, or in the case of Urbanspoon, control the outcome by locking one or more slots. These random experiences never seize control of the experience—they enhance it.

In each of these cases, *surprise* is central to the experience. However, surprise could also be some small change. Consider a confirmation message that is different every time: "Got it!" or "Your data is safe with me!" or "Home run!" Or an image that changes on a frequently visited page. These little changes make the interaction seem more human. Do you want to work with an automaton or a feeling human? Surprise can be a very minor change that adds flavor and variety to an otherwise routine experience.

Surprise can also be external to the experience you've created. In advertising, there's the idea of zigging when everyone else zags (and vice versa). If everyone else is doing something one way, you'll get noticed if you go against the grain. How you can get people's attention by deviating from expected patterns set by other sites or experiences external to your site? Certainly design patterns and tested usability principles should be followed, but there's always room to do or present things differently. *Are there any small surprises in the experience you've designed?*

MIXING SURPRISE WITH REWARDS

Surprises may also come in the form of variable rewards. Slot machines are a sinister example

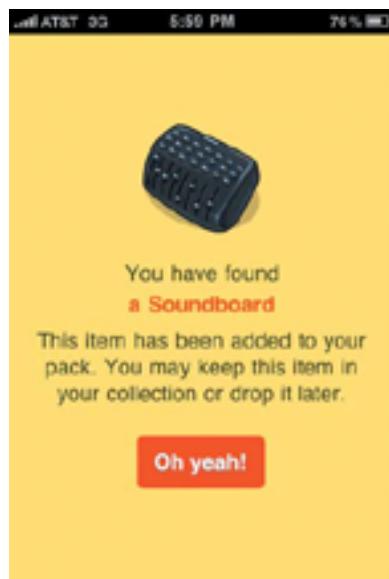


of *variable rewards*. Just when you're ready to give up, you win a little bit. This win, however small, suggests that there is more to be had—if you just keep playing. The rewards may seem random, but in fact they're calculated to keep you feeding the machine. Too many rewards and you come to expect wins, and will be likely to quit when they don't come. Too few rewards and you'll give up. This technique is commonly used by dog trainers and parents alike to reinforce a desired behavior.

Gowalla is a mobile check-in service that uses variable rewards. Part of the game of playing Gowalla is collecting and trading virtual stickers. How do you get stickers? By checking into places. Imagine you're grabbing a peppermint mocha from the café around the corner. You pull out your phone, check in, and see, "Congratulations! You've found a virtual soundboard." You've been given something for this activity. Chances are good that you'll check in again at the next location you visit. But here's the catch: you're more likely to repeat this behavior if the reward of finding a sticker occurs at variable intervals.

If you were rewarded every time you performed the activity, you'd come to expect it. By randomizing the reward schedule, you become addicted (for better or worse).

Some variable rewards are calculated. Others are naturally occurring. Consider Twitter. If I'm following someone for a particular purpose, such as staying up on the latest tech news, these useful tweets are often mixed in with other kinds of expression, say, what the person is having for lunch. All it takes is one or two really useful links shared by some folks for me to be engaged again, looking for even more. This is part of the addictiveness of this service (and why it can be so disruptive to normal routines). If I don't follow the stream, I may miss out on something useful! The variable reward of something really useful or personally relevant makes it difficult to turn away from the never-ending stream of tweets.



Gowalla's random reward notification.

DELIGHTERS

Another kind of surprise comes in the form of *delighters*. Like variable rewards, these occur at unpredictable times. They're unexpected. However, their intent is not to reinforce a particular behavior, but to simply bring joy—delight—to the user.

The term *delighters* comes from the hospitality industry and is used to refer to little things added to an experience that create delight and joy. Think of chocolates on your pillow. A plush towel. Free movies. Maybe a hot cup of tea when you're checking in.

I was on a trip to New York scouting locations for a workshop. As I walked up the stairs to exit the basement of the Ace Hotel, something caught my eye. A message had been written on one of the steps:



I smiled. *Everything is going to be alright*. I've never seen anyone use a staircase in this way. And the message made me grin. Everything is going to be all right.

This is a perfect example of a *delighter*: it was *unexpected*—who thought of using that space? It was *unnecessary*—it wasn't critical to the act of getting up and down the stairs. It was also *pleasant*.

While different from a gift in that nothing was given to me, this very minor, creative addition to the hotel staircase offered the same effect as a gift: making people feel good. There is no functional justification for adding the text to the staircase, but it does engage people emotionally.

It was an unnecessary, unexpected, but altogether delightful surprise.

A good date is full of delightful moments—some planned, some not—that make the overall experience memorable and pleasant. What does my wife remember from our first date? Among other fond memories, it rained and we got our shoes stuck in the mud!

Think of an unexpected pleasure. A free dessert after your meal. A hidden level in your favorite video game. The Google logo changing to celebrate a holiday or a person.

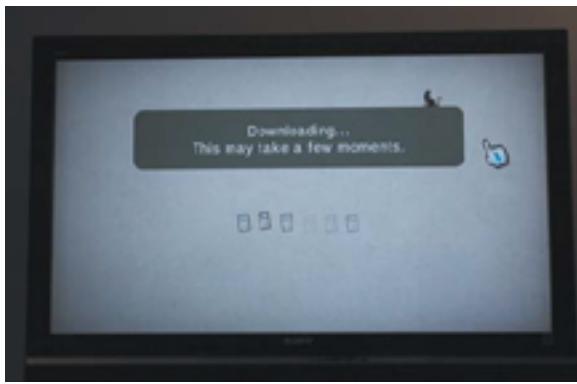
Opening the MOO.com sticker packaging in a way you're not supposed to reveals a hidden message:



On rare occasions, a simple Google search will reveal a hidden surprise:

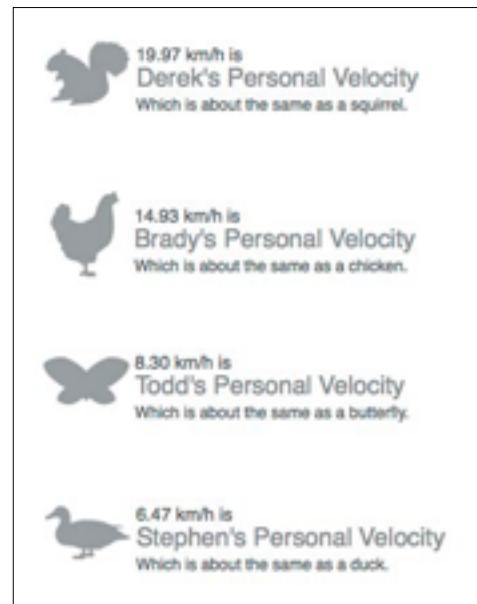


If you own a Nintendo Wii, you may have discovered the Wii Help Cat. It's a cat that wanders on the screen after some period of inactivity. If you move your cursor quickly to grab the cat, he runs away. But if you move carefully, you can sneak up on the cat and grab him (your cursor becomes a hand). Once you've grabbed the cat, you unlock a secret tip about how to use the Wii dashboard.



This is certainly not an efficient or easy way to present help text. But it does create engagement by reframing these tips as challenges to grab (literally). It also works because these tips are not essential to use the Wii, but are instead undocumented, secret features.

The notion of delighters first came to my attention in a post written by Matt Jones, co-founder of the travel site Dopplr. The site had recently rolled out a new feature: Personal Velocity, your total distance traveled divided out over a twelve-month period. By itself, the number isn't all that interesting. Who cares that my personal velocity is 6.47 km/hr? What's interesting is what they added to this metric: an avatar from the animal kingdom whose average speed is about the same as my personal velocity. (It turns out that 6.47 km/hr is about the same speed as a duck.) This small addition turned a boring metric into a playful and fun design element, and one that was well received by many Dopplr users.



This leads me into one final kind of surprise: *gifting*.

MY PERSONAL ANNUAL TRAVEL REPORT

In late 2008, I received an interesting e-mail from the travel site Dopplr:

"We'll be sending you your personal annual report in January. To make sure your data is accurate in Dopplr, be sure to add any past trips in 2008 that you might have missed."

Dopplr is a service where you can track your travel itineraries and share them with other people. It was built by a group of frequent

travelers to increase the likelihood of chance meet-ups during travel.

The e-mail went on to show a sample of what to expect. I logged into my account, added the two or three trips I had forgotten to log, and then waited patiently for my personal annual report. Two weeks later, my poster shown below, arrived as promised. It was a gorgeous, personalized poster, delivered as a PDF file. And it was, quite simply, a delightful experience, one that exceeded my expectations of the service.



My 2008 personal annual travel report from Dopplr.

Brandon Schauer, consultant for Adaptive Path, speaks frequently about “the long wow” moment, when companies plan a string of thoughtful and delightful experiences that go well beyond the initial sign-up or purchase. He had this to say about Dopplr:

This month Dopplr delightfully surprised me, supplying me with something I didn't know I needed. The result: I'm now a more loyal Dopplr user. It's a great example of a long wow moment.

What's remarkable is how much it delighted me and caused me to change my relationship with Dopplr. It delighted me because it was perhaps the best-designed statement I've ever received. I pine for the day that a bank or a phone company delivers a statement to me that provides insight about my behavior and makes me want to hang it on my refrigerator. But it also made me change my behavior:

- I immediately added my other trips that were missing from 2008 into Dopplr.*
- I searched around to see if my most desired Dopplr feature existed yet. It does!*
- And now I'm paying more attention to my update e-mails from Dopplr and spending more time with it. In all, I've reinvested in Dopplr all because they delivered something I wasn't expecting.*

So what's going on here? While we could certainly discuss any number of things (delighters,



self-expression, humor effect, feedback loops, set completion), I'd like to focus on the idea of gifting, also known as *reciprocity*.

Reciprocity states that we are all bound—even driven—to repay debts of all kinds. When someone does something for you, you feel obligated to repay. Social scientists say it's an almost automatic reaction.

So what about online interactions? Is there some gift you can give away to your customers?

Unfortunately, when the idea of a gift is suggested, we go straight for gift cards, expensive prizes, or cheap trinkets branded with the company logo. (Hint: a good gift isn't about you.) Giving something away sends us straight to “it'll cost us money.” Fortunately, as interaction designers, most of us deal in a digital currency of ones and zeros that has no hard physical costs: information and virtual gifts.

We're fortunate to work in a time and profession where information of all kinds is valued. But first, let's talk about what makes a good gift.

WHAT MAKES A GOOD GIFT?

Remember the Saturday Night Live skit "Pumping Up with Hans and Franz" and the characters' catch phrase: "We're gonna pump you up!" Well, forget bulging biceps and Austrian accents. A good gift is one that pumps up the recipient.

P.

PERSONAL

Seriously, how many more pens with a stamped company logo does the world need? Unless you're Harley Davidson, Apple, or Whole Foods, you might think about something a bit less self-centered. Writer Dr. Robert Cialdini recalls checking into the Mandarin Oriental Hotel in Hong Kong. When he went to grab some stationery from the desk, he found his name embossed on it! Not the hotel's branding—his own name. He now recommends that hotel more than any other he has ever visited.

The Dopplr travel journal was a report about me. Not the cities I visited. Not about Dopplr's users. It was about me. My travels. My friends. My carbon footprint (for better or worse).

U.

UNEXPECTED

This one is easy enough. The more unexpected a gift is, the more pleasant it will be. Companies fall into the trap of copying one another's unique ideas. And guess what? Pretty soon that unique idea isn't so unique. I'm sure the first time a company gave away some stationery it was a nice thing. Now, it's cliché.

The idea of a personal travel report had never occurred to me. I might have expected a few stats presented online, like Web site analytics are presented, but not a nicely designed, printable personal travel report.

M.

MEANINGFUL (USEFUL, NOT GENERIC)

In the late 1990s, Red Sky Interactive had a knack for creating meaningful promotions. How? They promoted brands by giving customers something they could actually use—something meaningful. For Sutter Home, America's leading producer of premium varietal wine, they created the Mood Maker, an app combining animated, user-customizable blends of wine country sights and sounds.

Some of the best promotional items I've held onto were useful to me in some way. The "It's Miller Time" campaign (for Miller beer) was extended through a Miller branded instant-messaging client. Colorado-based EffectiveUI recently gave away a dot grid sketchpad and a stencil set for sketching wireframes. How much better might the swag given away at a conference be if companies started with the premise that their logo would not be making an appearance?

My Dopplr travel report was something I could print out and put on a wall. It was an artifact of my travels and held a mirror up to such things as the number of days I was on the road and my carbon footprint (as measured in Hummers).

P.

PLEASANTLY PACKAGED

A friend of mine who designs packaging for such brands as Maui Jim, Tag Heuer, and Zales jewelers talks frequently about the "point of gifting," how the presentation of something is critical to shaping the perception that something is a nice gift to be cherished.

Essentially, Dopplr gave away information. But the information was packaged as an attractively designed poster. It was fun to look at and share with others.

Other than personal informatics like the Dopplr report, what are some other kinds of information you can give away? Information could also be exclusive research, or an unpublished or personally relevant article. Perhaps a podcast or a music download. Some authors provide notes or slides to go along with their books. For Web apps, maybe it's a free month (just for being a customer) or access to a locked or not-yet-released feature. Information could be expert knowledge and advice, experiences, stories, links, PDF files, photographs, humorous videos, recipes, tips, free online seminars, white papers, code, plug-ins, wallpapers, screen-savers, audio files, transcripts, or online tools. There's no shortage of things we can reproduce and share—publicly or privately—online.

If you've succeeded in creating your own online community, or if you operate inside an existing online community, virtual gifts such as stickers, avatars, and badges are also highly esteemed within an active social group. A powerful game mechanic behind the explosive growth of the game FarmVille was the idea of social gifting, where you need to give away seeds and items to other players to advance in the game.

Also, look beyond your own domain. How can you work with other services to provide a unique gift? Partner with a service like Gowalla to create a unique badge. Gift someone in the form of a public compliment or post.

I prefer to stay away from monetary gifts for a reason other than cost. In his 2008 book *Predictably Irrational*, Dan Ariely distinguishes between "social exchanges" and "monetary exchanges." People might gladly do something as a favor, but feel insulted when they



To promote their books, speaking, and training, brothers Dan and Chip Heath give away one-page crib notes, podcasts, and other goodies—in exchange for your e-mail address.

are offered compensation. You wouldn't get up from the table after a wonderful meal at someone's house and ask your host how much you owe them for dinner—that would be insulting. Similarly, many lawyers would rather do a small amount of volunteer work than offer their services at a reduced rate.

When negotiating new consulting opportunities, I make it a habit to share articles and links that the prospective client will find relevant to their business. This shows that I'm listening to their needs and that I am interested in the topic. While I have no expectations associated with passing on this information—it is given freely—I do hope this small gesture is reciprocated by a continued dialogue, hopefully leading to work. But there are no strings attached.

Remember, reciprocity is about gifts being given with no expectation of getting something in return. Some sites push this principle a bit further and suggest ways to return the favor. While suggesting actions is certainly okay, especially if it's something small such as liking

Give Back to the Scribd Community

Hey there—that document you’re reading was uploaded by someone just like you. Give back to the Scribd community and upload something!



Class Notes



Thesis & Dissertations



Drawings



Poetry



Short Stories



Spreadsheets & Presentations



Family Recipes



Résumés

Scribd, a social reading and publishing site, suggests you “give back” to the community—transaction or gift?

a page or sharing it with other people, this can come off as pushy if handled poorly. Think about the dating analogy: “I paid for dinner, now it’s your turn to put out.” You don’t want your Web app to be like that creep.

Also, a bit of real-world relationship advice: Gifts given freely, with no expectations, tend to pay off more than *quid pro quo* transactions masquerading as gifts. The whole “I’ll do this for you, if you’ll do this in exchange” idea cheapens otherwise nice gestures. If you’re in love with someone, you want to do things for them. You want to make them happy. And here’s the irony: when you give with no expectation of return, the favor is often returned in some meaningful way. But this isn’t an exchange of services. These are acts of caring.

In the case of Dopplr, they didn’t ask for anything in return for sending my personal annual report, but I told people about Dopplr. I mentioned them in my presentations. I invited my friends. Even as more of my friends began using

Dopplr’s primary competitor, TripIt, I remained loyal for several years.

A word of caution: today’s gift may be tomorrow’s commodity. If other companies or similar services are giving away the same thing, it’s not a gift anymore—it’s an expectation. For example, if everyone offers a free sample chapter from their book, that becomes the new expectation. It is no longer a gift.

Case in point: a classic gifting study involves the inclusion of personalized mailing address labels. When a nonprofit site included personalized mailing labels in their solicitation letter, donations nearly doubled, from 18% to 35%. Here’s the problem: how many solicitations have you received that include personalized mailing labels? At one point, I would save and use these. Now, they go straight to the recycling bin. What was once a unique and meaningful gift is no more.

Gifts can also include the joy of discovery, something that you’ll see in the next chapter.

Are You Mysterious?

112

IN THE PREVIOUS CHAPTER, we looked at how people delight in the puzzle-solving aspect of pattern recognition. Now, let's go a bit deeper and explore what drives this pattern seeking behavior: *curiosity*.

Great storytellers know how to turn an ordinary event—say, a trip to the grocer—into a suspenseful one by withholding information. In new relationships, flirtation often involves some element of playful teasing, whether through conversation or more sensual revelations. And newsrooms have made a science out of crafting irresistible headlines: “Your PC might be infected!” or “Are you prepared for the tax law changes?”

We are captivated by unanswered questions.



CURIOS MARKETING

In recent years, Hot Wheels has begun including a “mystery car” in their store shipments. While the other cars are encased in clear plastic, the mystery car is shielded by opaque black plastic. You have no idea what kind of car is in there.

With two or three dozen Hot Wheels to choose from, guess which one the kids go for? In my experience, the one that gets attention (and allowances) is the mystery car—the one that is unknown.



Crazy? Perhaps it is. But this exact same bit of psychology also works on grown-ups.

Here's a rather interesting promotion from California Pizza Kitchen. At the end of my dinner, I was given the bill and a CPK “Don't Open It” Thank You Card.



It's a coupon with an interesting twist: you bring this card with you the next time you come to CPK. You've already won something, from a free appetizer up to \$50 dollars (or more). But you won't know what you've won until your *next* visit. The instructions are pretty clear: whatever you do, do not open the card or your prize is null and void! A manager has to open the card for you when you return. You are guaranteed to get *something* worthwhile—and this is a critical part of arousing curiosity. Coupons are too explicit: "Here is your 20% off." Scratch-offs and lottery tickets are most likely to reveal that you've won nothing. With the CPK coupon, the fine print teases you with a list of the possible

prizes. Now I'm curious: *which prize have I won?* This is a mystery that needs solving.

So are there ways that we—as interaction designers—can leverage curiosity in our designs?

113

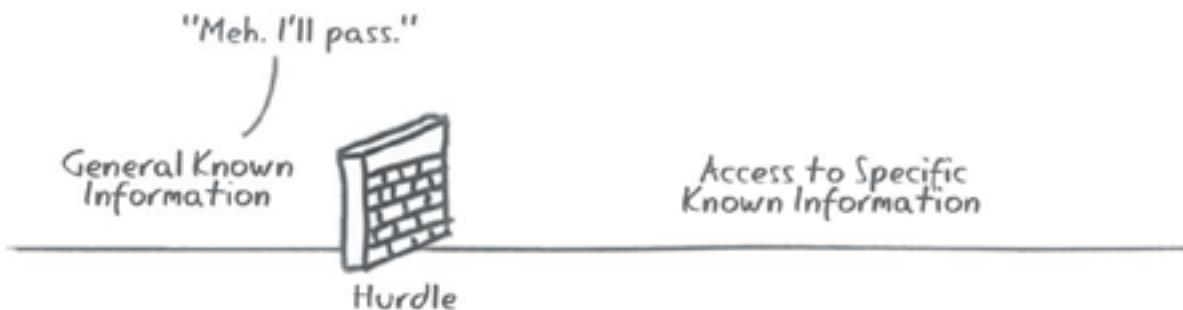
VENTURING INTO THE UNKNOWN

I've been thinking about two kinds of information: "known" and "unknown."

As user experience professionals, we excel at making things known. If it's unknown, it's unclear and likely to be confusing. A puzzling button label? Make it clear. A confusing process? Make it more familiar. For good reasons, we value things like user control, clarity, and consistency. We remove uncertainty in interfaces.

But once we've removed all the usability potholes from a particular path, how can we reintroduce the simple thrill of driving? How can interactions be made more effective—and fun—by introducing a bit of controlled uncertainty?

Let's go back to our Hot Wheels and CPK examples. Did you notice these things?



- Some tiny bit of information makes us aware of something that is unknown. Black plastic packaging hides a toy inside, or we are presented with a mysterious card.
- Context provides some relevance. These are kids shopping for a toy. I'm eating at a restaurant that I presumably like.
- Enough clues are given to help us make a judgment about the personal value of that unknown information. Kids can infer that the mystery car will be similar to other Hot Wheels. The fine print on the back of the CPK card explains the range of possible prizes. Value can come in many forms: the winning lottery ticket, the satisfaction of solving a puzzle, being entertained by a story.

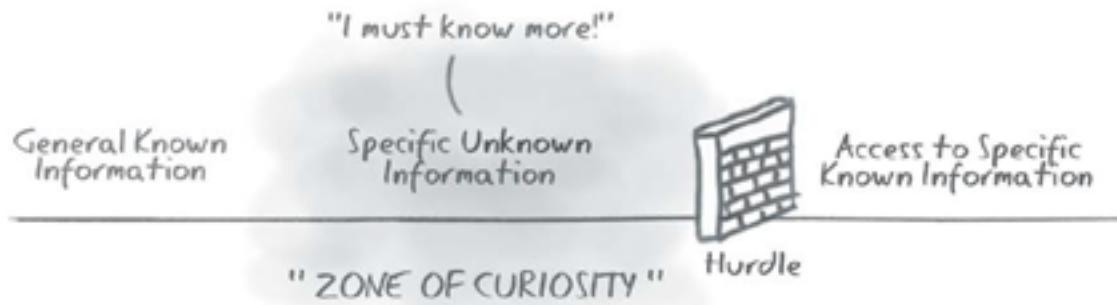
Information can be presented in a manner that is straightforward or curious. If we opt for the latter, we are guaranteed not only attention, but probably higher engagement as well—curiosity demands that we know more! What was known information (another toy car or a simple coupon) that might have been

ignored has been converted into something unknown, something mysterious, something that demands resolution.

THE INFORMATION GAP THEORY

When we become aware that information is missing—when something changes from being known (or so we thought) to an unknown state—we become curious. This is the explanation of curiosity posed by behavioral economist George Loewenstein in his information gap theory. Loewenstein says, “Curiosity happens when we feel a gap in our knowledge.”

The feeling we get from these information gaps is best described as deprivation, which is critical to understanding why we are motivated by curiosity. To “eliminate the feeling of deprivation,” we seek out the missing information. This is ironic, of course, considering that we routinely seek out puzzles, mystery novels, and other curious situations that create this sense of deprivation. However, it’s important to note that many researchers once viewed curiosity as something aversive; a decision-theoretic view



suggests that we should only want to know something if it helps us make more informed decisions. Why would we be attracted to something that offers no extrinsic benefit? Many other debates have surrounded curiosity: Is curiosity internally or externally stimulated? Is curiosity a primary drive, like hunger or fear? Is curiosity a state or trait? And this one: “If people like positive levels of curiosity, why do they attempt to resolve the curiosity?”

In his 1994 paper “The Psychology of Curiosity,” Loewenstein surveys the body of curiosity research, much of which was done in the early 1960s and 1970s. In doing so, he provides a backdrop by which to understand his own research and how it resolves many of the debates surrounding curiosity. Simply stated: I’m curious because there’s a gap between “what I know and what I want to know.” Two notable implications come from this perspective:

- The *intensity* of curiosity correlates to the likelihood of certain information to resolve the information gap. Loewenstein’s own tests confirmed that subjects were more curious when given parts of a greater whole—the need to complete enough of a picture puzzle to determine what it was (a picture of an animal) resulted in more interaction than a scenario where each block was a discrete picture.
- Curiosity correlates with our own understanding of a particular domain. The more we know about some topic, the more likely we are to focus on our own information gaps. If I know eight of ten items, I’m more curious about the remaining two than if I only know two of ten things.

BUSINESS APPLICATION?

Given that curiosity reflects a desire to close information gaps, how can we apply this to interaction design?

LinkedIn

Let’s illustrate this gap in knowledge with a look at the professional networking site LinkedIn. One of the site’s business goals is to sell paid accounts. Like most businesses, LinkedIn has a generic description of the benefits you receive with a paid account. Think of this as generally known information. While this information could certainly be compelling, there’s a population for whom the cost may not be worth the perceived value.

Of course, those customers with paid memberships have access to specific known information.

This is how most businesses run: “*Cross the [registration/paid account/personal information] threshold and you can have all this!*” Unfortunately, this generic description of benefits is often not enough for many people.

LinkedIn gives you a *personalized* glimpse of what could be known, essentially teasing you with relevant information such as, “Someone at [company name] viewed your profile.” The site moves you into an unknown state by sharing bits of knowledge that can only be fully known as a paid member. Nothing has been given away for free—I still don’t know who looked at my profile, but I’m aware of some partial knowledge that might be worthwhile to know in full (see the top screen on the following page). As one friend said, “If someone from Apple has been looking at my profile, you can bet I want to know who!”

If this partial information proves relevant and valuable, you'll want to know more, right? In essence, they've created a "zone of curiosity" between two previously known states.

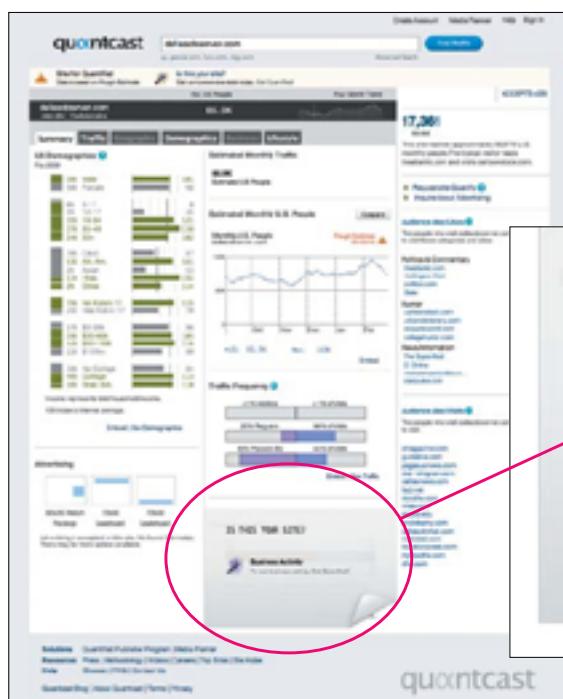


Your profile has been viewed by 11 people in the last 2 weeks, including:

- Someone at **Team One Advertising**
- Principal** in the design industry
- Owner** in the internet industry
- Someone at **Pcms**
- Someone at **Nhs**

To see 6 more people, [upgrade your account](#)

LinkedIn teases you with partial information about who has viewed your profile.



The Quantcast dashboard shows various site metrics. A large, semi-transparent white sticker with a black border is overlaid on the "Business Activity" section. The sticker contains the text "IS THIS YOUR SITE?" at the top, followed by "Business Activity" and "To see business activity, Get Quantified!" Below this, there is more text and a "Get Quantified" button. A pink arrow points from the "Business Activity" text on the sticker to the "Business Activity" section on the dashboard.

On the Quantcast site, the text barely showing beneath the sticker is intriguing.

Quantcast

Quantcast does something similar, only they've created a much larger zone of curiosity. With nothing required on your part, you can get a ton of free and quite useful site metrics: traffic stats, demographic information, lists of similar sites, and so on. The value to a site owner is obvious. But there's a bit of information withheld: to get business activity data, you must "Get Quantified."

What's nice about this version of Quantcast's call to action is what our brains see: something being hidden from us (see screens below). You can almost see through the sticker covering some data! Obviously, this is a static image—there is no live data there beneath a sticker. But we think in images and this visual affordance registers as, "Here's a sticker. We need to know what's underneath it. We can't allow this knowledge to remain unknown!"

Netflix

Netflix leverages these same ideas when returning a movie rental. For Netflix, the data from your movie rental preferences is gold. Rating a movie not only improves your recommendations, but collectively improves the entire recommendation system. Consequently, the site is built around the idea of rating movies. Why then would the site ask you, “Rate your recent return to reveal two movies you’ll love?”



There is immediacy to this request—we see the empty slots where two movies will be revealed. Sure, I can rate movies and get recommendations all over the site, but there’s something more immediate and novel about how this is presented.

As with Quantcast, I see the thing I want to take action on. I’m presented with two unknowns. For the “cost” of rating this movie, I can reveal two more (hopefully interesting) movies. I can make the unknown known. Even using a word like “reveal” suggests that there are already two movies waiting for my response.

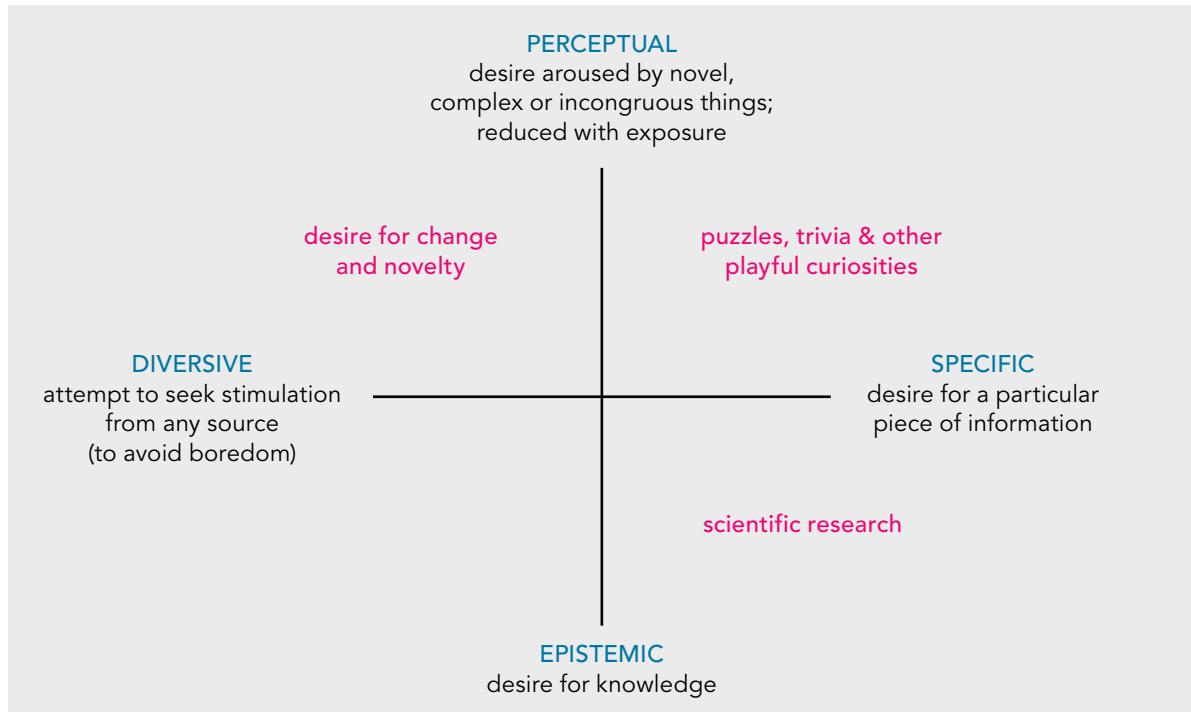
SPECIFIC MOTIVATION

It’s human to be curious. And it’s human to pursue a mystery until it’s resolved. If teased with a bit of interesting information, we want to know more. But to be clear, what we’re talking about here is a very specific kind of curiosity.

In the early 1950s, D. E. Berlyne identified two dimensions of curiosity: one extending between perceptual and epistemic curiosity, the other spanning specific and diverse curiosity. I’ve plotted these at the top of the following page with a few of examples based on my understanding of his research.

Although Berlyne’s concept of curiosity has been challenged, it remains the backdrop against which many subsequent curiosity studies have defined their research. I’ve found this model useful for thinking about different kinds of curiosity and for clarifying which type is most easily applied to interaction design.

In the context of this book, I’m referring to perceptual-specific curiosity, one in which we confront people with very specific gaps in their knowledge in a novel manner or context. While you can certainly create gaps in knowledge in a variety of ways, the examples in this chapter are more concerned with a variety of curiosity akin to teasing. After all, this book is about seductive interaction design.



This matrix presents the different dimensions of curiosity originally proposed by D.E. Berlyne in the early 1950s.

NOW WHAT?

If you want to make someone curious, make them aware of something they don't know. Find information you can use to tease people. Chances are, you're either withholding all the specific information or giving it all away. To get attention and engage the senses, look for ways to turn these direct messages into a quest. A few tips:

- Make your tease interesting, or at least proportionate in appeal to the cost.

- Strive to make the information personally relevant to the user.
- Offer the promise of something worthwhile—what will it cost?
- Establish trust through previous experiences and context clues.
- Use visuals to suggest or create the immediate perception of mystery.
- Don't try to lure users with something that is given away freely elsewhere.

ANXIETY AND DELIGHT

Interview with Giles Colborne

When creating delightful experiences, we focus on making things pleasant, removing points of frustration, maybe adding playful elements.

You have a different perspective.

Yes. When I ask people about experiences they've found delightful, their stories involve unpleasant situations or where playfulness would be inappropriate, like online banking. When I ask them about playful experiences, they seem underwhelmed and far from delighted.

Themes of anxiety and disappointment run through people's stories of delightful experiences. The passenger ringing his airline to complain is anxious. He is delighted when the anxiety is unexpectedly removed. I think that *anxiety*, present or vividly remembered, is an important part of experiencing delight. The contrast makes the delight intense and memorable.

Are you saying anxiety needs to be present to create a delightful experience? What about simple forms of delight like play or humor?

Anxiety was a common thread in the stories. You could take that two ways: either anxiety needs to be present, or the heightened emotion led people to remember the events more vividly. Personally, I think you need contrasts when you're designing emotion—as you need them to design a page layout.

However, the stories did not always include an immediate source of anxiety. One person—a young father—told me about his delightful moment coming across a device for feeding babies without mess. The anxiety was remembered.

Out of the thirty people I spoke to about

delightful experiences, none of them mentioned play or humor. I was asking them specifically about companies or services, but still no one mentioned 'paint balling' or 'comedy club.' They seemed to separate delight from play. They were much more keen to tell me about their mobile phones or vacuum cleaners.

But play and humor also need contrasts between the positive and negative feelings. Play provides safe rules for exploring situations like conflict (think: team sports) or for exceeding normal social bounds (think: using dolls to play 'grown-ups'). Humor also is a safe way of exploring negative situations—often relying on shock or on setting the audience off balance in some way.

How might you apply this? Can we design for anxiety? Can the experiences we design be solutions—heroes, if you will—to anxieties? Conversely, would you advocate creating a service experience that manufactures anxiety?

Absolutely. As designers, we should seek out anxiety. We should ask users: tell me about the times you felt anxious. Fix the problems that users remember and fear the most. If you can, you'll delight your users.

To take it to the next level, you could try to create anxiety, in order to release it. But you'd need to be very careful about doing that. It's appropriate in playful situations—people understand play as a safe way of experimenting with anxiety, often it's expected. Enhancing anxiety in a credit card transaction is not such a good idea.*

*Visit www.sixdbook.com to read full interview.

100 things every designer needs to know about people

Susan M. Weinschenk

Most failures in interaction design are the result of a designer misunderstanding his users. You've carefully explained in your interface how the users should behave, and they don't even read the instructions. Your application behaves consistently, but the user keeps expecting different things. You've told them how it works in simple terms, but nobody understands until you show them.

People are strange, irrational creatures. They decide quickly, and on a whim. An application that feels logical doesn't normally look it from the programmer's perspective. An application that feels logical does what the user expects. And that can be a difficult thing to predict.



THE MORE UNCERTAIN PEOPLE ARE, THE MORE THEY DEFEND THEIR IDEAS

122

I'm a staunch Apple convert. I wasn't always an Apple fan; I used to be a Windows/PC person. Realize that I go all the way back to when PCs first came out. I used to have a marvelous "portable" PC that ran on a CPM operating system and had two (count 'em, two) 360 KB (yes, I said KB) floppy disk drives (in other words, *no* hard drive). I was a PC person, *not* an Apple person. Apples were for teachers and then later, for artsy people. That was not me.

Fast forward to today and I will be talking on my iPhone, while charging my iPod for my afternoon exercise, while transferring a movie to my iPad from my MacBook Pro, which I might decide to watch on my television via Apple TV. What the heck happened here? (I describe the story of how I changed my loyalty from PCs to Apple in my book *Neuro Web Design: What Makes Them Click?* It's a matter of starting with small changes and commitments and then growing to more loyalty).

So you might be able to guess what happened when I went to dinner with a colleague who was showing me his Android phone. He loves his new Android phone and wanted to show me all the ways it was as good as, or better than, my iPhone. I was totally uninterested in hearing about it. I didn't even want to look at it. Basically, I didn't want to allow into my brain any information that would conflict with my opinion that anything besides an iPhone was even a possibility. I was showing classical symptoms of *cognitive dissonance denial*.

ALTER OUR BELIEFS OR DENY THE INFORMATION?

In 1956 Leon Festinger wrote a book called *When Prophecy Fails*. In it he describes the idea of cognitive dissonance. Cognitive dissonance is the uncomfortable feeling you get when you have two ideas that conflict with each other. You don't like the feeling, so you'll try to get rid of the dissonance. There are two main ways you can do that: change your belief, or deny one of the ideas.

When forced, people will change their beliefs

In the original research on cognitive dissonance, people were forced to defend an opinion that they did not believe in. The result was that people tended to change their beliefs to fit the new idea.

In new research by Vincent Van Veen (2009) had people “argue” that the fMRI scan experience was pleasant (it’s not). When “forced” to make statements that the experience was pleasant, certain parts of the brain lit up (the dorsal anterior cingulate cortex and the anterior insular cortex). The more these regions were activated, the more the participant would claim that he really did think the fMRI was pleasant.

123

When not forced, people dig in

There’s another reaction that sometimes occurs. What if you are not forced to state that you believe something you don’t; what if instead you are presented with information that opposes your beliefs, but you are not forced to espouse a new belief. In these situations the tendency is to deny the new information instead of changing your belief to fit.

If uncertain, people will argue harder

David Gal and Derek Rucker (2010) recently conducted research where they used framing techniques to make people feel uncertain. (For example, they told one group to remember a time when they were full of certainty, and the other group to remember a time when they were full of doubt.) Then they asked the participants whether they were meat-eaters, vegetarians, vegans, or otherwise, how important this was to them, and how confident they were in their opinions. People who were asked to remember a time of uncertainty were less confident of their eating choices. However, when asked to write up their beliefs to persuade someone else to eat the way they did, they would write more and stronger arguments than those who were certain of their choice. Gal and Rucker performed the research with different topics (for example, preferences for a Mac versus a PC computer) and found similar results. When people were less certain, they would dig in and argue even harder.

Takeaways

- * Don’t spend a lot of time trying to change someone’s ingrained beliefs.
- * The best way to change a belief is to get someone to commit to something very small.
- * Don’t just give people evidence that their belief is not logical, or tenable, or a good choice. This may backfire and make them dig in even harder.

124

Imagine that you've never seen an iPad, but I've just handed you one and told you that you can read books on it. Before you turn on the iPad, before you use it, you have a model in your head of what reading a book on the iPad will be like. You have assumptions about what the book will look like on the screen, what things you will be able to do, and how you will do them—things like turning a page, or using a bookmark. You have a *mental model* of reading a book on the iPad, even if you've never done it before.

What that mental model in your head looks and acts like depends on many things. If you've used an iPad before, your mental model of reading a book on an iPad will be different than that of someone who has never used one, or who doesn't even know what an iPad is. If you've been using a Kindle for the past year, then your mental model will be different from someone who has never read a book electronically. And once you get the iPad and read a couple books on it, whichever mental model you had in your head before will start to change and adjust to reflect your experience.

I've been talking about mental models (and their counterparts, conceptual models, discussed below) since the 1980s. I've been designing interfaces for software, Web sites, medical devices, and various products for many years. I always enjoy the challenge of matching what's going on in people's brains with the constraints and opportunities presented by technology. Interface environments come and go (for example, the green screen of character-based systems, or the blue screen of early graphical user interfaces), but people change more slowly. Some of the age-old user interface design concepts are still extremely relevant and important. Mental models and conceptual models are some of the most useful design concepts that I believe have passed the test of time.



The origin of the term *mental models*

The first person to talk about mental models was Kenneth Craik in his 1943 book, *The Nature of Explanation*. Shortly thereafter, Craik died in a bicycle accident and the concept went dormant for many years. It reappeared in the 1980s, when two books were published with the title *Mental Models*, one by Philip Johnson-Laird and the other by Dedre Gentner.

The best history I've found about mental models as they relate to software and usability is a 1999 article by Mary Jo Davidson, Laura Dove, and Julie Weltz called "Mental Models and Usability." (<http://www.lauradove.info/reports/mental%20models.htm>)

WHAT EXACTLY IS A MENTAL MODEL?

There are many definitions for mental models that have been around for at least 25 years. One of my favorites is from Susan Carey's 1986 journal article "Cognitive Science and Science Education," which states:

"A mental model represents a person's thought process for how something works (i.e., a person's understanding of the surrounding world). Mental models are based on incomplete facts, past experiences, and even intuitive perceptions. They help shape actions and behavior, influence what people pay attention to in complicated situations, and define how people approach and solve problems."

WHAT IS A MENTAL MODEL IN DESIGN?

In the field of design, a mental model refers to the representation of something—the real world, a device, software, and so on—that a person has in mind. People create mental models very quickly, often before they even use the software or device. Their mental models come from their prior experience with similar software or devices, assumptions they have, things they've heard others say, and also from their direct experience with the product or device. Mental models are subject to change. People refer to mental models to predict what the system, software, or product is going to do, or what they should do with it.



Terminology can be confusing

The way I'm using the term mental model is, I believe, the most common definition, but it does not fit with at least one of the new definitions I've seen lately. In her book *Mental Models*, Indi Young uses the term in a different way. She diagrams the behavior of a particular audience doing a series of tasks, including their goals and motivations. Under the diagrams she describes what the "system" will do to match the task. She calls this entire structure a "mental model." The results seem useful, but I would not call them a mental model. It's a different use of the term.

Takeaways

- * People always have a mental model.
- * People get their mental models from past experience.
- * Not everyone has the same mental model.
- * An important reason for doing user or customer research is so you can understand the mental models of your target audience.

PEOPLE PROCESS INFORMATION BEST IN STORY FORM

126

One day, many years ago, I found myself in front of a classroom full of user interface designers who did not want to be there. Their boss had told them they had to attend the talk I was giving. I knew that many or most of them thought the class was a waste of time, and knowing that was making me nervous. I decided to be brave and forge ahead. Certainly my great content would grab their attention, right? I took a deep breath, smiled, and with a strong voice, I started the session with a big, “Hello, everyone. I’m certainly glad to be here.” More than half the class wasn’t even looking at me. They were reading their e-mails and writing out to-do lists. One guy had the morning newspaper open and was reading that. It was one of those moments where seconds seem like hours.

I thought to myself in panic, “What am I going to do?” Then I had an idea. “Let me tell you a story,” I said. At the word story everyone’s head jerked up and all eyes were on me. I knew I only had a few seconds to start a story that would hold their attention.

“It was 1988 and a team of Navy officers were staring at a computer screen. Something had just appeared on the radar in protected air space. They had orders to shoot down any unknown aircraft. Was this an unknown aircraft? Was it a military plane? Was it a commercial airliner? They had two minutes to decide what to do.” I had them! Everyone was interested and riveted. I finished the story, which nicely made my point about why it can be so important that they should care about designing usable interfaces that avoid user uncertainty, and we were off to a great start. The rest of the day flew by, everyone was interested and engaged, and I got some of my best teacher evaluations ever. Now I make sure to use that magic phrase, “Let me tell you a story,” at least once in every talk I give or class I teach.

You may have realized that what I did in the paragraph above was tell a story. Stories are very powerful. They grab and hold attention. But they do more than that. They also help people process information and they imply causation.

TRIED-AND-TRUE STORY FORMATS

Aristotle identified the basic structure of stories, and many people have expounded on his ideas since. One model is the basic three-act structure: beginning, middle, and end. This may not sound very unusual, but when Aristotle came up with it over two thousand years ago it was probably pretty radical.

In the beginning you introduce your audience to the setting, the characters, and the situation or conflict. In the story above I introduced you to the setting (I had to give a class), the characters (me and students), and the conflict (the students don’t want to be there).

My story was very short, so the middle part was short too. In the middle part of a story, there are typically obstacles and conflicts that the main character has to overcome. These are usually somewhat resolved, but not completely resolved. In my story above the main character tried her usual opening and it failed. Then she started to panic.

In the end of the story the conflict comes to a climax and then is resolved. In my story above I thought of what to do (tell a story to the class), which I did, and which succeeded.

This is just a basic outline. There are many variations and plots that can be added and woven in.

CLASSIC STORIES

There are many stories that appear over and over in literature and in movies. Here are some of the popular themes that have been identified:

- ★ The Great Journey
- ★ Love
- ★ Coming of Age
- ★ Fate
- ★ The Sacrifice
- ★ Revenge
- ★ The Epic Battle
- ★ The Trick
- ★ The Fall from Grace
- ★ Mystery

STORIES IMPLY CAUSATION

Stories may create causation when none is there. Because stories usually involve some form of chronological narrative (first this happens, next this happens), they imply causation even where none exists. Christopher Chabris and Daniel Simons give this example in their book *The Invisible Gorilla* (2010). Look at these two passages:

Joey's big brother punched him again and again. The next day his body was covered by bruises.

Joey's crazy mother became furiously angry with him. The next day his body was covered by bruises.

In the first passage you don't need to assume much. Joey got punched and he has bruises. He got the bruises from being punched. In the second passage the inference is not quite so clear. Research shows that your brain will actually take a little bit longer to ponder the second paragraph. Yet most people will conclude that Joey has bruises because of his mother, even though the passage doesn't say that. In fact, if you ask people later to remember the passage, they will believe that they read in the story that Joey's mother actually hit him, even though that is not what the paragraph says.

People are quick to assign causality. Just as the visual cortex is filling in what you see to find and detect patterns (see the “How People See” chapter), our thought processes do the same thing. You are always looking for causation. Your brain assumes you have been given all the pertinent information and that there is causation. Stories make it even easier to make this causal leap.

STORIES ARE IMPORTANT IN ALL COMMUNICATIONS

Sometimes clients say to me, “Stories are fine at some Web sites, but not the one I’m working on now. I’m designing the Web site for the company’s annual report. Stories aren’t appropriate there; it’s just financial information.” Not true. There are appropriate stories you can use any time you are trying to communicate.

Medtronic is a medical technology company. Take a look at their annual report. (The online version is the same as the print version: <http://216.139.227.101/interactive/mdc2010/>). The cover of the report is a high-quality photo of Antoinette Walters, a patient who was helped by one of Medtronic’s products. Later in the report there is a short story about Antoinette:

“Antoinette Walters, shown here and on the cover, had such a severe lumbar scoliosis that the pain incapacitated her, and the deformity was progressively getting worse. Then she underwent spinal fusion surgery using Medtronic spinal products to correct the alignment. Today, Antoinette’s spine is much straighter, her pain is virtually gone, and she is several inches taller.”

Antoinette’s is not the only story in the annual report. Sprinkled in with the financial information are high-quality photos as well as stories about people like Antoinette and employees who invented various technologies. The stories make the rest of the information in the report more interesting, and also create a link between the financial numbers and the stated mission of the company.

Takeaways

- * Stories are the natural way people process information.
- * Use a story if you want people to make a causal leap.
- * Stories aren’t just for fun. No matter how dry you think your information is, using stories will make it understandable, interesting, and memorable.

Let's say you're a marketing person and you want to email your customers about a new product offering. Take a minute to glance through some directions on how to build an e-mail campaign using the MailChimp service we discussed earlier:

1. From the Dashboard or the Campaign Tab click on the big ol' "Create Campaign" button and select the type of campaign you'd like to create (start with regular ol' campaign.).
2. On Step 1 of the Campaign Builder, select the list you'd like to send to. Once you've selected the list use the "next" option to move forward, or click "send to entire list".
3. On Step 2 of the Campaign Builder, you will have the options to name your campaign, set up a subject line, from name reply-to email and personalize your "To:" field with *|MERGETAGS|*. You will also find your options for tracking, authentication, analytics tracking and social sharing. (Use the "next" and "back" options to navigate through the steps (not your browser's back button)).
4. Select a Template for your email by clicking on "pre-designed", "autoconnect", "premium", or "start from scratch", etc (to get a basic template layout that you can fully customize) under the templates heading. Templates you've set up and saved will live under "my templates". If you're providing your own code use the "paste/import HTML" or "import from URL" options. If you want to create an editable (or non-editable) Template for your clients, choose "code custom templates".
5. Once you choose your template you'll remain on Step 3 of the Campaign Builder. The content editor is where you will edit your styles and content. Click on "show style editor" to bring up the style options.
6. With the Style Editor visible and you'll have options to edit the styles for each section. Here the "Body" tab is selected and the "title style" subheading has been clicked. This will allow you to set the line height, font size and more for this section.
7. Click anywhere inside the dotted red borders to bring up the content editor box.

8. After you click save wait for your content to refresh then click on the “next” option. Our plain text generator will automagically create the plain text version from your HTML version. Just look this version over to make sure it looks the way you like and click “next” to move to the last step of the Campaign Builder.
9. Step 5 of the Campaign Builder is a “pre-delivery checklist”. If we see anything missing on your campaign you’ll be alerted in red on this screen. Click on “edit” to be taken directly back to any area that needs attention.

You can preview the campaign once more by clicking on the “pop up preview” button.

Then we recommend sending tests to several email addresses to see how the campaign looks in your recipient’s inboxes. If everything looks good, you can schedule or send out your campaign.

Long and difficult to understand, right? Luckily this is *not* how the information is actually presented at MailChimp. The text is the same, but it is combined with screen shots to show an example of what the text is talking about. **Figure 34.1** shows a portion of what the screen really looks like, with text and picture together.

2. On Step 1 of the Campaign Builder, select the list you’d like to send to. Once you’ve selected the list use the “next” option to move forward, or click “send to entire list”.



campaign creation

3. On Step 2 of the Campaign Builder, you will have the options to name your campaign, set up a subject line, from name reply-to email and personalize your “To:” field with “[MERGETAGS]”. You will also find your options for tracking, authentication, analytics tracking and social sharing. (Use the “next” and “back” options to navigate through the steps (not your browser’s back button)).

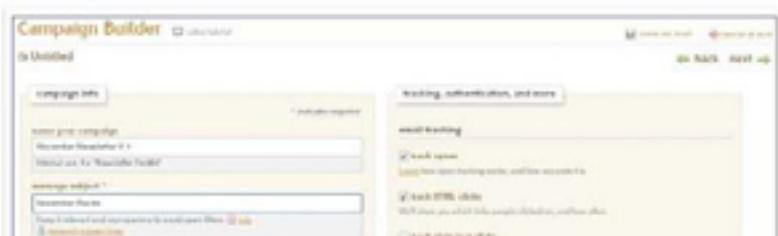


FIGURE 34.1 MailChimp (mailchimp.com) uses pictures to give examples of the steps. (From MailChimp. MailChimp is a trademark of The Rocket Science Group, LLC.)

Screen shots or pictures are not the only way to provide examples. At the MailChimp site there are also links to videos that walk you through the same steps (Figure 34.2). Video is one of the most effective ways to give examples online. Videos combine movement, sound, and vision, and don't require reading, so they are attention-getting and engaging.



FIGURE 34.2 MailChimp also uses videos to give examples. (From MailChimp. MailChimp is a trademark of The Rocket Science Group, LLC.)

Takeaways

- * People learn best by example. Don't just tell people what to do. Show them.
- * Use pictures and screen shots to show by example.
- * Better yet, use short videos as examples.

50

PEOPLE ARE MORE MOTIVATED AS THEY GET CLOSER TO A GOAL

132

You're given a frequent buyer card for your local coffee shop. Each time you buy a cup of coffee you get a stamp on your card. When the card is filled, you get a free cup of coffee. Here are two different scenarios:

- ★ **Card A:** The card has 10 boxes for the stamps, and when you get the card, all the boxes are blank.
- ★ **Card B:** The card has 12 boxes for the stamps, and when you get the card the first two boxes are already stamped.

Question: How long will it take you to get the card filled up? Will it take longer or shorter for scenario A versus scenario B? After all, you have to buy 10 cups of coffee in both scenarios in order to get the free coffee. So does it make a difference which card you use?

The answer, apparently, is yes. You'll fill up the card faster with Card B than with Card A. And the reason is called the *goal-gradient effect*.

The goal-gradient effect was first studied in 1934 by Clark Hull using rats. He found that rats that were running a maze to get food at the end would run faster as they got to the end of the maze.

The goal-gradient effect says that you will accelerate your behavior as you progress closer to your goal. The coffee reward card scenarios I describe above were part of a research study by Ran Kivetz (2006) to see if people would act like the rats did in the original 1934 study. And the answer is, yes, they do. In addition to the coffee shop study, Kivetz found that people would go to a Web site more frequently and rate more songs during each visit as they got closer to a reward goal at the site.

The Dropbox Web site (**Figure 50.1**) shows how close you are to reaching a goal that gives you extra storage space. As you get closer to the goal, you'll be more motivated to take the one or two steps left to reach it.



People focus on what's left more than what's completed

Minjung Koo and Ayelet Fishbach (2010) conducted research to see which would motivate people more to reach a goal: a) focusing on what they'd already completed, or b) focusing on what remained to accomplish. The answer was b—people were more motivated to continue when they focused on what was left to do.



FIGURE 50.1 Dropbox shows you how close you are to the goal

Takeaways

- * The shorter the distance to the goal, the more motivated people are to reach it. People are even more motivated when the end is in sight.
- * You can get this extra motivation even with the illusion of progress, as in the coffee card B example in this section. There really isn't any progress (you still have to buy 10 coffees), but it seems like there has been some progress so it has the same effect.
- * People enjoy being part of a reward program. When compared to customers who were not part of the program, Kivetz found that the customers with reward cards smiled more, chatted longer with café employees, said “thank you” more often, and left a tip more often.
- * Motivation and purchases plummet right after the goal is reached. This is called a *post-reward resetting phenomenon*. If you have a second reward level people won't initially be very motivated to reach that second reward.
- * You're most at risk of losing your customer right after a reward is reached.

57

PEOPLE ARE INHERENTLY LAZY

134

It might be exaggerating a bit to say that people are inherently lazy. But research does show us that people will do the least amount of work possible to get a task done.

IS LAZY ANOTHER WORD FOR EFFICIENT?

Over eons of evolution, humans have learned that they will survive longer and better if they conserve their energy. You want to spend enough energy to have enough resources (food, water, sex, shelter), but beyond that you are wasting your energy if you spend too much time running around getting or doing more stuff. Of course, questions about how much is enough, and whether we have enough stuff yet, and how long should the stuff last (and on and on), still vex us, but putting the philosophical questions aside, for most activities, most of the time, humans work on a principle called *satisficing*.

SATISFY PLUS SUFFICE EQUALS SATISFICE

Herbert Simon is credited with coining the term *satisfice*. He used it to describe a decision-making strategy in which the person decides to pick the option that is adequate, rather than optimal. The idea of satisficing is that the cost of making a complete analysis of all the options is not only not worth it, but may be impossible. According to Simon we often don't have the cognitive faculties to weigh all the options. So it makes more sense to make a decision based on "what will do" or what is "good enough" rather than trying to find the optimal or perfect solution. If people satisfice rather than optimize, there are implications for the design of Web sites, software, and other products.

DESIGN WEB SITES FOR SCANNING, NOT READING

In his book *Don't Make Me Think* (2005), Steve Krug applies the idea of satisficing to the behavior you can observe when someone comes to your Web site. You're hoping the visitor will read the whole page, but, as Krug says, "What they actually do most of the time (if we're lucky) is *glance* at each new page, scan some of the text, and click on the first link that catches their interest or vaguely resembles the thing they're looking for. There are usually large parts of the page that they don't even look at." Krug talks about Web pages being like billboards. You have to assume that people are taking a quick glance.

Keeping this idea in mind, look quickly at the following four screenshots of the home pages of several state government Web sites in the U.S. Imagine that you're making a trip to the state, and you're looking for tourism information. Don't study any of the pages, just glance briefly at **Figure 57.1**, **Figure 57.2**, **Figure 57.3**, and **Figure 57.4**.



FIGURE 57.1 Rhode Island state Web site



FIGURE 57.2 Mississippi state Web site

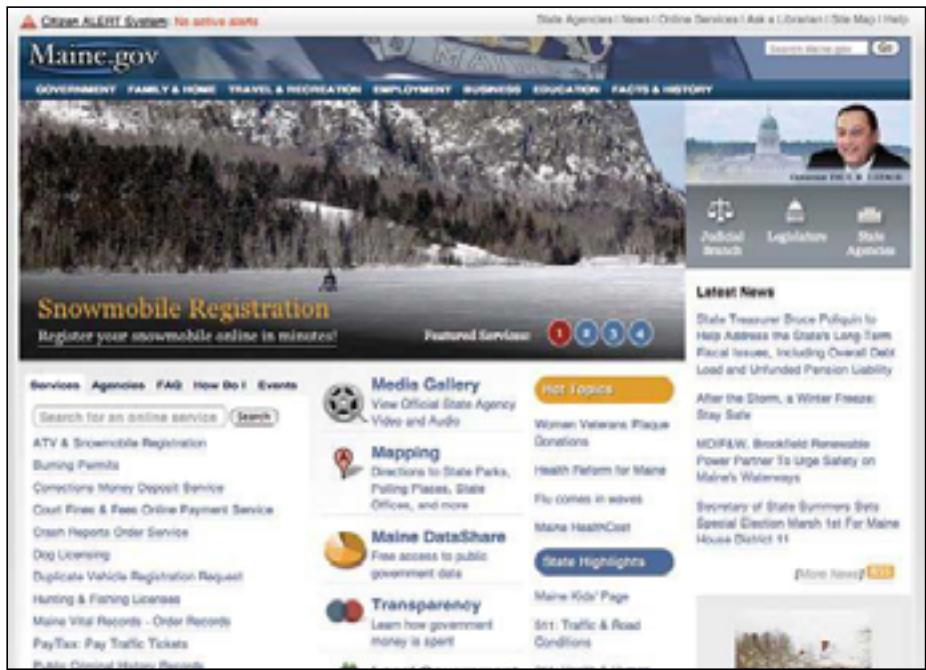


FIGURE 57.3 Maine state Web site



FIGURE 57.4 Texas state Web site

With this quick look you might get the feeling that the Maine and Texas Web sites will require less work than the others. You make a decision that a particular Web site will be easy to use based on the impression the site provides in one or two seconds of viewing. The Maine and Texas sites have more white space and a larger font size. Plus the Texas

site puts Search literally front and center. These factors make it seem like it will be easy enough or good enough to find the information you're looking for. The first impressions about satisficing can be critically important in determining whether someone stays at the Web site or not.

137

Takeaways

- * Assume that people will get things done with the least amount of work possible. That may not always be the case, but it's true more often than not.
- * People will satisfice, that is, look for the good-enough solution rather than the optimal solution.

Don't make me think

Steve Krug

“And there’s really only one way to answer that kind of question: **testing**. You have to use the collective skill, experience, creativity and common sense of the team to build some version of the thing (even a crude version), then watch ordinary people carefully as they try to figure out what it is and how to use it.

There’s no substitute for it.”



“The Farmer and the Cowman Should Be Friends”

WHY MOST WEB DESIGN TEAM ARGUMENTS
ABOUT USABILITY ARE A WASTE OF TIME, AND
HOW TO AVOID THEM

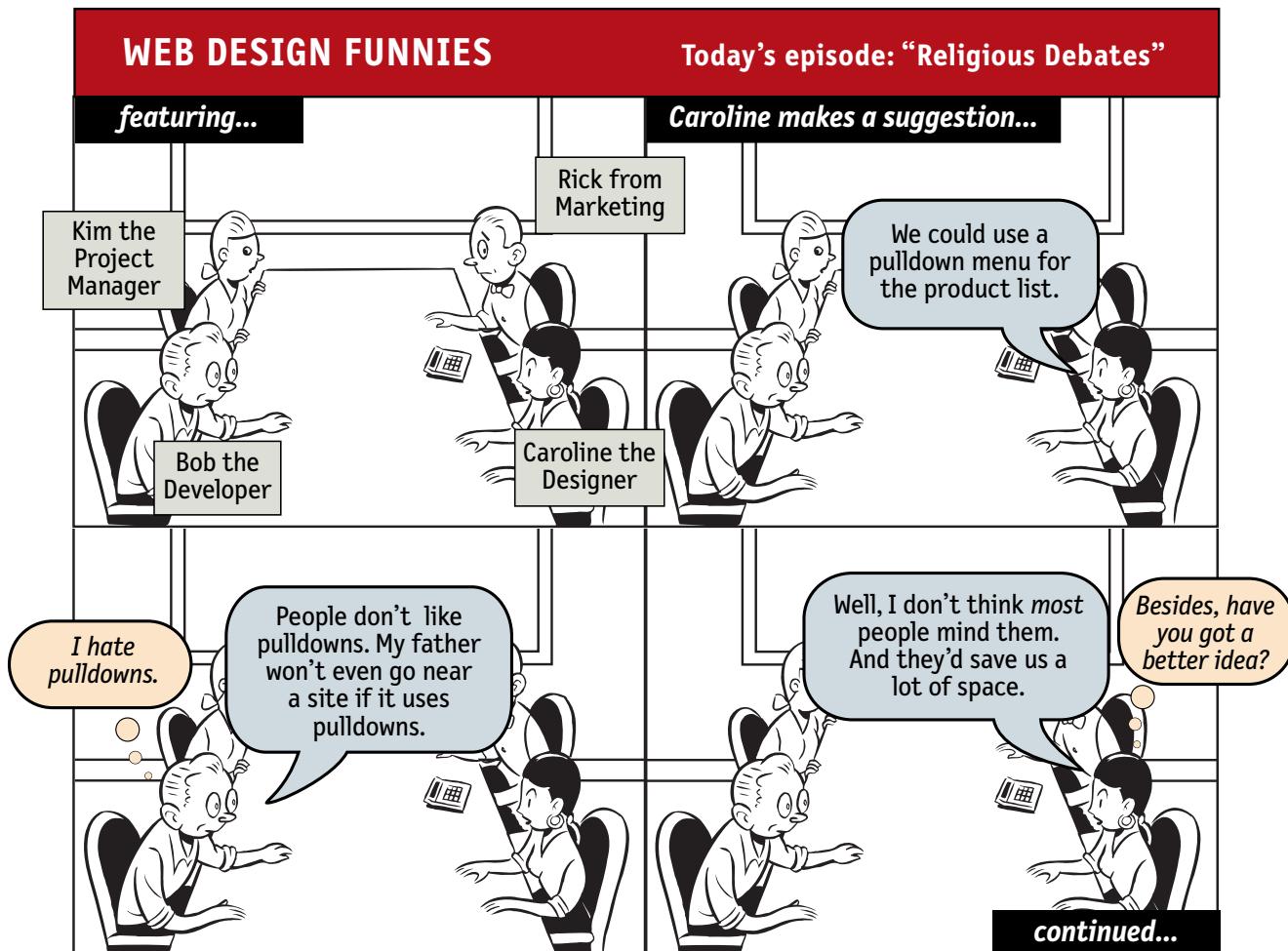
*One man likes to push a plough
The other likes to chase a cow
But that's no reason why they can't be friends*

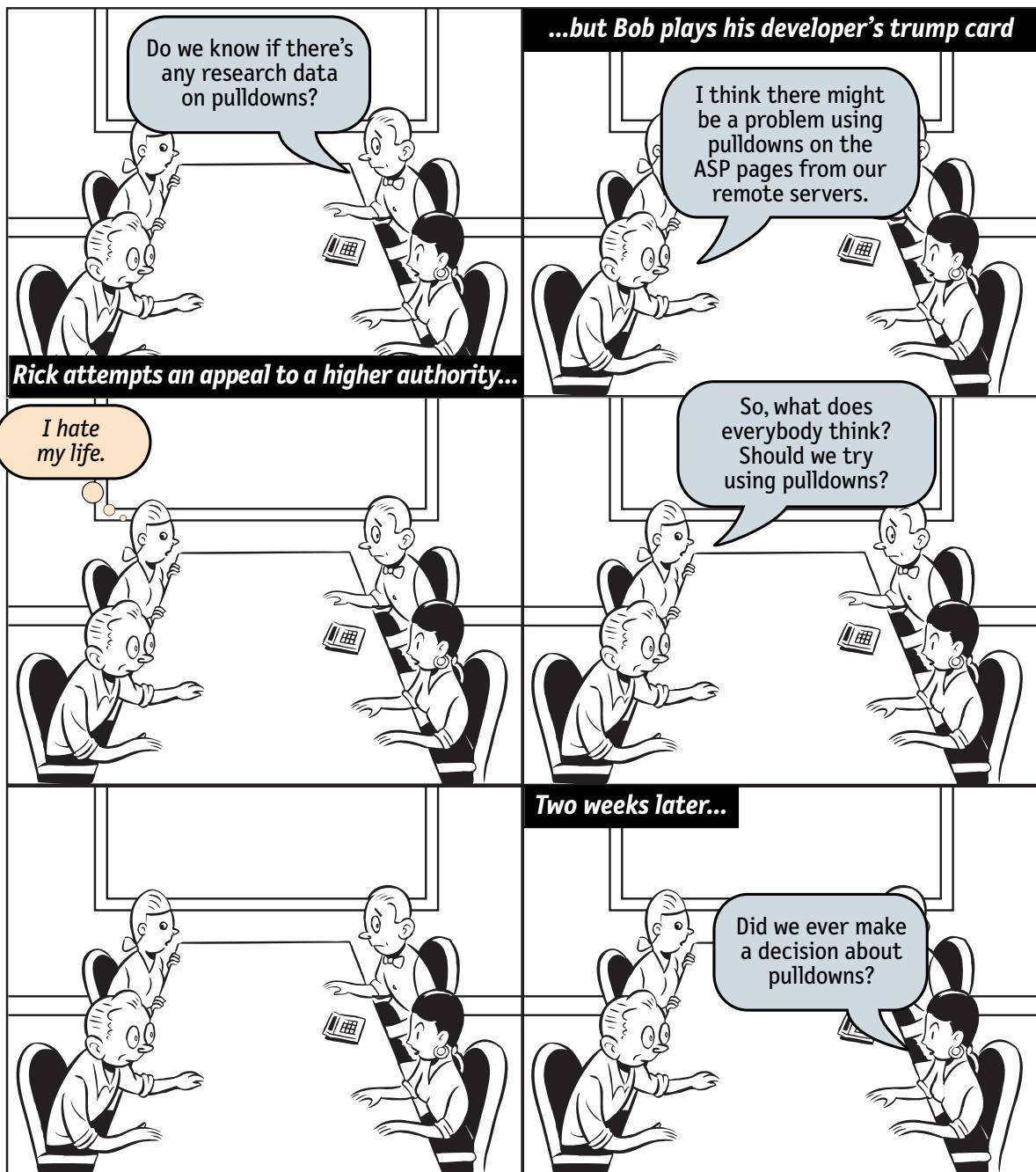
—OKLAHOMA!, OSCAR HAMMERSTEIN II

141

LEFT TO THEIR OWN DEVICES, WEB DEVELOPMENT TEAMS aren't notoriously successful at making decisions about usability questions. Most teams end up spending a lot of precious time rehashing the same issues over and over.

Consider this scene:





I usually call these endless discussions “religious debates,” because they have a lot in common with most discussions of religion and politics: They consist largely of people expressing strongly held personal beliefs about things that can’t be proven—supposedly in the interest of agreeing on the best way to do something

important (whether it's attaining eternal peace, governing effectively, or just designing Web pages). And, like most religious debates, they rarely result in anyone involved changing his or her point of view.

Besides wasting time, these arguments create tension and erode respect among team members, and can often prevent the team from making critical decisions.

Unfortunately, there are several forces at work in most Web teams that make these debates almost inevitable. In this chapter, I'll describe these forces, and explain what I think is the best antidote.

“Everybody likes ____.”

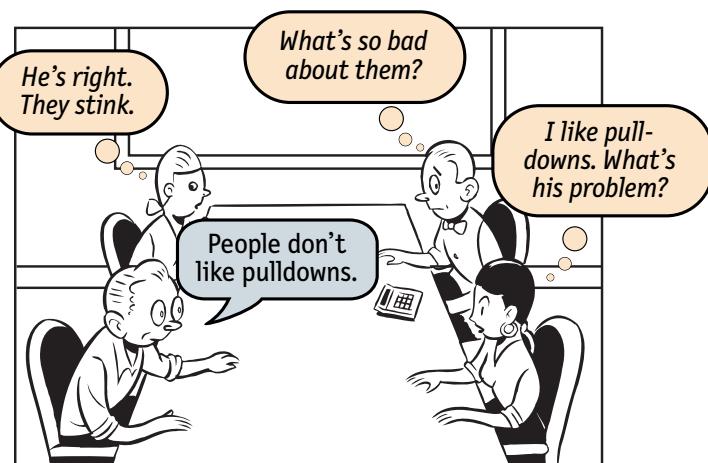
All of us who work on Web sites have one thing in common—we're also Web *users*. And like all Web users, we tend to have strong feelings about what we like and don't like about Web sites.

As individuals, we love Flash animations because they're cool; or we hate them because they take a long time to download. We love menus down the left side of each page because they're familiar and easy to use, or we hate them because they're so boring. We really enjoy using sites with ___, or we find ___ to be a royal pain.

And when we're working on a Web team, it turns out to be very hard to check those feelings at the door.

The result is usually a room full of individuals with strong personal convictions about what makes for a good Web site.

And given the strength of these convictions—and human nature—there's a natural tendency to project these likes and dislikes onto Web users in general: to think that most Web users like the same things we like. We tend to think that most Web users are like us.



It's not that we think that *everyone* is like us. We know there are *some* people out there who hate the things we love—after all, there are even some of them on our own Web team. But not *sensible* people. And there aren't many of them.

Farmers vs. cowmen

On top of this layer of personal passion, there's another layer: professional passion. Like the farmers and the cowmen in *Oklahoma!*, the players on a Web team have very different perspectives on what constitutes good Web design based on what they do for a living.¹

The ideal Web page as seen by someone whose job is...



CEO



Developer



Designer



Business development

Take designers and developers, for instance. Designers tend to think that most people like sites that are visually interesting because *they* like sites that are visually interesting. In fact, they probably became designers because they enjoy good design; they find that it makes things more interesting and easier to understand.²

Developers, on the other hand, tend to think people like sites with lots of cool features because *they* like sites with lots of cool features.

The result is that designers want to build sites that look great, and developers want to build sites with interesting, original, elegant features. I'm not sure who's the farmer and who's the cowman in this picture, but I do know that their differences in perspective often lead to conflict—and hard feelings—when it comes time to establish design priorities.

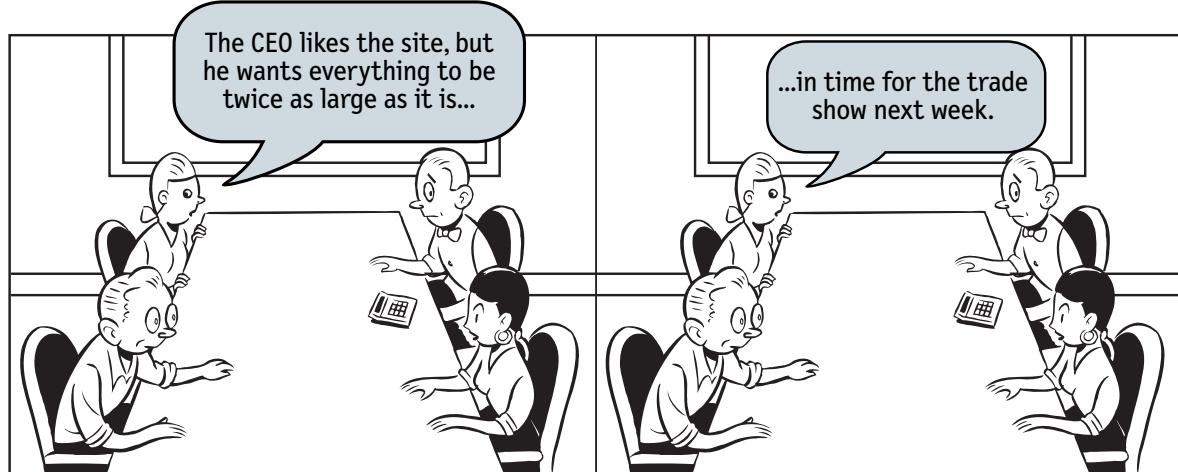
¹ In the play, the thrifty, God-fearing, family-oriented farmers are always at odds with the freewheeling, loose-living cowmen. Farmers love fences, cowmen love the open range.

² Yes, I'm dealing in stereotypes here. But I think they're useful stereotypes.

At the same time, designers and programmers find themselves siding together in another, larger clash between what Art Kleiner describes as the cultures of hype and craft.³

While the hype culture (upper management, marketing, and business development) is focused on making whatever promises are necessary to attract venture capital, users, strategic partners, and revenue-generating deals to the site, the burden of delivering on those promises lands on the shoulders of the craft culture artisans like the designers and programmers.

This Internet version of the perennial struggle between art and commerce (or perhaps farmers and cowmen vs. the railroad barons) adds another level of complexity to any discussions of usability issues—often in the form of apparently arbitrary edicts handed down from the hype side of the fence.⁴



³ See “Corporate Culture in Internet Time” in strategy+business magazine (www.strategy-business.com/press/article/10374, free registration required).

⁴ I once saw a particularly puzzling feature on the Home page of a prominent—and otherwise sensibly designed—site. When I asked about it, I was told, “Oh, that. It came to our CEO in a dream, so we had to add it.” True story.

The myth of the Average User

The belief that most Web users are like us is enough to produce gridlock in the average Web design meeting. But behind that belief lies another one, even more insidious: the belief that most Web users are like *anything*.

As soon as the clash of personal and professional opinions results in a stalemate, the conversation usually turns to finding some way (whether it's an expert opinion, research, focus groups, or user tests) to determine what *most* users like or don't like—to figure out what the Average Web User is really like. The only problem is, there is no Average User.

In fact, all of the time I've spent watching people use the Web has led me to the opposite conclusion: all Web users are unique, and all Web use is basically idiosyncratic.

The more you watch users carefully and listen to them articulate their intentions, motivations, and thought processes, the more you realize that their individual reactions to Web pages are based on so many variables that attempts to describe users in terms of one-dimensional likes and dislikes are futile and counter-productive. Good design, on the other hand, takes this complexity into account.

And the worst thing about the myth of the Average User is that it reinforces the idea that good Web design is largely a matter of figuring out what people like. It's an attractive notion: either pulldowns are good (because most people like them), or they're bad (because most people don't). You should have links to everything in the site on the Home page, or you shouldn't. Menus on the top work better than menus down the side. Frames, pages that scroll, etc. are either good or bad, black or white.

The problem is there *are* no simple “right” answers for most Web design questions (at least not for the important ones). What works is good, integrated design that fills a need—carefully thought out, well executed, and tested.

Take the use of Flash, for example.⁵ If asked, some percent of users will say they really like Flash, and an equal percent will probably say they hate it. But what

⁵ *Flash*, Macromedia's tool for creating animated and interactive user interfaces, not *flash* (lowercase), the arbitrary use of whiz-bang features to make a site more interesting.

they really hate is Flash used badly: large, complicated animations that take a long time to download and don't add any value. If you observe them carefully and ask the right questions, you'll likely find that these same people will appreciate sites that use small, hardworking, well-thought-out bits of Flash to add a pleasant bit of sizzle or useful functionality without getting in the way.

That's not to say that there aren't some things you should *never* do, and some things you should *rarely* do. There are some ways to design Web pages that are clearly wrong. It's just that they aren't the things that Web teams usually argue about.

The antidote for religious debates

The point is, it's not productive to ask questions like "Do most people like pulldown menus?" The right kind of question to ask is "Does *this* pulldown, with *these* items and *this* wording in *this* context on *this* page create a good experience for most people who are likely to use *this* site?"

And there's really only one way to answer that kind of question: testing. You have to use the collective skill, experience, creativity, and common sense of the team to build some version of the thing (even a crude version), then watch ordinary people carefully as they try to figure out what it is and how to use it.

There's no substitute for it.

Where debates about what people like waste time and drain the team's energy, testing tends to defuse arguments and break impasses by moving the discussion away from the realm of what's right or wrong and into the realm of what works or doesn't work. And by opening our eyes to just how varied users' motivations, perceptions, and responses are, testing makes it hard to keep thinking that all users are like us.

Can you tell that I think testing is a good thing?

The next chapter explains how to test your own site.

CHAPTER

9

Usability testing on 10 cents a day

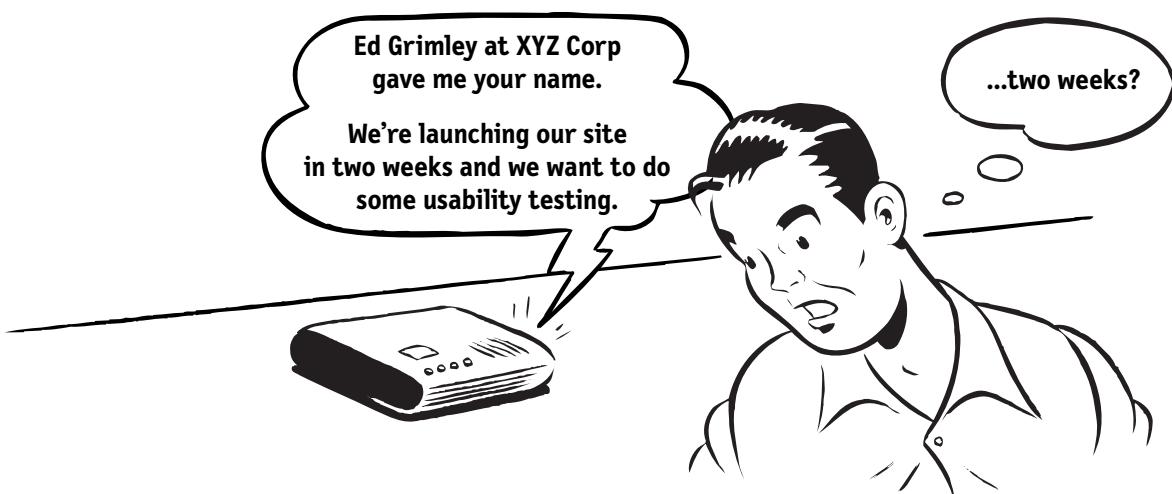
KEEPING TESTING SIMPLE—SO YOU DO ENOUGH OF IT

Why didn't we do this sooner?

—WHAT EVERYONE SAYS AT SOME POINT DURING THE FIRST USABILITY TEST OF THEIR WEB SITE

149

About once a month, I get one of these phone calls:



As soon as I hear “launching in two weeks” (or even “two months”) and “usability testing” in the same sentence, I start to get that old fireman-headed-into-the-burning-chemical-factory feeling, because I have a pretty good idea of what’s going on.

If it’s two weeks, then it’s almost certainly a request for a disaster check. The launch is fast approaching and everyone’s getting nervous, and someone finally says, “Maybe we better do some usability testing.”

If it’s two months, then odds are that what they want is to settle some ongoing internal debates—usually about something very specific like color schemes. Opinion around the office is split between two different designs; some people like the sexy one, some like the elegant one. Finally someone with enough clout to authorize the expense gets tired of the arguing and says, “All right, let’s get some testing done to settle this.”

And while usability testing will sometimes settle these arguments, the main thing it usually ends up doing is revealing that the things they were arguing about aren't all that important. People often test to decide which color drapes are best, only to learn that they forgot to put windows in the room. For instance, they might discover that it doesn't make much difference whether you go with the horizontal navigation bar or the vertical menus if nobody understands the value proposition of your site.

Sadly, this is how most usability testing gets done: too little, too late, and for all the wrong reasons.

Repeat after me: Focus groups are not usability tests.

Sometimes that initial phone call is even scarier:



When the last-minute request is for a focus group, it's usually a sign that the request originated in Marketing. When Web sites are being designed, the folks in Marketing often feel like they don't have much clout. Even though they're the ones who spend the most time trying to figure out who the site's audience is and what they want, the designers and developers are the ones with most of the hands-on control over how the site actually gets put together.

As the launch date approaches, the Marketing people may feel that their only hope of sanity prevailing is to appeal to a higher authority: research. And the kind of research they know is focus groups.

I often have to work very hard to make clients understand that what they need is usability testing, not focus groups. Here's the difference in a nutshell:

- › In a **focus group**, a small group of people (usually 5 to 8) sit around a table and react to ideas and designs that are shown to them. It's a group process, and much of its value comes from participants reacting to each other's opinions. Focus groups are good for quickly getting a sampling of users' opinions and feelings about things.
- › In a **usability test**, one user at a time is shown something (whether it's a Web site, a prototype of a site, or some sketches of individual pages) and asked to either (a) figure out what it is, or (b) try to use it to do a typical task.

Focus groups can be great for determining what your audience wants, needs, and likes—in the abstract. They're good for testing whether the idea behind the site makes sense and your value proposition is attractive. And they can be a good way to test the names you're using for features of your site, and to find out how people feel about your competitors.

But they're *not* good for learning about whether your site works and how to improve it.

The kinds of things you can learn from focus groups are the things you need to learn early on, *before* you begin designing the site. Focus groups are for **EARLY** in the process. You can even run them late in the process if you want to do a reality check and fine-tune your message, but *don't* mistake them for usability testing. They *won't* tell you whether people can actually use your site.

Several true things about testing

Here are the main things I know about testing:

- › **If you want a great site, you've got to test.** After you've worked on a site for even a few weeks, you can't see it freshly anymore. You know too much. The only way to find out if it really works is to test it.

Testing reminds you that not everyone thinks the way you do, knows what you know, uses the Web the way you do.

I used to say that the best way to think about testing was that it was like travel: a broadening experience. It reminds you how different—and the same—people are, and gives you a fresh perspective on things.

But I finally realized that testing is really more like having friends visiting from out of town. Inevitably, as you make the tourist rounds with them, you see things about your home town that you usually don't notice because you're so used to them. And at the same time, you realize that a lot of things that you take for granted aren't obvious to everybody.

- › **Testing one user is 100 percent better than testing none.** Testing always works, and even the worst test with the wrong user will show you important things you can do to improve your site. I make a point of always doing a live user test at my workshops so that people can see that it's very easy to do and it always produces an abundance of valuable insights. I ask for a volunteer and have him try to perform a task on a site belonging to one of the other attendees. These tests last less than ten minutes, but the person whose site is being tested usually scribbles several pages of notes. And they always ask if they can have the recording of the test to show to their team back home. (One person told me that after his team saw the recording, they made one change to their site which they later calculated had resulted in \$100,000 in savings.)
- › **Testing one user early in the project is better than testing 50 near the end.** Most people assume that testing needs to be a big deal. But if you make it into a big deal, you won't do it early enough or often enough to get the most out of it. A simple test early—while you still have time to use what you learn from it—is almost always more valuable than a sophisticated test later.

Part of the conventional wisdom about Web development is that it's very easy to go in and make changes. The truth is, it turns out that it's not that easy to make changes to a site once it's in use. Some percentage of users will resist almost any kind of change, and even apparently simple changes often turn out to have far-reaching effects, so anything you can keep from building wrong in the first place is gravy.

- › **The importance of recruiting representative users is overrated.** It's good to do your testing with people who are like the people who will use your site, but it's much more important to test early and often. My motto—as you'll see—is "Recruit loosely, and grade on a curve."
- › **The point of testing is not to prove or disprove something. It's to inform your judgment.** People like to think, for instance, that they can use testing to prove whether navigation system "a" is better than navigation system "b", but you can't. No one has the resources to set up the kind of controlled experiment you'd need. What testing *can* do is provide you with invaluable input which, taken together with your experience, professional judgment, and common sense, will make it easier for you to choose wisely—and with greater confidence—between "a" and "b."
- › **Testing is an iterative process.** Testing isn't something you do once. You make something, test it, fix it, and test it again.
- › **Nothing beats a live audience reaction.** One reason why the Marx Brothers' movies are so wonderful is that before they started filming they would go on tour on the vaudeville circuit and perform scenes from the movie, doing five shows a day, improvising constantly and noting which lines got the best laughs. Even after they'd settled on a line, Groucho would insist on trying slight variations to see if it could be improved.

Mrs. Teasdale (Margaret Dumont) and Rufus T. Firefly eavesdrop in *Duck Soup*.



Lost our lease, going-out-of-business-sale usability testing

Usability testing has been around for a long time, and the basic idea is pretty simple: If you want to know whether your software or your Web site or your VCR remote control is easy enough to use, watch some people while they try to use it and note where they run into trouble. Then fix it, and test it again.

In the beginning, though, usability testing was a very expensive proposition. You had to have a usability lab with an observation room behind a one-way mirror, and at least two video cameras so you could record the users' reactions *and* the thing they were using. You had to recruit a lot of people so you could get results

THE TOP FIVE PLAUSIBLE EXCUSES FOR NOT TESTING WEB SITES

	<p>It's true that most Web development schedules seem to be based on the punchline from a Dilbert cartoon. If testing is going to add to everybody's to-do list, if you have to adjust development schedules around tests and involve key people in preparing for them, then it won't get done. That's why you have to make testing as small a deal as possible. Done right, it will save time, because you won't have to (a) argue endlessly, and (b) redo things at the end.</p>
	<p>Forget \$5,000 to 15,000. If you can convince someone to bring in a camcorder from home, you'll only need to spend about \$300 for each round of tests.</p>
	<p>The least-known fact about usability testing is that it's incredibly easy to do. Yes, some people will be better at it than others, but I've never seen a usability test fail to produce useful results, no matter how poorly it was conducted.</p>
	<p>You don't need one. All you really need is a room with a desk, a computer, and two chairs where you won't be interrupted.</p>
	<p>One of the nicest things about usability testing is that the important lessons tend to be obvious to everyone who's watching. The serious problems are hard to miss.</p>

that were statistically significant. It was Science. It cost \$20,000 to \$50,000 a shot. It didn't happen very often.

But in 1989 Jakob Nielsen wrote a paper titled “Usability Engineering at a Discount”¹ and pointed out that it didn't have to be that way. You didn't need a

¹ *Proceedings of the Third International Conference on Human-Computer Interaction, Boston, MA, Sept. 1989.*

usability lab, and you could achieve the same results with a lot fewer users.

The idea of discount usability testing was a huge step forward. The only problem is that a decade later most people still perceive testing as a big deal, hiring someone to conduct a test still costs \$5,000 to \$15,000, and as a result it doesn't happen nearly often enough.

What I'm going to commend to you in this chapter is something even more drastic: Lost our lease, going-out-of-business-sale usability testing.

I'm going to try to explain how to do your own testing when you have *no* money and *no* time. Don't get me wrong: *If you can afford to hire a professional to do your testing, by all means do it!* But *don't* do it if it means you'll do less testing.

	TRADITIONAL TESTING	LOST-OUR-LEASE TESTING
NUMBER OF USERS PER TEST	Usually eight or more to justify the set-up costs	Three or four
RECRUITING EFFORT	Select carefully to match target audience	Grab some people. Almost anybody who uses the Web will do.
WHERE TO TEST	A usability lab, with an observation room and a one-way mirror	Any office or conference room
WHO DOES THE TESTING	An experienced usability professional	Any reasonably patient human being
ADVANCE PLANNING	Tests have to be scheduled weeks in advance to reserve a usability lab and allow time for recruiting	Tests can be done almost any time, with little advance scheduling
PREPARATION	Draft, discuss, and revise a test protocol	Decide what you're going to show
WHAT/WHEN DO YOU TEST?	Unless you have a huge budget, put all your eggs in one basket and test once when the site is nearly complete	Run small tests continually throughout the development process
COST	\$5,000 to \$15,000 (or more)	\$300 (a \$50 to \$100 stipend for each user) or less
WHAT HAPPENS AFTERWARDS	A 20-page written report appears a week later, then the development team meets to decide what changes to make	The development team (and interested stakeholders) debrief over lunch the same day

How many users should you test?

In most cases, I tend to think the ideal number of users for each round of testing is three, or at most four.

The first three users are very likely to encounter nearly all of the most significant problems,² and it's much more important to do more rounds of testing than to wring everything you can out of each round. Testing only three users helps ensure that you *will* do another round soon.³

Also, since you will have fixed the problems you uncovered in the first round, in the next round it's likely that all three users will uncover a new set of problems, since they won't be getting stuck on the first set of problems.

Testing only three or four users also makes it possible to test and debrief in the same day, so you can take advantage of what you've learned right away. Also, when you test more than four at a time, you usually end up with more notes than anyone has time to process—many of them about things that are really “nits,” which can actually make it harder to see the forest for the trees.

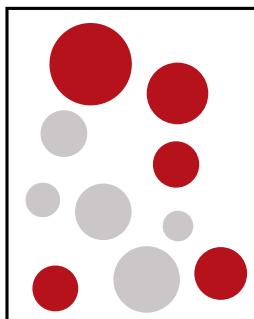
In fact this is one of the reasons why I've almost completely stopped generating written reports (what I refer to as the “big honking report”) for my expert reviews and for usability tests. I finally realized that for most Web teams their ability to *find* problems greatly exceeds the resources they have available to fix them, so it's important to stay focused on the most serious problems. Instead of written reports, nowadays I report my findings in a conference call with the entire Web team, which may last for an hour or two. By the end of the call, we've all agreed which problems are most important to fix, and how they're going to fix them.

² See Jakob Nielsen's March 2000 Alertbox column “Why You Only Need to Test with 5 Users” at www.useit.com for a good discussion of the topic.

³ If you're hiring someone to do the testing for you and money is no object, you might as well test six or eight users since the additional cost per user will be comparatively low. But only if it won't mean you'll do fewer rounds of testing.

ONE TEST WITH 8 USERS

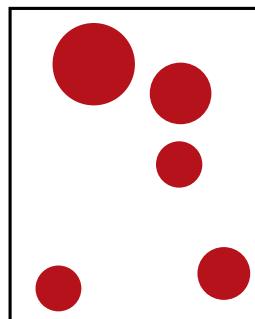
8 users



Eight users may find more problems in a single test.

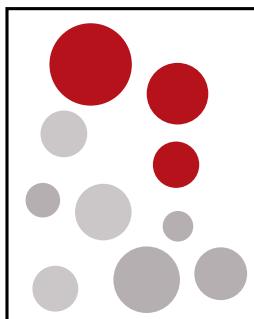
But the worst problems will usually keep them from getting far enough to encounter some others.

TOTAL PROBLEMS FOUND: 5



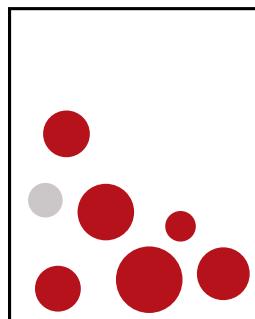
TWO TESTS WITH 3 USERS

First test: 3 users



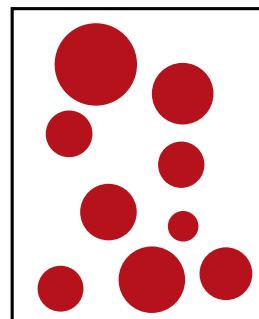
Three users may not find as many problems in a single test.

Second test: 3 users



But in the second test, with the first set of problems fixed, they'll find problems they couldn't have seen in the first test.

TOTAL PROBLEMS FOUND: 9



Recruit loosely and grade on a curve

When people decide to test, they often spend a lot of time trying to recruit users who they think will precisely reflect their target audience—for instance, male accountants between the ages of 25 and 30 with one to three years of computer experience who have recently purchased expensive shoes.

The best-kept secret of usability testing is the extent to which *it doesn't much matter who you test*.

For most sites, all you really need are people who have used the Web enough to know the basics.

If you can afford to hire someone to recruit the participants for you *and it won't* reduce the number of rounds of testing that you do, then by all means be as specific as you want. But if finding the ideal user means you're going to do fewer tests, I recommend a different approach:

Take anyone you can get (within limits) and grade on a curve.

In other words, try to find users who reflect your audience, but don't get hung up about it. Instead, try to make allowances for the differences between the people you test and your audience. I favor this approach for three reasons:

- › **We're all beginners under the skin.** Scratch an expert and you'll often find someone who's muddling through—just at a higher level.
- › **It's usually not a good idea to design a site so that only your target audience can use it.** If you design a site for accountants using terminology that you think all accountants will understand, what you'll probably discover is that a small but not insignificant number of accountants won't know what you're talking about. And in most cases, you need to be addressing novices as well as experts anyway, and if your grandmother can use it, an expert can.
- › **Experts are rarely insulted by something that is clear enough for beginners.** Everybody appreciates clarity. (True clarity, that is, and not just something that's been "dumbed down.")

The exceptions:

- › **If your site is going to be used almost exclusively by one type of user and it's no harder to recruit from that group,** then do it. For instance, if your audience will be almost entirely women, then by all means test just women.
- › **If your audience is split between clearly defined groups with very divergent interests and needs,** then you need to test users from each group at least once. For instance, if you're building a university site, for at least one round of testing you want to recruit two students, two professors, two high school seniors, and two administrators. But for the other rounds, you can choose any mix.

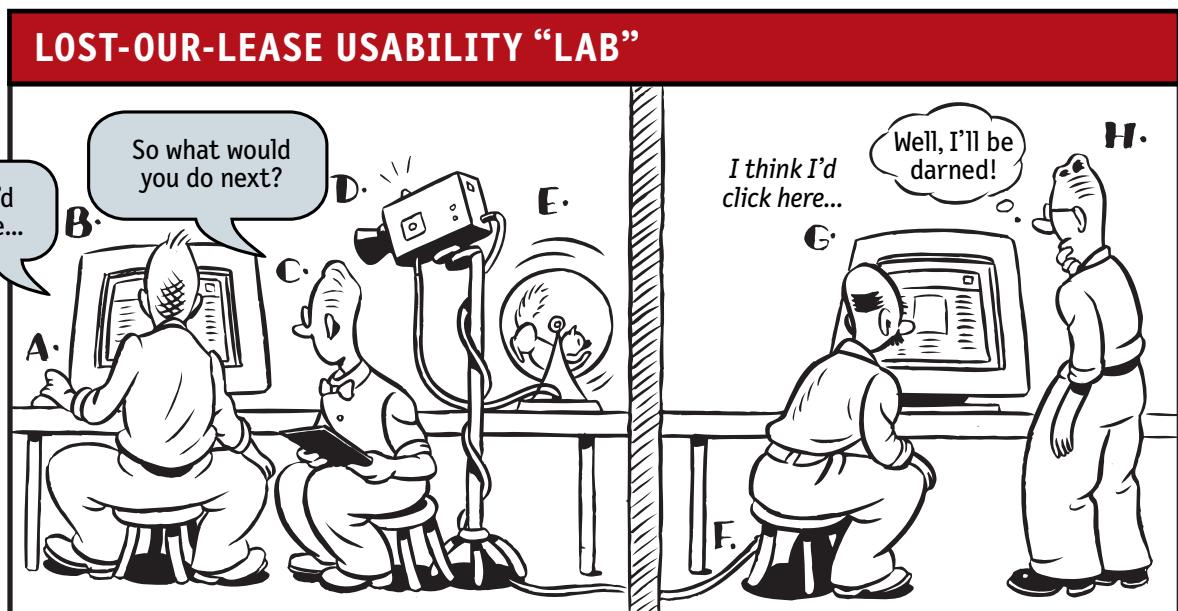
- › **If using your site requires specific domain knowledge** (e.g., a currency exchange site for money management professionals), then you need to recruit people with that domain knowledge for at least one round of tests. But don't do it for every round if it will reduce the number of tests you do.

When you're recruiting:

- › **Offer a reasonable incentive.** Typical stipends for a one-hour test session range from \$50 for “average” Web users to several hundred dollars for professionals from a specific domain, like cardiologists for instance. I like to offer people a little more than the going rate, since (a) it makes it clear that I value their opinion, and (b) people tend to show up on time, eager to participate. Remember, even if the session is only 30 minutes, people usually have to block out another hour for travel time. Also, I'd rather have people who are curious about the process than people who are desperate for the money.
- › **Keep the invitation simple.** “We need to have a few people look at our Web site and give us some feedback. It's very easy, and would take about forty-five minutes to an hour. And you'll be paid \$__ for your time.”
- › **Avoid discussing the site (or the organization behind the site) beforehand.** You want their first look to tell you whether they can figure out what it is from a standing start. (Of course, if they're coming to your office, they'll have a pretty good idea whose site it is.)
- › **Don't be embarrassed to ask friends and neighbors.** You don't have to feel like you're imposing if you ask friends or neighbors to participate. Most people enjoy the experience. It's fun to have someone take your opinion seriously and get paid for it, and they often learn something useful that they didn't know about the Web or computers in general.

Where do you test?

All you really need is an office or conference room with two chairs, a PC or Mac (with an Internet connection, if you're testing a live site), a camcorder, a long video cable, and a tripod.



Test subject (A) sits in front of computer monitor (B), while facilitator (C) tells him what to do and asks questions. Camcorder (D) powered by squirrel (E) is pointed at the monitor to record what the subject sees.

Meanwhile, cable (F) carries signal from camcorder to TV (G) in a nearby room where interested team members (H) can observe.

You can use the video cable to run the signal from the camcorder to a TV in another office—or even a cubicle—nearby so everyone on the development team can watch without disturbing the user.

The camcorder needs to transmit what the user sees (the computer screen or the designs on paper, depending on what you're testing) and what the user and the facilitator say. In a usability lab, you'll often see a second camera used to show the observers the user's face, but this isn't necessary: The user's tone of voice usually conveys frustration pretty effectively.

You can buy the camcorder, TV, cable, and tripod for less than \$600. But if your budget won't stretch that far, you can probably twist somebody's arm to bring in a camcorder from home on test days.

I don't recommend using the camcorder to videotape the sessions. In fact, I used to recommend not doing any video recording at all, because the tapes were almost never used and it made the whole process more complicated and expensive.

In the past few years though, three things have changed: PCs have gotten much faster, disk drives have gotten much larger, and screen recording software has improved dramatically. Screen recorders like Camtasia⁴ run in the background on the test PC and record everything that happens on the screen and everything the user and the facilitator say in a video file you can play on the PC. It turns out that these files are very valuable because they're much easier to review quickly than videotape and they're very easy to share over a network. I recommend that you always use a screen recorder during user tests.

Who should do the testing?

Almost anyone can facilitate a usability test; all it really takes is the courage to try it. With a little practice, most people can get quite good at it.

Try to choose someone who tends to be patient, calm, empathetic, a good listener, and inherently fair. Don't choose someone whom you would describe as "definitely not a people person" or "the office crank."

Who should observe?

Anybody who wants to. It's a good idea to encourage everyone—team members, people from marketing and business development, and any other stakeholders—to attend.

When people ask me how they can convince senior management that their organization should be investing in usability, my strongest recommendation doesn't have anything to do with things like "demonstrating return on

⁴ There are a number of screen recorders available, but I'm partial to Camtasia, made by TechSmith, the same company that makes the screen capture program SnagIt (<http://www.techsmith.com>). It's very reliable and has a number of extremely useful features, and it costs about \$300. For \$1,000 more, they have a product called Morae specifically designed for capturing usability tests—sort of like Camtasia on steroids—which allows observers to view the test live on a networked PC, eliminating the need for a camcorder.

investment.” The tactic that I think works best is getting management to observe even one user test. Tell them that you’re going to be doing some usability testing and it would be great for the Web team’s morale if they could just poke their head in for a few minutes. In my experience, executives often become fascinated and stay longer than they’d planned, because it’s the first time they’ve seen their site in action and it’s often not nearly as pretty a picture as they’d imagined.

What do you test, and when do you test it?

The key is to start testing early (it’s really *never* too early) and test often, at each phase of Web development.

Before you even begin designing your site, you should be testing comparable sites. They may be actual competitors, or they may be sites that are similar in style, organization, or features to what you have in mind.

Use them yourself, then watch one or two other people use them and see what works and what doesn’t. Many people overlook this step, but it’s invaluable—like having someone build a working prototype for you for free.

If you’ve never conducted a test before testing comparable sites, it will give you a pressure-free chance to get the hang of it. It will also give you a chance to develop a thick skin. The first few times you test your own site, it’s hard not to take it personally when people don’t get it. Testing someone else’s site first will help you see how people react to sites and give you a chance to get used to it.

Since the comparable sites are “live,” you can do two kinds of testing: “Get it” testing and key tasks.

- › **“Get it” testing** is just what it sounds like: show them the site, and see if they get it—do they understand the purpose of the site, the value proposition, how it’s organized, how it works, and so on.
- › **Key task testing** means asking the user to do something, then watching how well they do.

As a rule, you'll always get more revealing results if you can find a way to observe users doing tasks that they have a hand in choosing. It's much better, for instance, to say "Find a book you want to buy, or a book you bought recently" than "Find a cookbook for under \$14." When people are doing made-up tasks, they have no emotional investment in it, and they can't use as much of their personal knowledge.

As you begin designing your own site, it's never too early to start showing your design ideas to users, beginning with your first rough sketches. Designers are often reluctant to show work in progress, but users may actually feel freer to comment on something that looks unfinished, since they know you haven't got as much invested in it and it's still subject to change. Also, since it's not a polished design, users won't be distracted by details of implementation and they can focus on the essence and the wording.

Later, as you begin building parts of the site or functioning prototypes, you can begin testing key tasks on your own site.

I also recommend doing what I call Cubicle tests: Whenever you build a new kind of page—particularly forms—you should print the page out and show it to the person in the next cubicle and see if they can make sense out of it. This kind of informal testing can be very efficient, and eliminate a lot of potential problems.

A sample test session

Here's an annotated excerpt from a typical—but imaginary—test session. The site is real, but it has since been redesigned. The participant's name is Janice, and she's about 25 years old.

INTRODUCTION

164

Hi, Janice. My name is Steve Krug, and I'm going to be walking you through this session.

You probably already know, but let me explain why we've asked you to come here today. We're testing a Web site that we're working on so we can see what it's like for actual people to use it.

I want to make it clear right away that we're testing the *site*, not you. You can't do anything wrong here. In fact, this is probably the one place today where you don't have to worry about making mistakes.

We want to hear exactly what you think, so please don't worry that you're going to hurt our feelings.⁵ We want to improve it, so we need to know honestly what you think.

As we go along, I'm going to ask you to think out loud, to tell me what's going through your mind. This will help us.

This whole first section is the script that I use when I conduct tests.⁵

I always have a copy in front of me, and I don't hesitate to read from it, but I find it's good to ad lib a little, even if it means making mistakes. When the users see that I'm comfortable making mistakes, it helps take the pressure off them.

⁵ A copy of the script is available on my Web site (www.sensible.com) so you can download it and edit it for your own use.

⁶ If you didn't work on the part that's being tested, you can also say, "Don't worry about hurting my feelings. I didn't create the pages you're going to look at."

If you have questions, just ask. I may not be able to answer them right away, since we're interested in how people do when they don't have someone sitting next to them, but I will try to answer any questions you still have when we're done.

We have a lot to do, and I'm going to try to keep us moving, but we'll try to make sure that it's fun, too.

You may have noticed the camera. With your permission, we're going to record the computer screen and what you have to say. The recording will be used only to help us figure out how to improve the site, and it won't be seen by anyone except the people working on the project. It also helps me, because I don't have to take as many notes. There are also some people watching the screen in another room.

If you would, I'm going to ask you to sign something for us. It simply says that we have your permission to record you, but that it will only be seen by the people working on the project. It also says that you won't talk to anybody about what we're showing you today, since it hasn't been made public yet.

Do you have any questions before we begin?

No. I don't think so.

It's important to mention this, because it will seem rude not to answer their questions as you go along. You have to make it clear before you start that (a) it's nothing personal, and (b) you'll try to answer them at the end if they still want to know.

At this point, most people will say something like, "I'm not going to end up on *America's Funniest Home Videos*, am I?"

Give them the release and non-disclosure agreement (if required) to sign. Both should be as short as possible and written in plain English.⁷

⁷ You'll find a sample recording consent form on my Web site.

BACKGROUND QUESTIONS

166

Before we look at the site, I'd like to ask you just a few quick questions. First, what's your occupation?

I'm a router.

I've never heard of that before. What does a router do, exactly?

Not much. I take orders as they come in, and send them to the right office.

Good. Now, roughly how many hours a week would you say you spend using the Internet, including email?

Oh, I don't know. Probably an hour a day at work, and maybe four hours a week at home. Mostly that's on the weekend. I'm too tired at night to bother. But I like playing games sometimes.

How do you spend that time? In a typical day, for instance, tell me what you do, at work and at home.

Well, at the office I spend most of my time checking email. I get *a lot* of email, and a lot of it's junk but I have to go through it anyway. And sometimes I have to research something at work.

I find it's good to start with a few questions to get a feel for who they are and how they use the Internet. It gives them a chance to loosen up a little and gives you a chance to show that you're going to be listening attentively to what they say—and that there are no wrong or right answers.

Don't hesitate to admit your ignorance about anything. Your role here is not to come across as an expert, but as a good listener.

Notice that she's not sure how much time she really spends on the Internet. Most people aren't. Don't worry. Accurate answers aren't important here. The main point here is just to get her talking and thinking about how she uses the Internet and to give you a chance to gauge what kind of user she is.

<p>Do you have any favorite Web sites?</p> <p>Yahoo, I guess. I like Yahoo, and I use it all the time. And something called Snakes.com, because I have a pet snake.</p>	
<p>Really? What kind of snake?</p> <p>A python. He's about four feet long, but he should get to be eight or nine when he's fully grown.</p>	<p>Don't be afraid to digress and find out a little more about the user, as long as you come back to the topic before long.</p>
<p>Wow. OK, now, finally, have you bought anything on the Internet? How do you feel about buying things on the Internet?</p> <p>I've bought some things recently. I didn't do it for a long time, but only because I couldn't get things delivered. It was hard to get things delivered, because I'm not home during the day. But now one of my neighbors is home all the time, so I can.</p>	
<p>And what have you bought?</p> <p>Well, I ordered a raincoat from L.L. Bean, and it worked out <i>much</i> better than I thought it would. It was actually pretty easy.</p> <p>OK, great. We're done with the questions, and we can start looking at things.</p> <p>OK, I guess.</p>	

REACTIONS TO THE HOME PAGE

First, I'm just going to ask you to look at this page and tell me what you think it is, what strikes you about it, and what you think you would click on first.

For now, don't actually click on anything. Just tell me what you *would* click on.

And again, as much as possible, it will help us if you can try to think out loud so we know what you're thinking about.

The browser has been open, but minimized. At this point, I reach over and click to maximize it.



Well, I guess the first thing I notice is that I like the color. I like the shade of orange, and I like the little picture of the sun [at the top of the page, in the eLance logo].

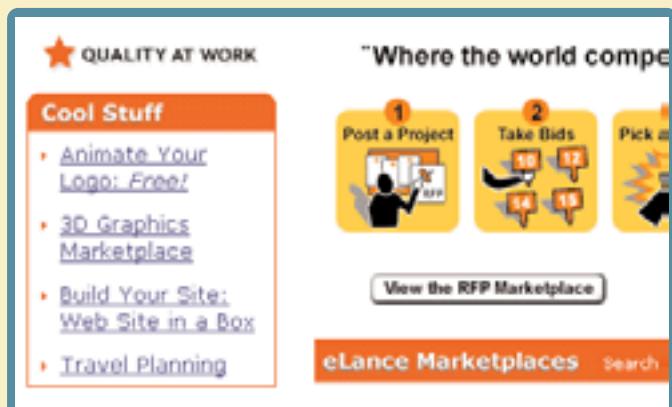
Let's see. [Reads.] "The global services market." "Where the world comes to get your job done."

In an average test, it's just as likely that the next user will say that she hates this shade of orange and that the drawing is too simplistic. Don't get too excited by individual reactions to site aesthetics.



I don't know what that means. I have no idea.

"Animate your logo free." [Looking at the Cool Stuff section on the left.] "3D graphics marketplace." "eLance community." "eLance marketplace."



There's a lot going on here. But I have no idea what any of it is.

If you had to take a guess, what do you think it might be?

Well, it seems to have something to do with buying and selling...something.

[Looks around the page again.] Now that I look at the list down here [the Yahoo-style category list halfway down the page], I guess maybe it must be services. Legal, financial, creative...they all sound like services.

This user is doing a good job of thinking out loud on her own. If she wasn't, this is where I'd start asking her, "What are you thinking?"

Business
Consulting, Data Entry, Report Production, Startup Services, Transcription, Translation, Word Processing...
[RFPs](#) | [Fixed-Price](#)

Financial
Accounting, Auditing, Bookkeeping, Estate Planning, Insurance, Financial Planning, Loans, Taxes...
[RFPs](#) | [Fixed-Price](#)

Legal
Claims, Corporate, Family, Immigration, Intellectual Property, International, Patent, Personal, Research, Wills/Trusts...
[RFPs](#) | [Fixed-Price](#)

Computer
Consulting, Software Development, Tech Support...
[RFPs](#) | [Fixed-Price](#)

Creative
Design, Illustration, Music, Photography, Writing...
[RFPs](#) | [Fixed-Price](#)

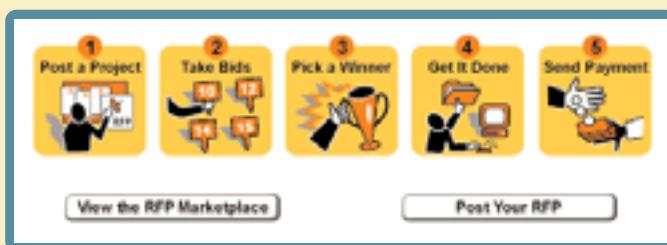
Marketing
Advertising, Direct...

So I guess that's what it is. Buying and selling services. Maybe like some kind of online *Yellow Pages*.

OK. Now, if you were at home, what would you click on first?

I guess I'd click on that 3D graphics thing.
I'm interested in 3D graphics.

Before you click on it, I have one more question. What about these pictures near the top of the page—the ones with the numbers? What did you make of them?



I noticed them, but I really didn't try to figure them out. I guess I thought they were telling me what the steps in the process would be.

Any reason why you didn't pay much attention to them?

No. I guess I just wasn't ready to start the process yet. I didn't know if I *wanted* to use it yet. I just wanted to look around first.

OK. Great.

I ask this question because the site's designers think most users are going to start by clicking on the pictures of the five steps, and that everyone will at least look at them.

TESTING A TASK

172

OK, now we're going to try something else.

Can you think of something you might want to post as a project if you were using this site?

Hmm. Let me think. I think I saw "Home Improvement" there somewhere. We're thinking of building a deck. Maybe I would post that.

So if you were going to post the deck as a project, what would you do first?

I guess I'd click on one of the categories down here. I think I saw home improvement. [Looks.] There it is, under "Family and Household."

So what would you do?

Well, I'd click.... [Hesitates, looking at the two links under "Family and Household."]

Now I give her a task to perform so we can see whether she can use the site for its intended purpose.

Whenever possible, it's good to let the user have some say in choosing the task.

Family & Household
Food & Cooking, Gardening,
Genealogy, Home Improvement,
Interior Design, Parenting, Pets, Real
Estate...
[RFPs](#) | [Fixed-Price](#)

Well, now I'm not sure *what* to do. I can't click on Home Improvement, so it looks like I have to click on either "RFPs" or "Fixed-Price." But I don't know what the difference is.

Fixed price I sort of understand; they'll give me a quote, and then they have to stick to it. But I'm not sure what RFPs is.

As it turns out, she's mistaken. Fixed-price (in this case) means services available for a fixed hourly rate, while an RFP (or Request for Proposal) is actually the choice that will elicit quotes. This is the kind of misunderstanding that often surprises the people who built the site.

Well, which one do you think you'd click on?

Fixed price, I guess.

Why don't you go ahead and do it?

From here on, I just watch while she tries to post a project, letting her continue until either (a) she finishes the task, (b) she gets really frustrated, or (c) we're not learning anything new by watching her try to muddle through.

I'd give her three or four more tasks to do, which should take not more than 45 minutes altogether.

Review the results right away

After each round of tests, you should make time as soon as possible for the development team to review everyone's observations and decide what to do next. I strongly recommend that you do three or four tests in a morning and then debrief over lunch.

You're doing two things at this meeting:

- › **Triage**—reviewing the problems people saw and deciding which ones need to be fixed.
- › **Problem solving**—figuring out how to fix them.

It might seem that this would be a difficult process. After all, these are the same team members who've been arguing about the right way to do things all along. So what's going to make this session any different?

Just this:

The important things that you learn from usability testing usually *just make sense*. They tend to be obvious to anyone who watches the sessions.

Also, the experience of seeing your handiwork through someone else's eyes will often suggest entirely new solutions for problems, or let you see an old idea in a new light.

And remember, this is a cyclic process, so the team doesn't have to agree on the perfect solution. You just need to figure out what to try next.

Typical problems

Here are the types of problems you're going to see most often when you test:

- › **Users are unclear on the concept.** They just don't get it. They look at the site or a page and they either don't know what to make of it, or they think they do but they're wrong.
- › **The words they're looking for aren't there.** This usually means that either

(a) the categories you've used to organize your content aren't the ones they would use, or (b) the categories are what they expect, but you're just not using the names they expect.

- › **There's too much going on.** Sometimes what they're looking for is right there on the page, but they're just not seeing it. In this case, you need to either (a) reduce the overall noise on the page, or (b) turn up the volume on the things they need to see so they "pop" out of the visual hierarchy more.

Some triage guidelines

Here's the best advice I can give you about deciding what to fix—and what not to.

- › **Ignore "kayak" problems.** In any test, you're likely to see several cases where users will go astray momentarily but manage to get back on track almost immediately without any help. It's kind of like rolling over in a kayak; as long as the kayak rights itself quickly enough, it's all part of the so-called fun. In basketball terms, no harm, no foul.

As long as (a) everyone who has the problem notices that they're no longer headed in the right direction quickly, and (b) they manage to recover without help, and (c) it doesn't seem to faze them, you can ignore the problem. In general, if the user's second guess about where to find things is always right, that's good enough.

Of course, if there's an easy and obvious fix that won't break anything else, then by all means fix it. But kayak problems usually don't come as a surprise to the development team. They're usually there because of some ambiguity for which there is no simple resolution. For example, there are usually at least one or two oddball items that don't fit perfectly into any of the top-level categories of a site. So half the users may look for movie listings in *Lifestyles* first, and the other half will look for them in *Arts* first. Whatever you do, half of them are going to be wrong on their first guess, but everyone will get it on their second guess, which is fine.⁸

⁸ You may be thinking "Well, why not just put it in both categories?" In general, I think it's best for things to "live" in only one place in a hierarchy, with a prominent "see also" crosslink in any other places where people are likely to look for them.

- › **Resist the impulse to add things.** When it's obvious in testing that users aren't getting something, most people's first reaction is to add something, like an explanation or some instructions.

Very often, the right solution is to take something (or things) away that are obscuring the meaning, rather than adding yet another distraction.

- › **Take “new feature” requests with a grain of salt.** People will often say, “I'd like it better if it could do x .” It always pays to be suspicious of these requests for new features. If you probe deeper, it often turns out that they already have a perfectly fine source for x and wouldn't be likely to switch; they're just telling you what they like.
- › **Grab the low-hanging fruit.** The main thing you're looking for in each round of testing is the big, cheap wins. These fall into two categories:
 - › **Head slappers.** These are the surprises that show up during testing where the problem and the solution were obvious to everyone the moment they saw the first user try to muddle through. These are like found money, and you should fix them right away.
 - › **Cheap hits.** Also try to implement any changes that (a) require almost no effort, or (b) require a *little* effort but are highly visible.

And finally, there's one last piece of advice about “making changes” that deserves its own section:

Don't throw the baby out with the dishes

Like any good design, successful Web pages are usually a delicate balance, and it's important to keep in mind that even a minor change can have a major impact. Sometimes the real challenge isn't fixing the problems you find—it's fixing them *without* breaking the parts that already work.

Whenever you're making a change, think carefully about what else is going to be affected. In particular, when you're making something more prominent than it was, consider what else might end up being de-emphasized as a result.

One morning a month: that's all we ask

Ideally, I think every Web development team should spend one morning a month doing usability testing.

In a morning, you can test three or four users, then debrief over lunch. That's it.

When you leave lunch, the team will have decided what you're going to fix, and you'll be done with testing for the month. No reports, no endless meetings.

Doing it all in a morning also greatly increases the chances that most team members will make time to come and watch at least some of the sessions, which is highly desirable.

If you're going to try doing some testing yourself—and I hope you will—you'll find some more advice about how to do it in a chapter called “Usability testing: The Movie” that was in the first edition of this book.⁹ My next book is going to be all about do-it-yourself usability testing, but I do *not* want you to wait for it before you start testing. Start now.

⁹ You can download the chapter for free at <http://www.sensible.com/secondedition>.

Web form design: filling in the blanks

Luke Wroblewski

“Forms suck. If you don’t believe me, try to find people who like filling them in. You may turn up an accountant who gets a rush when wrapping up a client’s tax return or perhaps a desk clerk who loves to tidy up office payroll. But for most of us, forms are just an annoyance. What we want to do is to vote, apply for a job, buy a book online, join a group, or get a rebate back from a recent purchase. Forms just stand in our way.”



WEB FORM DESIGN

Filling in the Blanks

by LUKE WROBLEWSKI foreword by Jared Spool

Rosenfeld



Form Organization

180

What to Include	32
Have a Conversation	37
Organizing Content	40
Group Distinctions	48
Best Practices	56

Web Form Design: Filling in the Blanks by Luke Wroblewski
Rosenfeld Media, 2008; version 1.0

Although many visual and interaction design considerations play an important role in how people complete forms, it's often the content within the form and how we organize it that either leaves people scratching their heads or allows them to whiz through unperturbed.

What to Include

People need to parse every question you ask them, formulate their response to that question, and then enter their response into the space you have provided. The best way to speed up that process is not to ask the question at all. That means if you want to be vigilant about optimizing your forms, put every question you are asking people to the test. Do you really need to ask this question? Is it information that you can get automatically? Is there a better time or place to get an answer from people? Though this process appears tedious, you may be surprised when you discover what you can leave off your forms.

Deciding what stays on a form may mean challenging the information collected when

the form was a paper document. Often, legacy questions that are no longer applicable are simply ported over when a paper form is digitized.

Agreeing as to which questions should remain on a form may also be a discussion among several departments in your company or organization. The marketing team may have specific questions to understand customers better. The engineering team may require specific information to identify unique individuals. The legal team might mandate certain terms and conditions that have to be accepted by new customers. And the list goes on.

Though all these teams may have questions they want to pose to your customers, your forms need to speak with one voice. To achieve that goal, teams will need to come together and work out which questions make it into each form. Take a look at Caroline Jarrett's "Keep, cut, postpone, and explain" framework (outlined in the sidebar) for a way to decide what makes the cut.

33
Chapter 2
Form Organization

Perspective: Caroline Jarrett

Usability consultant, Effortmark Ltd.

Co-author: *Forms That Work* (Morgan Kaufmann, in press)

Co-author: *User Interface Design and Evaluation* (Morgan Kaufmann, 2005)

People Before Pixels

WHAT TO THINK ABOUT BEFORE YOU START

I love forms, mostly because they offer so many opportunities for improvement. And I love discussing forms with designers. So I encourage people to write to me with questions about their forms.

Often, these questions show that designers are thinking hard, which is great, but perhaps they're missing the people aspect while concentrating on pixels—the fine details, such as whether to put a colon on the end of the label. Users really don't care about colons.

USERS REALLY DO CARE ABOUT WHAT THEY'RE ASKED AND WHY

Users care about what they're asked, why they are asked it, and beyond everything else, whether those questions are appropriate to the context, meaning whatever the user is trying to achieve by filling in the form.

For example: a street address. If you have to put your street address into a Web site before browsing it, chances are that you'll react badly. Many of us maintain a convenient false set of personal answers, including an address and email that we use when we consider that it's impertinent to be asked personal questions right now.

But if you've decided on buying something that needs to be shipped, then it would be distinctly strange if the site did not ask you for a street address. And you'll probably take care to enter a real address accurately.

Perspective: Caroline Jarrett (Continued)

START BY THINKING ABOUT PEOPLE AND RELATIONSHIPS

So before you start thinking about where to place your questions on the page, think about people and relationships.

Why are users filling in your form? What is their relationship to your organization? Do they feel good or bad about it? Is this form just another stepping stone on the road to their continuing enthusiasm for your product, service, or whatever? Or is it a fearful barrier that's keeping them from something else they'd prefer to be doing? Or are you just battling indifference: they don't care one way or another, and may just bail because they can't see the point? If you don't know enough about your users to be sure, then ask them. Watch them using your Web site or talk to them, somehow.

If you've already got questions for your form, then why are you asking those particular questions? And why are you asking them right now, at this point in the relationship? If you don't know enough about your organization to be sure, then investigate. Find someone in the organization who does know. If there isn't anyone, that's telling you that maybe your whole approach needs rethinking. Is the form necessary at all?

KEEP, CUT, POSTPONE, OR EXPLAIN: FOUR STRATEGIES FOR BETTER QUESTIONS

Maybe, as with my "shipping" example, you and your users are in harmony: you're asking for answers that they are eager to give you. Well done—keep those questions and move to thinking about the details of design.

35
Chapter 2
Form Organization

Perspective: Caroline Jarrett (Continued)

But perhaps you're asking a question that you don't *really* need *right now*. Cut: get rid of the question and help everyone. That translates to less work for you in design, less work for your users, and no long-term storage.

Or maybe it's the "right now" part: postpone asking that question until later, until the point where it moves from unnecessary or intrusive to harmonious.

Or maybe it's one of those difficult questions that your users don't want to answer: personal data, such as a phone number, or something that requires research or extra thought. But you've investigated your form, you know that there is a real value to your organization in asking these questions, and there is some important reason why you have to ask them ahead of time. Your strategy is to explain: write a very short but clear reason why you're asking. Make sure it offers a benefit to the user—for example, "Asking you this now helps us to process your order more quickly."

And if you can't think of any benefit to the user, then you'd better go back to finding out whether you really need that question because you're going to find that you lose users at that point in the form.

YOUR VIEW, MY VIEW: BALANCING USER AND BUSINESS NEEDS

Of course, there's nothing new about being told to think about your users before starting your design. My message is about balancing user needs and business needs—harder work than stressing about labels and colons, but with a much greater impact on your form design.

Like I said: People before pixels.

36
Chapter 2
Form Organization

Have a Conversation

Because forms facilitate conversation between a person and a company or organization, it helps to think about organizing the structure of a form as a conversation. Consider the following scenario.

You encounter a stranger who asks you: “What’s your name?” “What’s your address?” “What’s your email address?” “What’s your birth date?” Before too long you, find yourself asking: “Who is this person?” “Why does he (or she) need all this information?” “Why am I telling him (or her) all this?” Quite quickly, you become uneasy and wish the stranger would tell you something about himself or herself instead of barraging you with questions. That barrage of questions is—of course—our friend, the form.

Thinking about how a form can be organized as a conversation instead of an interrogation can go a long way toward making new customers feel welcome. I still have a vivid memory of a woman who was interviewed

during a field study for a major Web retailer remarking, “This site wants to know so much about me, but I know nothing about it.”

Giving people the confidence to complete forms starts with how we ask them the questions required to complete a form successfully. Input fields are the elements on a form responsible for gathering people’s answers to our questions. Labels are the form elements responsible for asking the questions. Whenever these two elements can act as a natural part of a meaningful conversation, people are likely to respond with answers easily and readily.

Consider the difference between the following questions from two different versions of the Yahoo! registration form, as shown in Figure 2.1.

FIGURE 2.1
Two ways conversational language can clarify questions.
“Day” and “Year” vs. “dd” and “yyyy”; “Preferred content” vs. “I prefer content from.”

Which version seems more approachable? Which one are you more likely to have an answer for? Treating inputs as part of the question being asked (the label) mirrors the way we answer questions in the real world. This becomes even more important as the questions you ask become more complex or unfamiliar.

Consider the label “Issuing Bank.” What is that asking? Now, if we rephrased it as “What bank issued you this document?” odds are that you’d have a quicker answer. Of course, both of these labels will be made clearer by their surrounding context. For example, are you filling a form about a missing financial document or a form to set up a new online account?

The terms you use in your labels also play a pivotal role in determining how quickly people can provide an answer. To continue with our banking example, do people understand the term “issuing”? Is that vocabulary they’d use, or is it a term used by the bank? Perhaps

39
Chapter 2
Form Organization

people think instead: “Which bank gave you this document?” Using the terms your customers use to describe their actions helps frame questions in a more understandable way.

This doesn’t mean that all of the labels on a form should be reworded as sentences. There are many instances when concise, single-word labels work *much* better than longer, more descriptive labels. But when there’s potential ambiguity in your questions, clear conversational language often helps clear things up.

Organizing Content

In order to keep the conversation flowing smoothly, it’s a good idea to organize the questions you’re asking people into meaningful groups. Depending on their size and context, these groups could then be presented across multiple Web pages or as sections of a single Web page.

As an example, the Yahoo! registration form in Figure 2.2 groups questions about you,

40
Chapter 2
Form Organization

the account you are creating, a way for you to reaccess your account, and a few trust and safety items (terms of service and spam protection) into four distinct sections. These

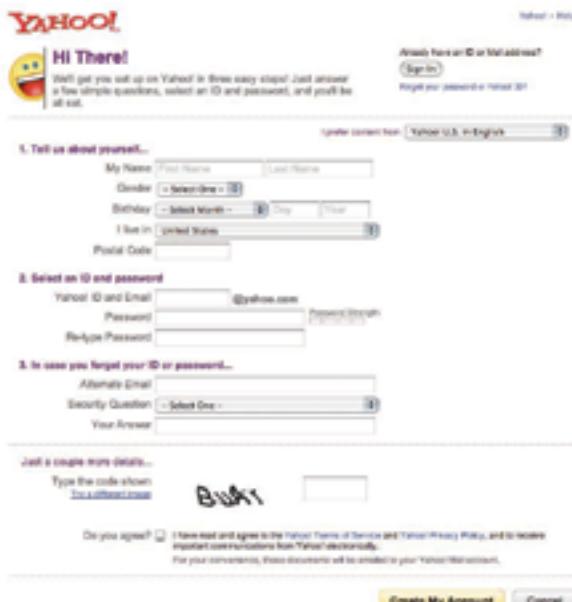


FIGURE 2.2 <http://www.flickr.com/photos/rosenfeldmedia/2367260580>
The new Yahoo! registration form uses a conversational tone to engage new members.

41
Chapter 2
Form Organization

Web Form Design: Filling in the Blanks by Luke Wroblewski
Rosenfeld Media, 2008; version 1.0

sections are labeled with headers that stand out from the rest of the elements on the page. The bold purple font in which they are displayed carries more visual “weight” than the other form labels, allowing you to quickly scan the form to see what type of information you’ll need to provide.

Longer or more complex forms may need to distribute content groups across multiple pages, as seen in online real estate site Redfin’s form for buying houses online. This overly complex process—not through any fault of Redfin—also benefits from being organized in a way that allows people to easily scan required sections they need to answer. In case someone didn’t know what he was getting into when buying a home, Redfin’s eight-page form makes it vividly clear (see Figure 2.3)! It’s worth noting that forms this long benefit from additional feedback and interactions, which we’ll discuss in later chapters.

When deciding how to organize forms, designers will often wonder if they are better

42
Chapter 2
Form Organization

Other Costs (if this apply)
Who will pay the monthly transfer fee?
 Buyer
 Seller
Who will pay the city transfer fee?
 Buyer
 Seller
Who will pay the home owner's association transfer fee?
 Buyer
 Seller
Who will pay the the home owner's association transfer documents?
 Buyer
 Seller

Home Warranty
Do you want to order a home warranty?

Who will pay for the home warranty?
 Buyer
 Seller
How much home warranty coverage?

Which home warranty options do you want?
 Air conditioner
 Well
 Septic
 Pool
 Washer / Dryer / Refrigerator
Other:

Liquidated Damages
Liquidated damages can be assessed if the buyer fails to complete the purchase because of default. If the buyer agrees to use liquidated damages in case of default, then the seller retains the deposit initially paid by the buyer.
If you default, do you agree to pay liquidated damages?
 Yes
 No

Dispute Resolution
Rather than having disputes resolved in courts, buyers and sellers can agree to have all disputes resolved by arbitration as provided for California law.
Do you agree to submit disputes to neutral arbitration?
 Yes
 No

Expiration
When do you want your offer to expire? (Generally 3 calendar days after the buyer signs and dates the offer)

The offer will officially expire, be deemed rejected, and the deposit will be returned, unless the offer is signed by the seller and a copy of the offer is postmarked received by the buyer at 5 p.m. on the third day after the offer is presented to the buyer.
If the seller makes a counter-offer, your Redfin Agent will help you respond appropriately.

FIGURE 2.3  <http://www.flickr.com/photos/rosenfeldmedia/2366424765>
Redfin groups the myriad of steps required to purchase a home into a series of manageable content groups. Each section has a title and some also include a bit of descriptive text.

43
Chapter 2
Form Organization

Web Form Design: Filling in the Blanks by Luke Wroblewski
Rosenfeld Media, 2008; version 1.0

off grouping all their content areas into a single Web page or dividing them into a series of pages. And if a form is divided into a series of pages, how many pages is too many? The answer, of course, is... it depends. But we can get a better answer by understanding the context for each form we design. Who is filling the form in and why? Answering this up front allows us to think about our forms as a deliberate conversation with a specific person instead of the inputs for a database.

When you approach forms as a conversation, natural breaks will emerge between topics. First, let's talk about who you are. Now let's discuss where you live. When these distinct topics are short enough to fit into a few sections, a single Web page will probably work best to organize them. When each section begins to run long, multiple Web pages may be required to break up the conversation into meaningful, understandable topics.

44
Chapter 2
Form Organization

In certain situations, several sections with lots of questions may need to be asked in sequence because they don't make sense out of context. People need to see all the questions together in order to answer each. In this case, one long Web page may very well be the best answer.

In other situations, some sections will perform best after a form is completed. For instance, optional marketing questions such as "How did you first hear about us?" or "Would you like additional information about our services?" may actually get higher response rates when asked after someone has completed a form. In one redesign I've seen, asking these questions after a registration form was filled out increased answers by almost 40 percent! The reason behind this may be that optional questions feel less invasive when presented as follow-up topics instead of requirements for form completion.

Because your forms aren't alone on the Web, another way to decide how to structure your conversations with customers is to conduct a

45
Chapter 2
Form Organization

Web conventions survey to see if any patterns emerge. A Web conventions survey is simply a comparison of design solutions across a number of similar Web sites. It usually helps to look at the top performing sites in a specific category (like ecommerce) to ensure that the sites being compared share common measures of success.

A Web conventions survey may lead you to uncover common form organization structures that have emerged on the Web. For instance, mapping out what information is asked in ecommerce shopping cart forms (see Figure 2.4) reveals some interesting insights. The first page tends to be Sign In; the second, personal information. After that it's usually shipping preferences. And so on.

Web conventions survey to see if any patterns emerge. A Web conventions survey is simply a comparison of design solutions across a number of similar Web sites. It usually helps to look at the top performing sites in a specific category (like ecommerce) to ensure that the sites being compared share common measures of success.

A Web conventions survey may lead you to uncover common form organization structures that have emerged on the Web. For instance, mapping out what information is asked in ecommerce shopping cart forms (see Figure 2.4) reveals some interesting insights. The first page tends to be Sign In; the second, personal information. After that it's usually shipping preferences. And so on.

46
Chapter 2
Form Organization

Web Form Design: Filling in the Blanks by Luke Wroblewski
Rosenfeld Media, 2008; version 1.0

Page 1	Page 2	Page 3	Page 4	Page 5	Page 6	Page 7	Page 8	Page 9	Page 10	Page 11	Page 12	Page 13	Page 14	Page 15
First Name														
Last Name														
Address														
City														
State														
Zip														
Phone														
E-mail														
Comments														

FIGURE 2.4 <http://www.flickr.com/photos/rosenfeldmedia/2367260748>
In this Web conventions survey, the questions asked by 15 ecommerce checkout forms are organized by the Web page on which they appear: page 1, page 2, and so on.

47
Chapter 2
Form Organization

These conventions can provide a great starting point for thinking about how to organize the conversation on your shopping cart form. Since people are likely to be familiar with these patterns, chances are they could work well in your ecommerce site. However, it's important to work from the patterns a Web conventions survey uncovers and not simply copy what the competition is doing on their site. Usually a direct replica of someone else's form organization won't be the right fit for your specific situation.

189

Group Distinctions

In both the Yahoo! and Redfin examples we saw earlier, each content group was visually differentiated from the rest of the form: a bold purple font on Yahoo! and a bold font and subtle background color on Redfin. As these examples illustrate, communicating meaningful distinctions between content groups doesn't require a lot of visual difference. In fact, too

48
Chapter 2
Form Organization

much contrast between content groupings often creates excessive visual noise that gets in the way of people's ability to scan a form.

Consider the differences between the following two forms in Figures 2.5 and 2.6. One relies on yellow borders, a yellow background color, red section headers, and merged table cells to group related content. The other simply relies on a subtle background color change to separate meaningful sections of the form. Using a minimum amount of visual information helps keep the focus on a form's content and not its presentation.

49
Chapter 2
Form Organization

FIGURE 2.5 <http://www.flickr.com/photos/rosenfeldmedia/2367260810>
Many distinct visual elements on this form get in the way of seeing the questions the form is asking.

FIGURE 2.6 <http://www.flickr.com/photos/rosenfeldmedia/2366425019>
A subtle background color change or thin rule is often all you need to effectively group related content in a form.

50
Chapter 2
Form Organization

Web Form Design: Filling in the Blanks by Luke Wroblewski
Rosenfeld Media, 2008; version 1.0

But even subtle distinctions between content groups can be overused. To account for what they consider to be shortcomings of left-aligned form labels, some designers opt to use alternating background colors to group left-aligned labels with their right-aligned inputs, as seen in Figure 2.7. However, eye-tracking studies done on label placement reveal that people generally don't have problems correlating inputs to labels in a left-aligned layout (as we'll see in Chapter 4). It just takes them longer to do so. As a result, this approach doesn't really solve the problem. In fact, it can actually create a different issue.

¹ Matteo Penzo's Label Placement in Forms study from UXmatters July 2006: <http://tinyurl.com/fefbx>



FIGURE 2.7 <http://www.flickr.com/photos/rosenfeldmedia/2366424971>
Although it may be tempting to use alternating background colors to group left-aligned labels and their corresponding inputs, these elements can add a lot of visual noise to a form.

Consider the example in Figure 2.8 where two different background colors are used to distinguish labels and inputs and a horizontal rule is used to separate each label and input field pair. This approach ultimately adds an additional 15 visual elements to the layout: the centerline, each background box, and each horizontal line. These elements begin to distract our eye and make it more difficult to focus on the most important elements in the

52
Chapter 2
Form Organization

layout: the labels and inputs. As information design expert Edward Tufte points out: “Information consists of differences that make a difference.”² In other words, any visual element that is not helping your layout ends up hurting it. This can be seen when you try to scan the left column of labels. Your eye repeatedly pauses (see the bottom of Figure 2.8) to consider each horizontal line and the

² Edward Tufte, *Envisioning Information*, 1990 Graphics Press

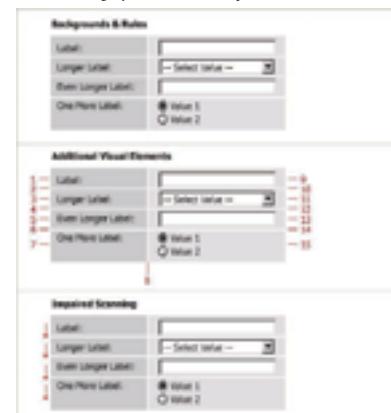


FIGURE 2.8 <http://www.flickr.com/photos/rosenfeldmedia/2366425057>
The addition of excessive visual elements can distract from a form’s primary content: and interrupt the scan line of a form.

53
Chapter 2
Form Organization

box created by each combination of line and background color.

192

Of course, this doesn't mean that background colors and rules should never be used within form layouts. They certainly have their place. But when thinking about how to distinguish between content groups, consider what the minimum amount of visual information needed is (see Figure 2.9). Chances are much more likely that it will become a distraction instead of an aid.

54
Chapter 2
Form Organization

Web Form Design: Filling in the Blanks by Luke Wroblewski
Rosenfeld Media, 2008; version 1.0

The screenshot shows a registration form titled "Enter Your Information" with a sub-instruction "(Already registered? Sign in)". Below this, a note says "Please enter your U.S. address and email address to create your account." The form is organized into several sections separated by thin horizontal lines:

- First Name** and **Last Name** fields.
- Street Address** and **City** fields.
- State** dropdown, **ZIP Code** input, and **Country or Region** dropdown (set to "United States"). A note states "U.S. addresses only, please."
- Phone Number** input field with a note "Needed if there are questions about your order."
- Email address** input field with a note "A valid email address is required to communicate with you."
- Re-enter Email address** input field.
- Create Password** input field with a note "Must be at least 8 characters, including a number or special character. (Example: 123!qwe123)" and a note "How secure is your password?" with a scale from "Very Weak" to "Very Strong". A note states "Create your password stronger - the higher, the better."
- Re-enter Password** input field.

At the bottom, a note says "By clicking "Register" you agree to eBay Express's privacy policy and terms of use. You also agree to be contacted for marketing purposes, but you can change your notification preferences in your account." A blue "Register" button is at the bottom right.

FIGURE 2.9 <http://www.flickr.com/photos/rosenfeldmedia/2367260984>
The eBay Express checkout form uses a thin rule to separate meaningful content sections. Just the minimum amount is needed to make a clear distinction.

55
Chapter 2
Form Organization

Best Practices

- Take the time to evaluate every question you are adding to your forms. Be vigilant about removing everything that isn't necessary.
- Strive for succinctness in all the questions (labels) you ask in your forms.
- When succinct labels may be misinterpreted, look for opportunities to use natural language to clarify the questions your forms ask people to answer.
- Ensure that your forms speak with one voice, despite questions from several different people or departments.
- Organize the content on your forms into logical groups to aid scanning and completion.
- When possible, structure your forms as a conversation. Natural breaks between topics will emerge that can help you organize your form.

193

Best Practices (continued)

- If a form naturally breaks down into a few short topics, a single Web page is likely to be a good way to organize the form.
- When a form contains a large number of questions that are only related by a few topics, multiple Web pages are probably a good way to organize the form.
- When a form contains a large number of questions related to a single topic, one long Web page is generally a good way to organize the form.
- Consider asking optional questions only after a form is completed. Chances are you'll get more answers than if these questions were part of the initial form.
- Consider using Web convention surveys to discover patterns in how forms are organized on specific kinds of sites.

Best Practices (continued)

- Use the minimal amount of visual information necessary to distinguish content groups.
- Use initial capital letters to make the titles of content groups easier to scan.

part 3—finishing

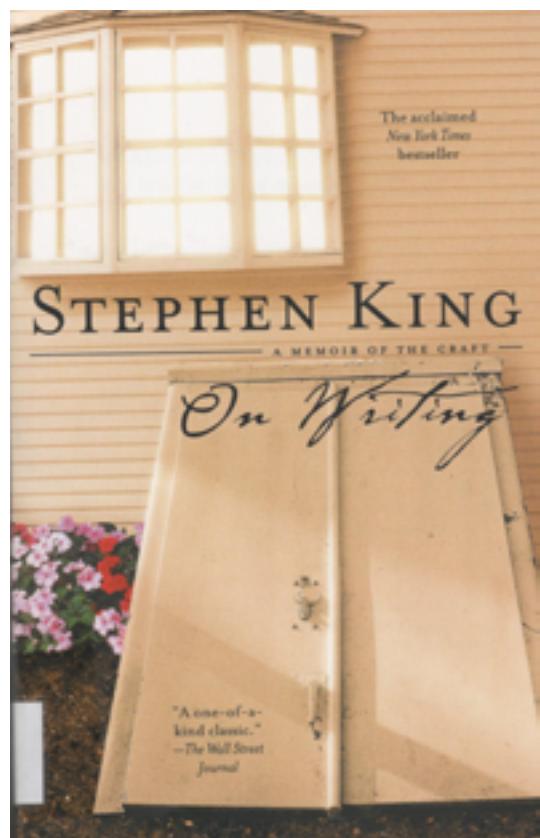


- a. God is in the details
- b. Form follows function
- c. The eye has the last word

On writing

Stephen King

“I believe that fear is at the root of most bad writing. If one’s writing is for one’s own pleasure, that fear may be mild—*timidity* is the word I’ve used here. If, however, one is working under a deadline—a school paper, a newspaper article, the SAT writing sample—that fear may be intense. Dumbo got airborne with the help of a magic feather; you may feel the urge to grasp **a passive verb or one of those nasty adverbs** for the same reason. Just remember before you do, that Dumbo didn’t need the feather; the magic was in him.”



On Writing

- 20 -

198
Hardly a week after being sprung from detention hall, I was once more invited to step down to the principal's office. I went with a sinking heart, wondering what new shit I'd stepped in.

It wasn't Mr. Higgins who wanted to see me, at least; this time the school guidance counsellor had issued the summons. There had been discussions about me, he said, and how to turn my "restless pen" into more constructive channels. He had enquired of John Gould, editor of Lisbon's weekly newspaper, and had discovered Gould had an opening for a sports reporter. While the school couldn't *insist* that I take this job, everyone in the front office felt it would be a good idea. *Do it or die*, the G.C.'s eyes suggested. Maybe that was just paranoia, but even now, almost forty years later, I don't think so.

I groaned inside. I was shut of *Dave's Rag*, almost shut of *The Drum*, and now here was the *Lisbon Weekly Enterprise*. Instead of being haunted by waters, like Norman Maclean in *A River Runs Through It*, I was as a teenager haunted by newspapers. Still, what could I do? I rechecked the look in the guidance counsellor's eyes and said I would be delighted to interview for the job.

Gould—not the well-known New England humorist or the novelist who wrote *The Greenleaf Fires* but a relation of both, I think—greeted me warily but with some interest. We would try each other out, he said, if that suited me.

Now that I was away from the administrative offices of Lisbon High, I felt able to muster a little honesty. I told Mr.

Stephen King

Gould that I didn't know much about sports. Gould said, "These are games people understand when they're watching them drunk in bars. You'll learn if you try."

He gave me a huge roll of yellow paper on which to type my copy—I think I still have it somewhere—and promised me a wage of half a cent a word. It was the first time someone had promised me wages for writing.

The first two pieces I turned in had to do with a basketball game in which an LHS player broke the school scoring record. One was a straight piece of reporting. The other was a sidebar about Robert Ransom's record-breaking performance. I brought both to Gould the day after the game so he'd have them for Friday, which was when the paper came out. He read the game piece, made two minor corrections, and spiked it. Then he started in on the feature piece with a large black pen.

I took my fair share of English Lit classes in my two remaining years at Lisbon, and my fair share of composition, fiction, and poetry classes in college, but John Gould taught me more than any of them, and in no more than ten minutes. I wish I still had the piece—it deserves to be framed, editorial corrections and all—but I can remember pretty well how it went and how it looked after Gould had combed through it with that black pen of his. Here's an example:

199

Last night, in the ~~well-loved~~ gymnasium of Lisbon High School, partisans and Jay Hills fans alike were stunned by an athletic performance unequalled in school history. Bob Ransom, ~~known as "Baloo" Bob~~ ~~for both his size and accuracy~~, scored thirty-seven points. Yes, you heard me right. ~~Plus~~ he did it with grace, speed . . . and with an odd courtesy as well,

Stephen King

I believe he would have subscribed to the notion), more will want to do the former than the latter.

“Vigorous writing is concise.

A sentence should contain no unnecessary words, a paragraph no unnecessary sentences, for the same reason that a drawing should have no unnecessary lines and a machine no unnecessary parts. This requires not that a writer make all his sentences short, or that he avoid all detail and treat his subjects only in outline, but that every word tell.”

—Willian Strunk Jr. *The Elements of Style*

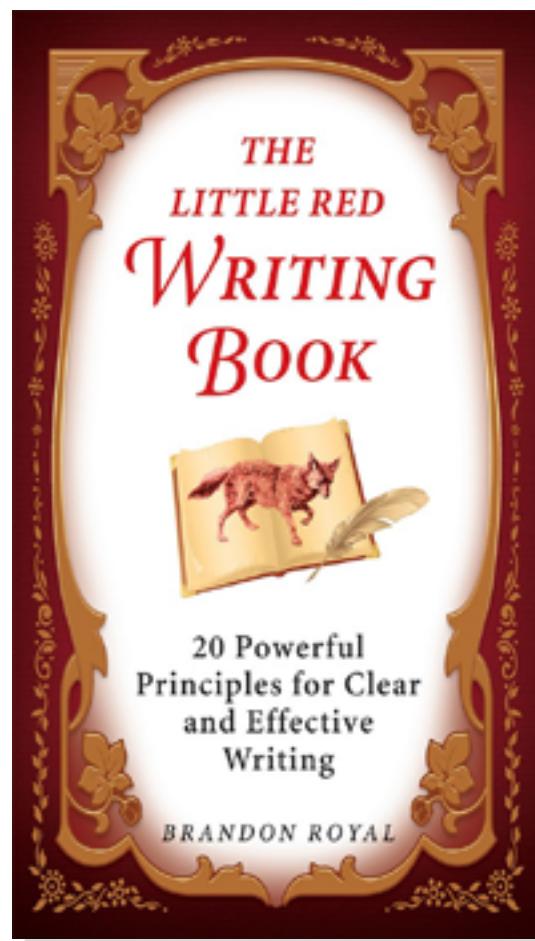
The little red writing book

Brandon Royal

“I have made [this letter] longer than usual because I have not had time to make it shorter.”

—Blaise Pascal, *Lettres Provinciales*

Writing well consists in writing as little as possible. If your interface requires you to explain something with a block of text, the battle is already lost. What writing there is should be concise and effective. To achieve this, you need to think about your text: you need to design it.





Principle 1

Write With a Top-Down Approach



Principle #1: Write your conclusion and place it first.

Writing done for everyday purposes falls into the category of expository writing, which includes newspaper articles, college essays, and business memos and letters. Expository writing explains and often summarizes a topic or issue. Strategically, the summary or conclusion should come at the beginning of an expository piece, not at the end. The reader is first told what the writing is about, then given the supporting facts or details. This way, the reader is not left guessing at the writer's main idea.

Whereas the primary purpose of expository writing is to explain, inform, or persuade, the primary purpose of fiction or creative writing is to enlighten or entertain. As far as fiction and creative writing are concerned, it is fine (even desirable) to delay the conclusion, as in the case of a surprise ending. But the hard-and-fast rule in expository writing is that we should not keep our conclusion from the reader. We should come out with it right away. When our purpose is to explain or inform, don't play, "I've got a secret."

Experienced writing instructors know that one of the easiest ways to fix students' writing is to have them place their conclusions near the top of the page, not the bottom. Instructors are fond of a trick that involves asking students to write a short piece on a random topic and, upon completion, walking up to each student

without reading what he or she has written, circling the last sentence, and moving it to the very top of the page. In a majority of cases, instructors know that the last lines written contain the conclusion. This technique is known as BLOT, or “bottom line on top.” It is human nature, and it seems logical, that we should conclude at the end rather than the beginning. But writing should be top-down, structured in the inverted pyramid style. The broad base of the inverted pyramid is analogous to the broad conclusion set forth at the beginning of a piece.

Favor the top-down approach to writing:

Most Important®

Next Most Important

Next Most Important

Least Important

Avoid the bottom-up approach to writing:

Least Important

Next Most Important

Next Most Important

Most Important®

The newspaper industry depends upon the top-down technique of writing. Reporters know that if their stories cannot fit into the allocated space, their editors will cut from the bottom up. Therefore, conclusions generally cannot appear in the last lines, which are reserved for minor details.

Errors of writing often mimic errors in conversation. When we write, we should think about giving the reader a destination first before giving him or her the directions on how to get there. If we fail to do this, we will not get our message across in the most effective way. The value of a top-down approach in real life conversation occurs in the following dialogue.

POOR VERSION

Dialogue between two coworkers:

"Alice, can you do something for me when you're downtown? If you're taking the subway to Main Street, get off and take the first exit out of the subway and walk down to Cross Street. At the intersection of Cross Street and Vine, you'll find Sandy's Stationery Store. Can you go in and pick up a pack of Pentel 0.5mm lead refills?"

BETTER VERSION

Dialogue between two coworkers:

Alice, can you do something for me when you're downtown? I need a pack of Pentel 0.5mm lead refills. The best place to get them is Sandy's Stationery Store. You can take the subway to Main Street, get off and take the first exit out of the subway and walk down to Cross Street. The store is at the intersection of Cross Street and Vine.

The conclusion is underlined in each version. Note how annoying the first version can be from the listener's perspective. If you have encountered a similar situation in everyday life, you may have felt like screaming. Once you finally find out what the speaker's point

is, you might have to ask him or her to repeat everything so you can remember the details. The same holds true for writing. It is just as frustrating when you are reading a piece of writing and you do not know where the discussion is going.

207

Conceptually, we want to think in terms of a descending writing structure — one in which we move “downhill” from conclusion to details rather than “uphill” from details to conclusion.

Compare the following two versions of the same piece of business writing. In evaluating the two samples, we find that the second one is more top-down in its approach. The conclusion is at the top: “Asia and Africa represent the biggest future international market for basic consumer goods if population is used as a measure.” Also, the second version uses statistics solely as detail.

LESS EFFECTIVE

Three-fourths of the world's people currently live in Asia and Africa — from South Africa to the Sahara, from the Middle East to Japan, from Siberia to Indonesia. This population statistic is quite revealing. If we selectively and representatively choose four persons from the entire world, here is what the probable outcome would be. One person would be from China, one would be from India, and one more would be from somewhere else in Asia or Africa. The fourth person would have to be chosen from all of North America, South America, Europe, and Oceania!

Basic consumer goods represent durable and nondurable daily necessities, including food and cooking utensils, clothing and textiles, toiletries, electronics, home furnishings, and mechanized and miscellaneous household products. Hence, Asia and Africa represent the biggest future international market for basic consumer goods if population is used as a measure.

MORE EFFECTIVE

Asia and Africa represent the biggest future international market for basic consumer goods if population is used as a measure. Basic consumer goods represent durable and nondurable daily necessities, including food

and cooking utensils, clothing and textiles, toiletries, electronics, home furnishings, and mechanized and miscellaneous household products.

Three-fourths of the world's people currently live in Asia and Africa – from South Africa to the Sahara, from the Middle East to Japan, from Siberia to Indonesia. This population statistic is quite revealing. If we selectively and representatively choose four persons from the entire world, here is what the probable outcome would be. One person would be from China, one would be from India, and one more would be from somewhere else in Asia or Africa. The fourth person would have to be chosen from all of North America, South America, Europe, and Oceania!

Now review this piece:

Hundreds of people packed into the auditorium seats on the evening of December 29. Being one of twelve opening performers, I was granted the opportunity to dance on stage for the first time in my life. Although my part only lasted five minutes, those five minutes became a significant moment in my life. Ever since rehearsals began two months before, I had spent many hours practicing on my own, in addition to the normal rehearsal sessions. Whether on a bus, waiting in a doctor's office, or walking to work, I always had my MP3 player on, listening to the music and trying to go through the steps in my mind over and over again. I was determined to do my best. Despite my best preparation, my nervousness caused me to slip during the performance. All of a sudden, my mind turned blank. I stood there, not knowing how to react to the music. Fifteen seconds seemed like 15 hours in a normal day.

The conclusion as underlined above is either well placed or ill placed depending on the writer's purpose in writing the passage. If the purpose is to inform the reader, then it is ill placed because the conclusion should be placed nearer the top. But as this is likely a creative writing piece, meant to entertain, the conclusion can be delayed. Just remember the rule of expository writing that governs everyday writing: Your conclusion should be at or very near the beginning of your written piece.

An airline pilot never leaves the runway without having a destination and flight pattern. When our purpose in writing is to explain or inform, we should conclude first then concentrate on supporting details. Don't play. "I've got a secret."



Principle 2

Break Things Down



Principle #2: Break your subject into two to four major parts and use a lead sentence.

Assuming that you know what you want to write about, you must decide what basic building blocks will comprise your work. You can break your subject into two to four major parts. Three parts are typically recommended, but for the sake of simplicity, no more than four categories should be introduced. The classic “five-paragraph” approach to writing can be used to outline, in one paragraph, any writing piece. In the upcoming example, all you have to do is supply the colors!

LEAD SENTENCES VS. TOPIC SENTENCES

Once you have broken down your topic into two to four major categories, next you will want to elaborate on these ideas. Consider using a *lead sentence*, which is similar to a *topic sentence*. Whereas a topic sentence summarizes the contents of a single



paragraph within an essay or report, a lead sentence summarizes the contents of an entire essay or report. Placed at the beginning of a piece, it foreshadows what is to come, highlighting what items will be discussed and, typically, the order in which they will be discussed. Each item in the lead should be developed into at least one separate paragraph within the body of the essay or report. For example, in a personal essay, this sentence could serve as an introduction or lead:

I would like to show who I am through a discussion of three special turning points in my personal and career development: when I went to university on a lacrosse scholarship, when I spent a year with the Peace Corps, and when I joined a commodity trading firm in London.

In a business report, the following could serve as a lead sentence, placed at the beginning of a report:

Based on information taken from a recent survey, this report summarizes the three biggest problems that our company faces: namely, employee turnover, store thefts, and poor customer service.

The number three is a magic number in writing. Think of building your writing around three key ideas or concepts.

The number three is a magic number in writing. Think of building your writing around three key ideas or concepts.



Principle 3

Use Transition Words



Principle #3: Use transition words to signal the flow of your writing.

"Transition" words, such as "but" and "however," have been called the traffic lights of language. They serve one of four primary purposes: to show contrast, illustration, continuation, or conclusion. On the next page, you will see transition words highlighted in two sample paragraphs. Words of illustration include "first," "second," "for instance," and "for example." "So" signals conclusion. "However" signals contrast. "Moreover" signals continuation.

Transition words appear underlined in the following examples.

EXAMPLE 1

Time management involves thinking in terms of effectiveness first and efficiency second. Whereas efficiency is concerned with doing a task in the

fastest possible manner, effectiveness is concerned with spending time doing the "right" things. Effectiveness is therefore a broader, more useful concept, which questions whether we should even do a particular task.

212

EXAMPLE 2

The process of evolution takes two distinct forms: organic and exosomatic. In the first, which is commonly called Darwinian evolution, a plant or animal develops a genetic mutation that may be either helpful or harmful. If the change is helpful, the organism is favored by the process of natural selection and flourishes; if it is harmful, the organism suffers and eventually dies out.

The whole of what we call human culture, on the other hand, is a result of exosomatic evolution. Such a change may be gradual, but it represents conscious choices that enable human beings to adapt to environments that would otherwise be inimical to their survival.

Put it before them briefly so they
will read it, clearly so they will
appreciate it, picturesquely so they
will remember it and, above all,
accurately, so they will be guided
by its light.

—Joseph Pulitzer

THE FOUR TYPES OF TRANSITION WORDS

213

I. Continuation Words

GREEN LIGHT

"Keep going in the same direction"

Examples:

- moreover • furthermore
- on the one hand
- undoubtedly
- coincidentally

II. Illustration Words

FLASHING GREEN

"Slow down and be watchful"

Examples:

- first, second, third
- for example • for instance
- in fact
- case in point



III. Contrast Words

FLASHING YELLOW

"Get ready to turn"

Examples:

- however
- but • yet
- on the other hand
- whereas
- conversely

IV. Conclusion Words

RED LIGHT

"You're about to arrive"

Examples:

- in conclusion
- finally • clearly
- hence • so • thus
- therefore
- as a result



Principle 5

Keep Like Things Together



Principle #5: Finish discussing one topic before going on to discuss other topics.

Imagine visiting the zoo to find that all the animals were in one big cage. It would not only be dangerous for the animals but also nearly impossible for visitors to view the animals in a coherent manner. Unfortunately, sometimes a piece of writing can be like a zoo, in which all of the different animals (ideas) are in one big cage, running wild. When we write (as when we speak), the ideas we describe should be grouped together. It is best to finish discussing one idea before going on to discuss another.

Here's an example of an essay with jumbled ideas.

ORIGINAL VERSION

In 1981, Roger Sperry received the Nobel Prize for his proof of the split-brain theory. According to Dr. Sperry, the brain has two hemispheres with different, but overlapping functions.

The left side of the brain is responsible for analytical, linear, verbal, and rational thought. Left-brain thinking is "spotlight" thinking. The right hemisphere is holistic, imaginative, nonverbal, and artistic. It is the left brain that a person relies on when balancing a checkbook,

remembering names and dates, or setting goals and objectives. Whenever a person recalls another person's face, becomes engrossed in a symphony, or simply daydreams, that person is engaging in right-brain functions. Right-brain thinking is "floodlight" thinking and right-brain processes are, to the chagrin of many, less often rewarded in school. Since most of the Western concepts of thinking come from Greek logic, which is a linear logic system, left-brained processes are most rewarded in the western educational system.

In summary, the right and left hemispheres of the brain each specialize in distinct types of thinking processes. In the most basic sense, the left brain is the analytical side while the right brain is the creative side.

Note that although the above writing piece employs a classic structure — containing an introduction, body, and conclusion — the content is difficult to read and absorb because ideas are tangled. If this discussion were to continue for a couple of pages, the reader might feel that his or her mind had turned to spaghetti. We know that there are two things under discussion — left-brain versus right-brain thinking — but the technique with which ideas are described and supported is deficient.

CORRECTED VERSION 1

In 1981, Roger Sperry received the Nobel Prize for his proof of the split-brain theory. According to Dr. Perry, the brain has two hemispheres with different but overlapping functions. Each hemisphere of the brain specializes in distinct types of thinking processes. In the most basic sense, the left brain is the analytical side while the right brain is the creative side.

The left side of the brain is responsible for analytical, linear, verbal, and rational thought. Left-brain thinking is characterized as "spotlight" thinking. It is the left brain that a person relies on when balancing a checkbook, remembering names and dates, or setting goals and objectives. The right hemisphere is holistic, imaginative, nonverbal, and artistic. Right-brain thinking is characterized as "floodlight" thinking. Whenever a person recalls another person's face, becomes engrossed in a symphony, or simply daydreams, that person is engaged in right-brain functions.

Since most Western concepts of thinking are derived from Greek

logic, which is a linear logic system, left-brained processes are most rewarded in the Western education system. Right-brain processes are, to the chagrin of many, less often rewarded in school.

In the corrected example above, we also have classic usage of introduction, body, and conclusion. The structure in the second paragraph proceeds as follows: left-brain thinking is described within the first two sentences, followed by a third supporting sentence which includes examples of left-brain thinking. Right-brain thinking is then described in two sentences, followed by a supporting sentence which includes examples of right-brain thinking. The third paragraph concludes with an implication of left- and right-brain thinking.

CORRECTED VERSION 2

In 1981, Roger Sperry received the Nobel Prize for his proof of the split-brain theory. According to Dr. Perry, the brain has two hemispheres with different but overlapping functions. Each hemisphere of the brain specializes in distinct types of thinking processes. In the most basic sense, the left brain is the analytical side while the right brain is the creative side.

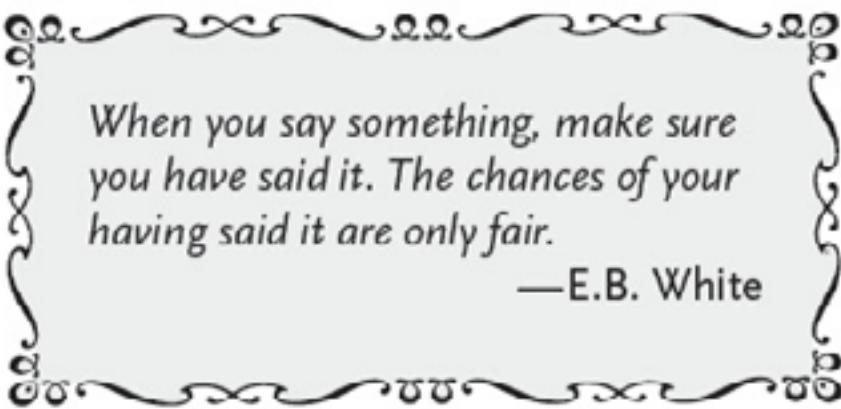
The left side of the brain is responsible for analytical, linear, verbal, and rational thought. Left-brain thinking is characterized as "spotlight" thinking. It is the left brain that a person relies on when balancing a checkbook, remembering names and dates, or setting goals and objectives. Since most Western concepts of thinking are derived from Greek logic, which is a linear logic system, left-brained processes are most rewarded in the Western education system.

The right hemisphere is holistic, imaginative, nonverbal, and artistic. Right-brain thinking is characterized as "floodlight" thinking. Whenever a person recalls another person's face, becomes engrossed in a symphony, or simply daydreams, that person is engaged in right-brain functions. Right-brain processes are, to the chagrin of many, less often rewarded in school.

The three-paragraph structure above is also a classic one: an introduction is followed by two paragraphs, each dedicated entirely to either left- or right-brain thinking. In the second

paragraph, left-brain thinking is described within the first two sentences, followed by a third supporting sentence containing examples of this type of thinking, and a concluding sentence highlighting an implication of left-brain thinking. In the third paragraph, right-brain thinking is described in two sentences, followed by a supporting sentence with examples of this type of thinking, and finally a one sentence implication of right-brain thinking.

217



*When you say something, make sure
you have said it. The chances of your
having said it are only fair.*

—E.B. White



Principle 6

Support What You Say



Principle #6: Use specific and concrete words to support what you say.

One major difference between good writing and mediocre writing lies with the specific and concrete examples that you use or fail to use. Say, for example, you are writing about an apple. Not all apples are identical. What kind of apple is it? Golden Delicious, Gala, Fuji, McIntosh, Granny Smith? What color is it? What shape is it? How does it taste? What is its texture? Where is it grown? Let's look at an example in a business context. Suppose you hear that your company's profits are down. What are the specifics? Did the sales volume decline? Was the sales price reduced? Did costs go up? And, if any of the above, then by how much?

Note the difference in each of the following statements:

GENERAL

Corporate profits decreased.

BETTER BUT STILL NOT SPECIFIC

Corporate profits decreased because costs increased.

SPECIFIC

Corporate profits decreased by 10 percent as overall costs increased by 20 percent.

219

EVEN BETTER

Corporate profits decreased by 10 percent as overall costs increased by 20 percent. In particular, higher salary expenses were the major reason for the increase in costs. Higher salary costs were primarily the result of increases in executive compensation; the aggregate wages paid to factory workers actually decreased by 5 percent due to a decrease in the number of overtime hours clocked.

Examples and details are the very things people remember long after reading a piece. Compare the two examples below describing the popular attitude toward science.

VERSION 1

The popular attitude toward science in the United States is a mix of superstition and awe. Quaint folklore portrays scientific genius as solitary and requiring no nurture. Within the public imagination, such pleasant thoughts go undisturbed by the reality of today's large research labs.

VERSION 2

The popular attitude toward science in the United States is a mix of superstition and awe. Quaint folklore portrays scientific genius as solitary and requiring no nurture. Within the public imagination are visions of the Wright Brothers at work in their bicycle shop, contriving the first flying machine, and of Thomas Edison plumbing the mysteries of electricity with a few magnets and some pieces of wire. Such pleasant thoughts go undisturbed by today's large research labs, whose members undergo highly specialized training in order to work on narrowly defined research problems.



The second version uses examples drawn from the Wright Brothers and Thomas Edison. This helps us visualize what the author is saying.

Consider the two memos below. Which one would convince you to attend the Calgary Stampede and Exhibition?

MEMO 1

The Calgary Stampede will be held during the first week of July. There will be loads of activities, fun, and food for all. Bring your cowboy hat and boots. See you there!

MEMO 2

The Calgary Stampede will be held during the first week of July. The exhibition grounds are home to two dozen midway rides, a myriad of food stalls (try those miniature doughnuts!), the sounds of live country music, First Nations exhibits, bustling saloons, and a large casino. For the youngsters, there is a petting zoo, magic tricks, and loads of games, with the chance to win giant stuffed animals. The opening day parade has a flotilla of floats, and daily rodeo events including calf roping, bull riding, and chuck wagon races. Fantastic fireworks each evening. See you there!

Note that the second and better example is longer than the original. Given that writing should be concise, why is the shorter example not better? A trade-off exists between brevity and detail. Sufficient detail will make a piece of writing longer, but this does not necessarily indicate wordiness. Conciseness requires a minimum number of words at the sentence level, whereas sufficient support may require more sentences.



221

Principle 17

Capitalize on Layout and Design



Principle #17: Add more space around your writing to increase readability.

The easiest way to make writing more readable is to increase your document's margin. Also, ensuring that a blank line separates paragraphs will let your composition breathe. Avoid writing one big block of words pressed tight against the edges of the page. Two versions of an identical document appear on the next two pages; the second is easier to read simply because it employs more space in the margins and between paragraphs.



Paradox of Our Time by Dr. Bob Moorehead

The paradox of our time in history is that we have taller buildings but shorter tempers; wider freeways, but narrower viewpoints. We spend more, but have less; we buy more, but enjoy less. We have bigger houses and smaller families; more conveniences, but less time. We have more degrees but less sense; more knowledge, but less judgment; more experts, yet more problems; more medicine, but less wellness.

We drink too much, smoke too much, spend too recklessly, laugh too little, drive too fast, get angry too quickly, stay up too late, get up too tired, read too little, watch TV too much, and pray too seldom.

We have multiplied our possessions, but reduced our values. We talk too much, love too seldom, and hate too often.

We've learned how to make a living, but not a life. We've added years to life, not life to years. We've been all the way to the moon and back, but have trouble crossing the street to meet a new neighbor. We conquered outer space, but not inner space. We've done larger things, but not better things.

We've cleaned up the air, but polluted the soul. We've conquered the atom, but not our prejudice. We write more, but learn less. We plan more, but accomplish less. We've learned to rush, but not to wait. We build more computers to hold more information, to produce more copies than ever, but we communicate less and less.

These are the times of fast foods and slow digestion; of tall men and short character; of steep profits and shallow relationships. These are the days of two incomes but more divorce; of fancier houses, but broken homes. These are days of quick trips, disposable diapers, throwaway morality, overweight bodies, and pills that do everything from cheer, to quiet, to kill. It is a time when there is much in the showroom window and nothing in the stockroom..

Paradox of Our Time by Dr. Bob Moorehead

The paradox of our time in history is that we have taller buildings but shorter tempers; wider freeways, but narrower viewpoints. We spend more, but have less; we buy more, but enjoy less. We have bigger houses and smaller families; more conveniences, but less time. We have more degrees but less sense; more knowledge, but less judgment; more experts, yet more problems; more medicine, but less wellness.

We drink too much, smoke too much, spend too recklessly, laugh too little, drive too fast, get angry too quickly, stay up too late, get up too tired, read too little, watch TV too much, and pray too seldom.

We have multiplied our possessions, but reduced our values. We talk too much, love too seldom, and hate too often.

We've learned how to make a living, but not a life. We've added years to life, not life to years. We've been all the way to the moon and back, but have trouble crossing the street to meet a new neighbor. We conquered outer space, but not inner space. We've done larger things, but not better things.

We've cleaned up the air, but polluted the soul. We've conquered the atom, but not our prejudice. We write more, but learn less. We plan more, but accomplish less. We've learned to rush, but not to wait. We build more computers to hold more information, to produce more copies than ever, but we communicate less and less.

These are the times of fast foods and slow digestion; of tall men and short character; of steep profits and shallow relationships. These are the days of two incomes but more divorce; of fancier houses, but broken homes. These are days of quick trips, disposable diapers, throwaway morality, overweight bodies, and pills that do everything from cheer, to quiet, to kill. It is a time when there is much in the showroom window and nothing in the stockroom.



Principle 18

Employ Readability Tools



Principle #18: Make key words and phrases stand out.

Painters, musicians, and poets are but a few individuals highly adept at judging what effect stylistic additions and deductions will have on an overall composition. Writing is also a balancing act. The writer seeks to retain those greater elements that most define a writing piece while looking for smaller adornments to bolster its appearance and readability. Such adornments might include boldface type, italics, dashes, bullets, enumerations, and shading.

BOLDS

Bolds (boldface type) may be used to emphasize keywords and help key ideas “jump out” at the reader. Bolds are especially useful for flyers, résumés, and other documents in which the reader may spend only a brief time reviewing. Italics or underlining can do the same job as bold type, though care must be exercised not to overdo it. For example, rarely do we want to see bolds and italics used in the same paragraph. One unwritten rule of writing and editing is to never use bolds, italics, and underlines together (ditto for bolds, italics, and full caps in combination). Be aware that if you use boldface type too liberally, you will dull the effect and perhaps patronize the reader.

ITALICS

There is artistry in the occasional use of italics. Italics, like bolds, serve similar purposes. Consider using italics to highlight certain key words, especially those that show contrast, or for small words, especially negative words such as not, no, and but. Be careful of overusing italics because they are tiring on the eye and can make the page look busy.

225

For stylistic purposes, many examples in this book appear in italics in order to distinguish them from explanatory text. Note that such text would not normally be italicized.

DASHES

Dashes can be used to vary the rhythm of a sentence and to place emphasis on words and phrases in a more dynamic manner than could be achieved through the use of a comma (or pair of commas) or semicolon.

The world's oldest dated book — the Diamond Sutra — is an elaborately decorated book containing a beautiful cover piece. Written in 868 A.D., it was discovered in 1907 in a cave near Dunhuang, China.

Note how the dashes in the previous example make the book appear more dramatic than in the example below, which contains commas:

The world's oldest dated book, the Diamond Sutra, is an elaborately decorated book containing a beautiful cover piece. Written in 868 A.D., it was discovered in 1907 in a cave near Dunhuang, China.

BULLETS

Bullets (•) are effective tools for paraphrasing information, especially for presenting information in short phrases when formal sentences are not required. Bullets are most commonly

used when preparing résumés, slides, or flyers. Bullets are not, however, recommended for use in the main body of an essay or report unless they are included within a table. It is also not considered good practice in formal writing to use hyphens (-) or asterisks (*) in place of bullets.

ENUMERATIONS

Enumerations involve the numbering of points. Listing items by number is more formal but very useful for ordering ideas or data.

EXAMPLE 1

I feel that my greatest long-term contributions working in this field will be measured by (1) my ability to find ways to define and quantify, in dollars and cents, the benefits of ethics and corporate citizenship, and (2) my ability to sell corporations on the proactive benefits of these programs as a means to market their company, products, and employees.

EXAMPLE 2

Each letter in the word “s.u.c.c.e.s.s.” embodies a single action:

1. Super effort
2. Unusual drive
3. Copy what works.
4. Change what doesn't.
5. Exercise now and cut out excess.
6. Save a little more, spend a little less.

7. *Start all over again the very next day.*

SHADING

227

Shading creates contrast on the page and can be a great device for formatting business reports. For example, you can highlight the start of each section of a report by using shaded section headings. Within the pages of a report, shading is often used to tint the top row (header row) of a table. Flyers also commonly use shading to call out information.

RÉSUMÉ EXAMPLE

The following is an excerpt from a résumé. Résumés provide a classic example of the use of readability tools, particularly in terms of bullets, bolds, and italics.

PROFESSIONAL EXPERIENCE

2010–present **BANK OF AMERICA**, Hartford, Conn.

Financial Analyst

- Analyzed branch performance and devised new strategies to improve regional market share. Formulated a two-year marketing plan for two branches.
- Developed a new commission system and assisted in its implementation.
- Presented tax saving strategies and advice on investment portfolio compositions for principal clients.



Principle 20

Go Back and Rework Your Writing



Principle #20: Wait until your writing stands still before you call it finished.

Rare is the writer who can sit down and knock out a perfect writing draft without corrections. Most proficient writers take at least three drafts to finish short pieces of writing. For example, you may be writing a cover letter to accompany your updated résumé. First, you write to get your ideas down on paper. Second, you edit through what you have written, add detail, make connections, and make corrections. Third, you wait twenty-four hours and reread, making minor changes. The longer the work, the more times this process is repeated for individual sections.

The number of drafts required for an entire work depends on the work's length and complexity. A two-line office memo is likely to be done in a single draft because it is short and simple. A one-page poem might take more than a dozen drafts because it is longer and more difficult.

WHEN IS IT REALLY FINISHED?

Making changes to your writing is annoying and grueling. But eventually, with changes made, you will likely be satisfied with what you have written and not want to add or delete anything. This is the point at which your writing is finished — your writing

is “standing still.” Unpolished writing is like shifting sand in a desert storm. Eventually the storm ceases, and the sand sits still.

The word “finished,” when referring to writing, should really be enclosed in quotation marks because writing is never actually finished. With respect to writing done for everyday purposes, completion is an end in itself. However, for more permanent written works, such as novels, writing can be continued indefinitely because it can always be improved. Even published books can be reworked and reedited. Weeks, months, and years after a book is published, an author will invariably contemplate changes.

APPRECIATE THE PROCESS

Writing is a creative process. You discover things as you force yourself to write. What is especially satisfying is turning “junk” writing into something worthwhile. When you put together a lengthy piece, such as a personal essay or business report, you will naturally begin by writing some areas well. Other areas you’ll not be satisfied with, and those must be reworked.

Let’s call the parts you like “flowers” and the parts you dislike “dirt.” As you focus your efforts on the “dirt,” you begin to make improvements, and sometimes to your surprise, these areas become as good as, or better than, one or more of the “flowers.” This is extremely satisfying. You are inspired. You gain energy. You now try to improve other “dirt” areas until there are none left. Later, you go back to an original “flowered” area and make it even better, thus raising it up one notch from anything done before. The writing process is a ongoing process of producing flowers and dirt.

Most people hate reworking their writing. It is human nature. The pressure and agony of writing is one reason why alcohol has been humorously dubbed “the occupational hazard of professional writers.” It is not writing per se, but the rewriting and redrafting process that can drive a person to drink. Worse is the reality of knowing that even before you begin to write — no matter how well you write — your writing will require revision. Fortunately, for most students and business professionals, the everyday writing process is not filled with the same emotional highs and lows as it is for a person who makes a living from

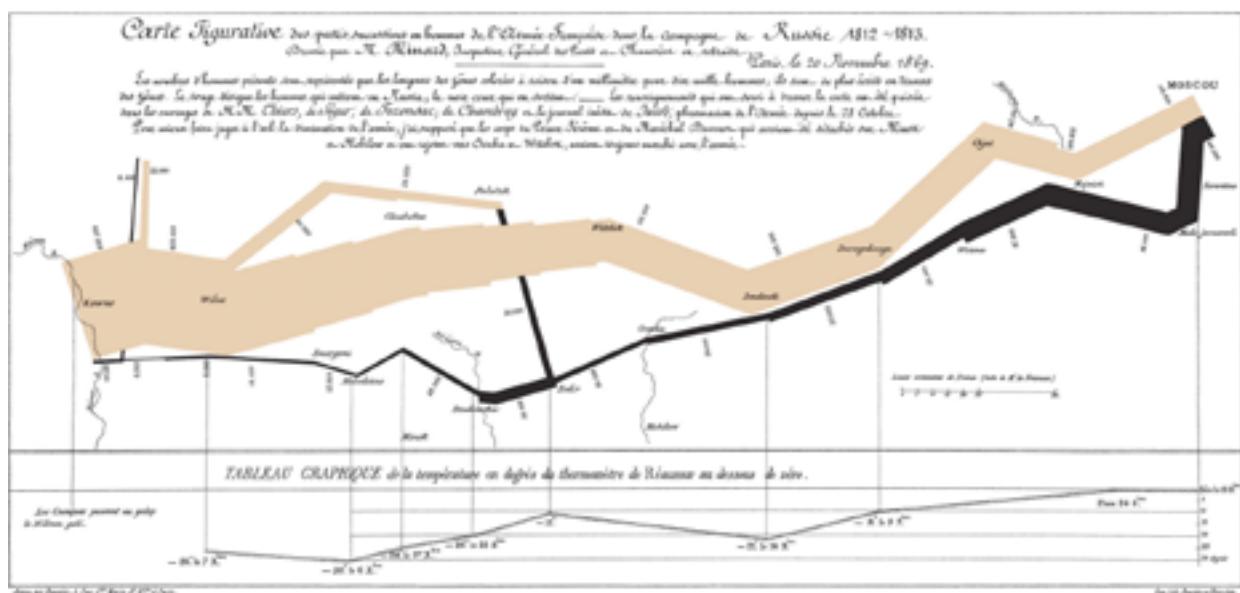


writing.

It is a great feeling to look at something you wrote a long time ago, be it an old college essay, business report, personal letter, or poem, and say to yourself, "Wow, this is funny. Some of this stuff blows me away! How did I come up with it?" There is no absolute answer. Skill, luck, boldness, and naiveté are key ingredients in the writing process.

*The pleasure of the first draft lies
in deceiving yourself that it is quite
close to the real thing. The pleasure
of the subsequent drafts lies partly
in realizing that you haven't been
gulled by the first draft.*

—Julian Barnes



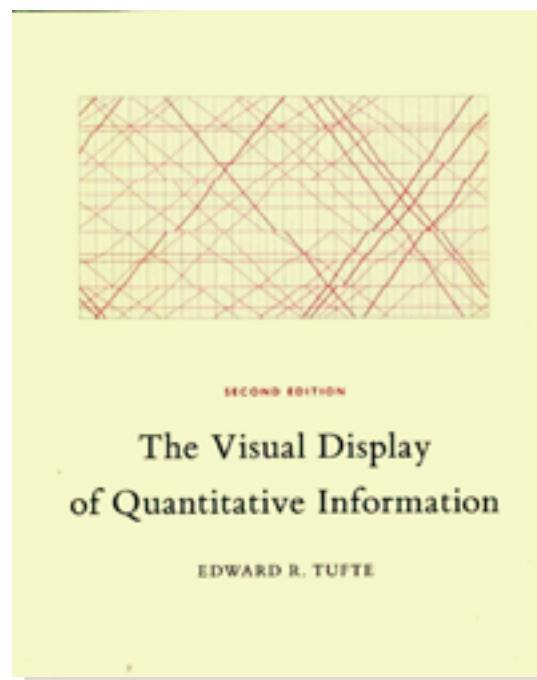
Charles Joseph Minard's 1869 graph of Napoleon's disastrous invasion of Russia. In Tufte's words, it "may well be the best statistical graphic ever drawn."

The visual display of quantitative information

Edward R. Tufte

“What makes for such graphical elegance? What accounts for the quality of Minard’s graphics, of those of Playfair and Marey, and of some recent work, such as the new view of the galaxies? Good design has two key elements:

Graphical elegance is often found in **simplicity of design** and **complexity of data.**”



Everyone spoke of an information overload, but what there was in fact was a non-information overload.

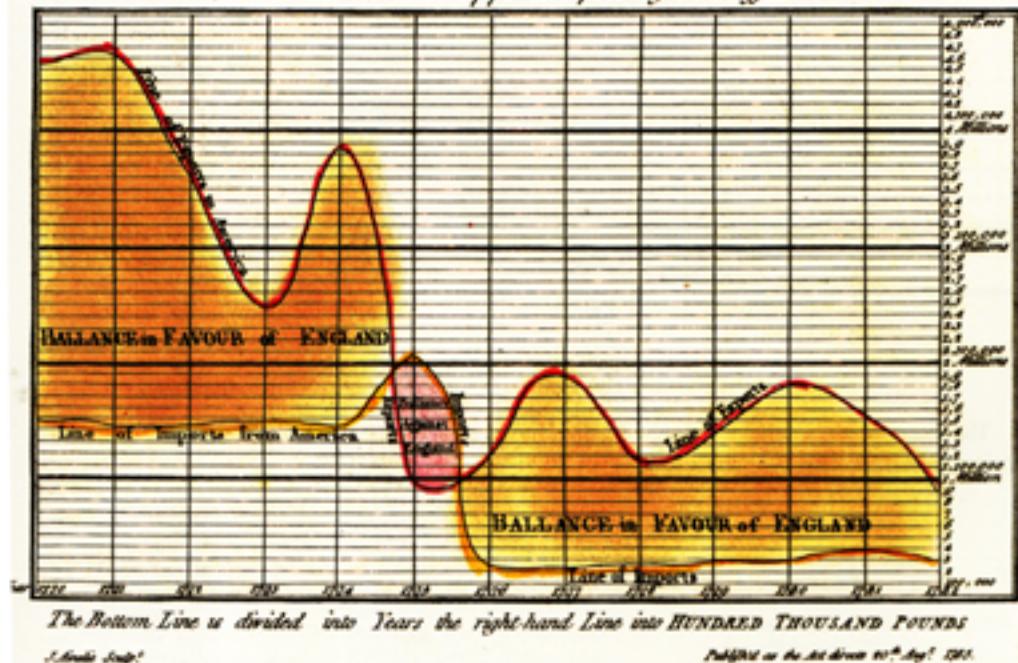
Richard Saul Wurman, *What-If, Could-Be* (Philadelphia, 1976)

4 Data-Ink and Graphical Redesign

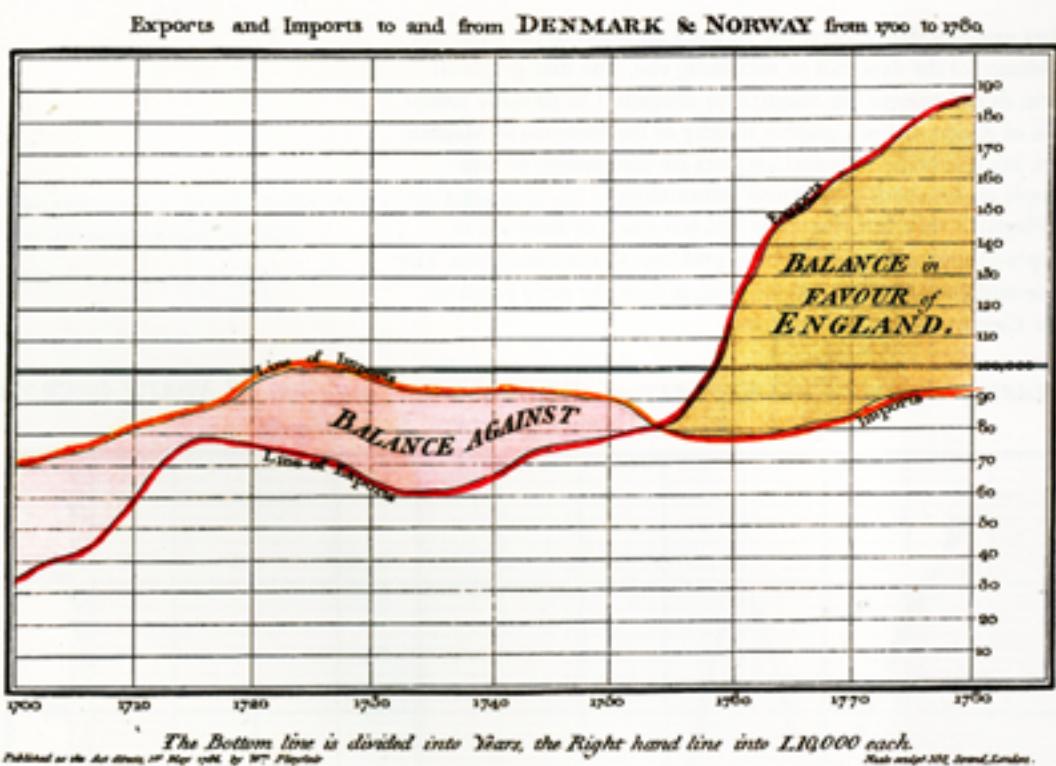
Data graphics should draw the viewer's attention to the sense and substance of the data, not to something else. The data graphical form should present the quantitative contents. Occasionally artfulness of design makes a graphic worthy of the Museum of Modern Art, but essentially statistical graphics are instruments to help people reason about quantitative information.

Playfair's very first charts devoted too much of their ink to graphical apparatus, with elaborate grid lines and detailed labels. This time-series, engraved in August 1785, is from the early pages of *The Commercial and Political Atlas*:

CHART of IMPORTS and EXPORTS of ENGLAND to and from all NORTH AMERICA
From the Year 1770 to 1782 by W. Playfair



Within a year Playfair had eliminated much of the non-data detail in favor of cleaner design that focused attention on the time-series itself. He then began working with a new engraver and was soon producing clear and elegant displays:



This improvement in graphical design illustrates the fundamental principle of good statistical graphics:

Above all else show the data.

The principle is the basis for a theory of data graphics.

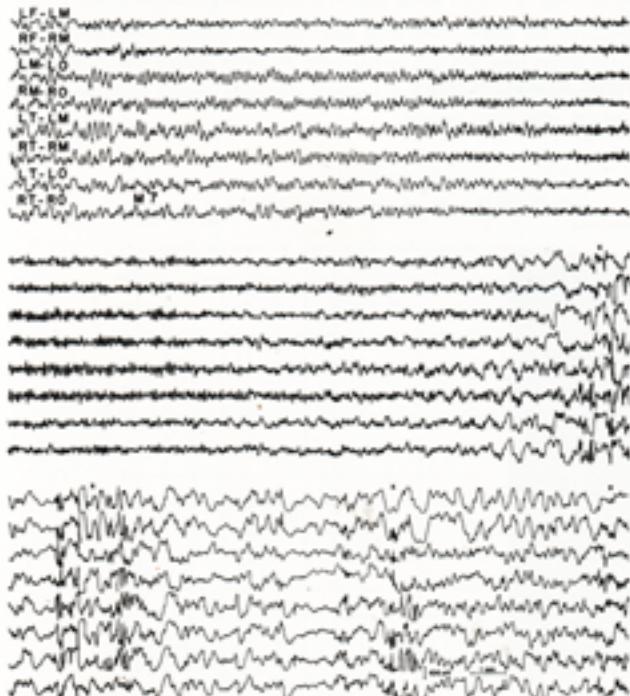
Data-Ink

A large share of ink on a graphic should present data-information, the ink changing as the data change. *Data-ink* is the non-erasable core of a graphic, the non-redundant ink arranged in response to variation in the numbers represented. Then,

data-ink
Data-ink ratio =
$$\frac{\text{data-ink}}{\text{total ink used to print the graphic}}$$

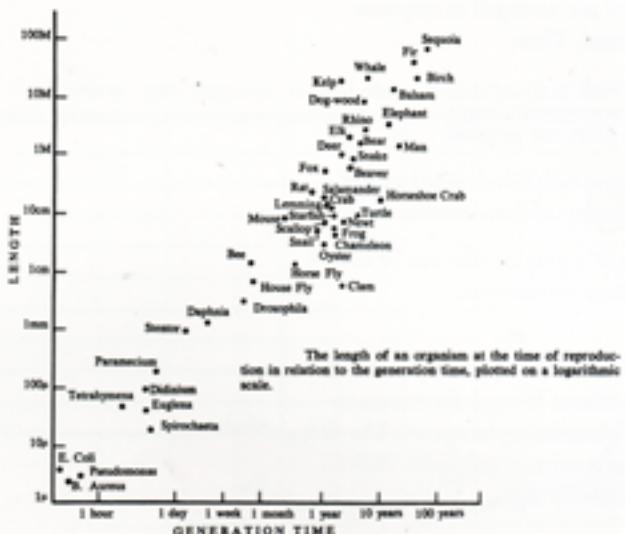
- = proportion of a graphic's ink devoted to the non-redundant display of data-information
- = 1.0 - proportion of a graphic that can be erased without loss of data-information.

A few graphics use every drop of their ink to convey measured quantities. Nothing can be erased without losing information in these continuous eight tracks of an electroencephalogram. The data change from background activity to a series of polyspike bursts. Note the scale in the bottom block, lower right:



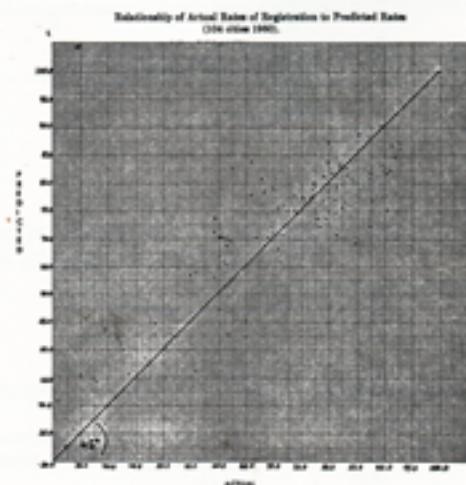
Kenneth A. Kooi, *Fundamentals of Electroencephalography* (New York, 1971), p. 110.

Most of the ink in this graphic is data-ink (the dots and labels on the diagonal), with perhaps 10–20 percent non-data-ink (the grid ticks and the frame):

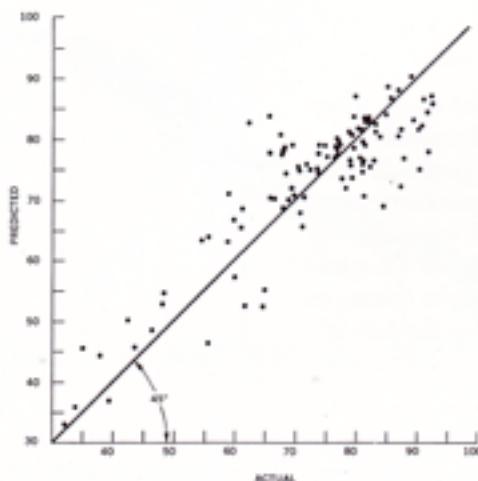


John Tyler Bonner, *Size and Cycle: An Essay on the Structure of Biology* (Princeton, 1965), p. 17.

In this display with nearly all its ink devoted to matters other than data, the grid sea overwhelms the numbers (the faint points scattered about the diagonal):

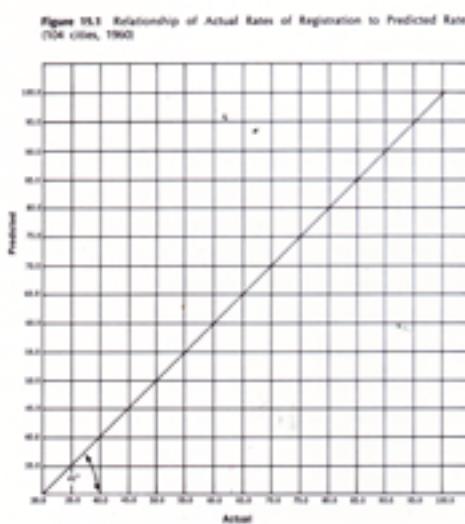


Another published version of the same data drove the share of data-ink up to about 0.7, an improvement:



Relationship of Actual Rates of Registration to Predicted Rates (104 cities 1960).

But a third reprint publication of the same figure forgot to plot the points and simply retraced the grid lines from the original, including the excess strip of grid along the top and right margins. The resulting figure achieves a graphical absolute zero, a null data-ink ratio:



The three graphics were published in, respectively, Stanley Kelley, Jr., Richard E. Ayres, and William G. Bowen, "Registration and Voting: Putting First Things First," *American Political Science Review*, 61 (1967), 371; then reprinted in Edward R. Tufte, ed., *The Quantitative Analysis of Social Problems* (Reading, Mass., 1970), p. 267; and reprinted again in William J. Crotty, ed., *Public Opinion and Politics: A Reader* (New York, 1970), p. 364.

Maximizing the Share of Data-Ink

The larger the share of a graphic's ink devoted to data, the better (other relevant matters being equal):

Maximize the data-ink ratio, within reason.

Every bit of ink on a graphic requires a reason. And nearly always that reason should be that the ink presents new information.

The principle has a great many consequences for graphical editing and design. The principle makes good sense and generates reasonable graphical advice—for perhaps two-thirds of all statistical graphics. For the others, the ratio is ill-defined or is just not appropriate. Most important, however, is that other principles bearing on graphical design follow from the idea of maximizing the share of data-ink.

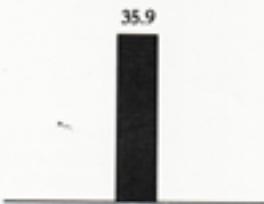
Two Erasing Principles

The other side of increasing the proportion of data-ink is an erasing principle:

Erase non-data-ink, within reason.

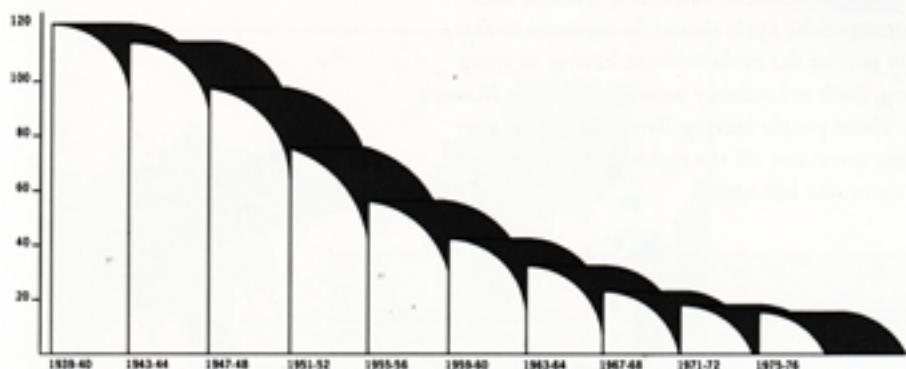
Ink that fails to depict statistical information does not have much interest to the viewer of a graphic; in fact, sometimes such non-data-ink clutters up the data, as in the case of a thick mesh of grid lines. While it is true that this boring ink sometimes helps set the stage for the data action, it is surprising, as we shall see in Chapter 7, how often the data themselves can serve as their own stage.

Redundant data-ink depicts the same number over and over. The labeled, shaded bar of the bar chart, for example,

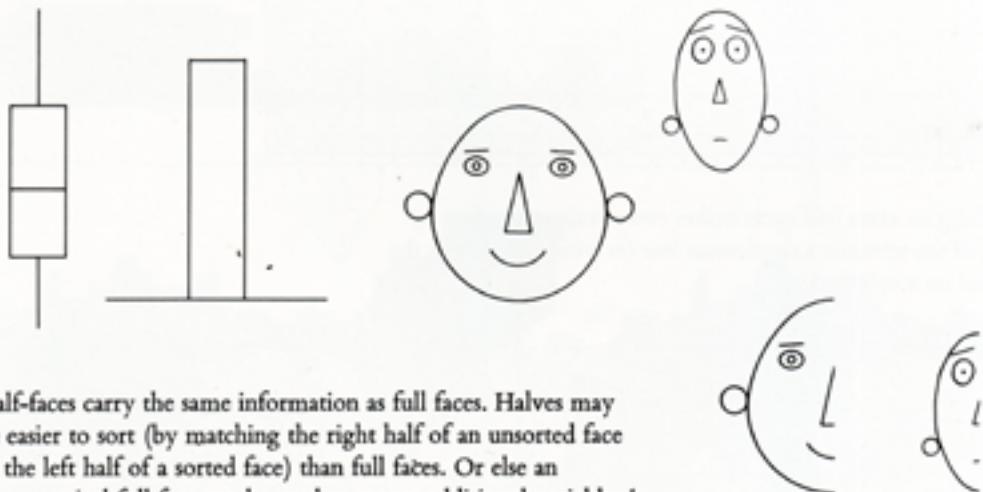


unambiguously locates the altitude in six separate ways (any five of the six can be erased and the sixth will still indicate the height): as the (1) height of the left line, (2) height of shading, (3) height of right line, (4) position of top horizontal line, (5) position (not content) of number at bar's top, and (6) the number itself. That is

more ways than are needed. Gratuitous decoration and reinforcement of the data measures generate much redundant data-ink:



Bilateral symmetry of data measures also creates redundancy, as in the box plot, the open bar, and Chernoff faces:



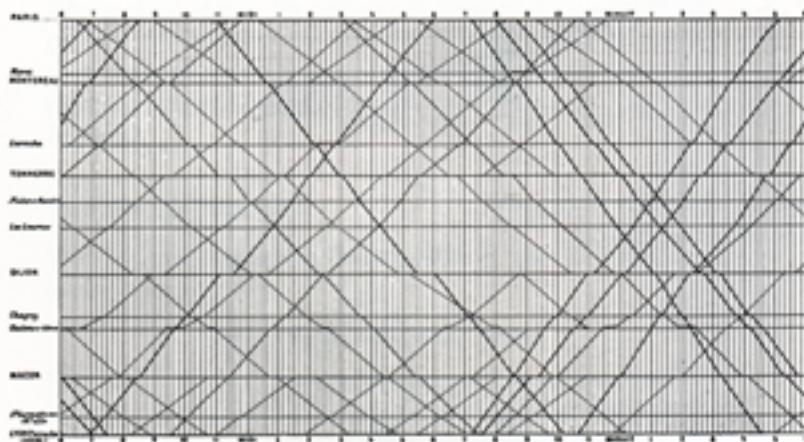
Half-faces carry the same information as full faces. Halves may be easier to sort (by matching the right half of an unsorted face to the left half of a sorted face) than full faces. Or else an asymmetrical full face can be used to report additional variables.¹

Bilateral symmetry doubles the space consumed by the design in a graphic, without adding new information. The few studies done on the perception of symmetrical designs indicate that "when looking at a vase, for instance, a subject would examine one of its symmetric halves, glance at the other half and, seeing that it was identical, cease his explorations. . . . The enjoyment of symmetry . . . lies not with the physical properties of the figure. At least eye movements suggest anything but symmetry, balance, or rest."²

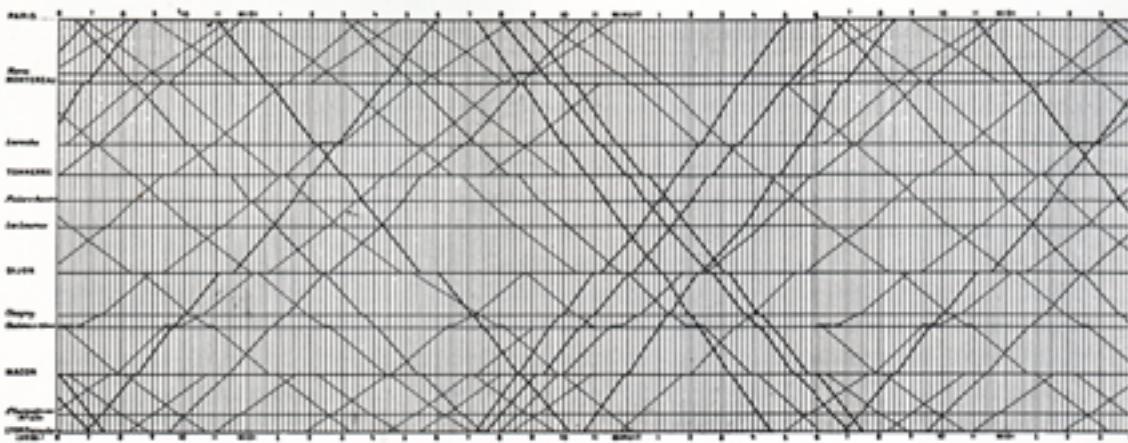
¹ Bernhard Flury and Hans Riedwyl, "Graphical Representation of Multivariate Data by Means of Asymmetrical Faces," *Journal of the American Statistical Association*, 76 (December 1981), 757-765.

² Leonard Zunne, *Visual Perception of Form* (New York, 1970), pp. 256-257.

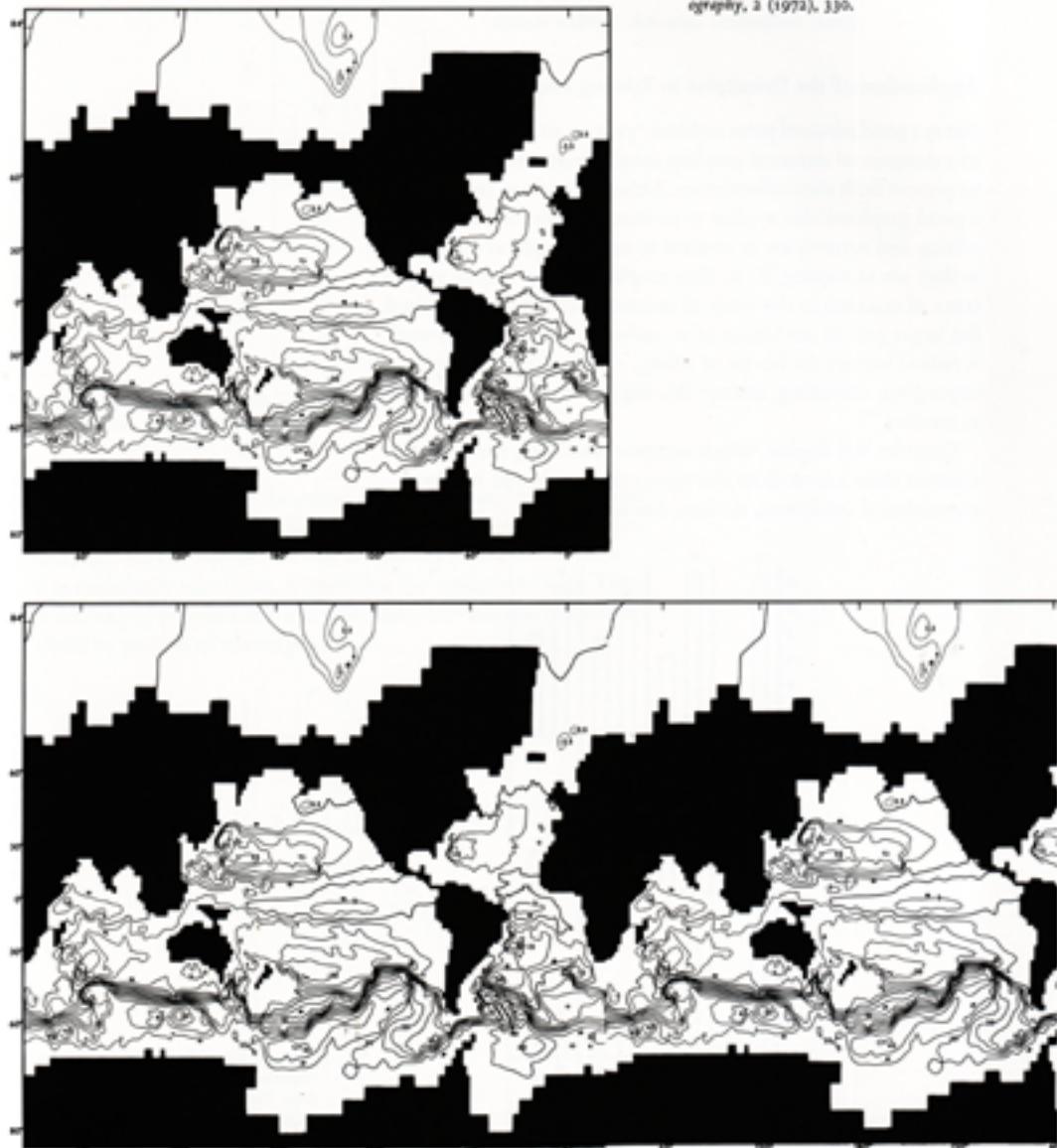
Redundancy, upon occasion, has its uses: giving a context and order to complexity, facilitating comparisons over various parts of the data, perhaps creating an aesthetic balance. In cyclical time-series, for example, parts of the cycle should be repeated so that the eye can track any part of the cycle without having to jump back to the beginning. Such redundancy possibly improves Marey's 1880 train schedule. Those people leaving Paris or Lyon in the evening find that their trains run off the right-hand edge of the chart, to be picked up on the left again:



Attaching an extra half cycle makes every train in the first 24 hours of the schedule a continuous line (as would mounting the original on a cylinder):



And, similarly, instead of once around the world in this display of surface ocean currents, one and two-thirds times around is better:



Kirk Bryan and Michael D. Cox, "The Circulation of the World Ocean: A Numerical Study. Part 1, A Homogeneous Model," *Journal of Physical Oceanography*, 2 (1972), 330.

Most data representations, however, are of a single, uncomplicated number, and little graphical repetition is needed. Unless redundancy has a distinctly worthy purpose, the second erasing principle applies:

Erase redundant data-ink, within reason.

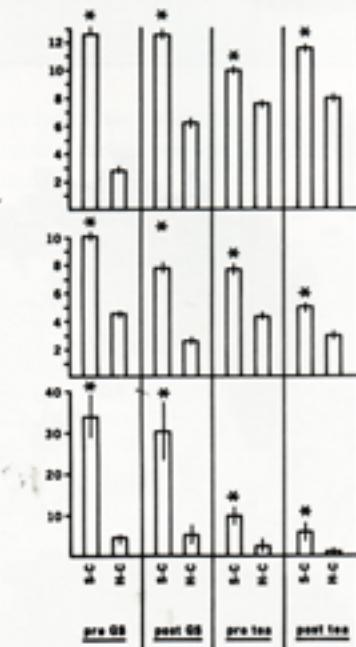
244

Application of the Principles in Editing and Redesign

Just as a good editor of prose ruthlessly prunes out unnecessary words, so a designer of statistical graphics should prune out ink that fails to present fresh data-information. Although nothing can replace a good graphical idea applied to an interesting set of numbers, editing and revision are as essential to sound graphical design work as they are to writing. T. S. Eliot emphasized the "capital importance of criticism in the work of creation itself. Probably, indeed, the larger part of the labour of an author in composing his work is critical labour; the labour of sifting, combining, constructing, expunging, correcting, testing: this frightful toil is as much critical as creative."³

Consider this display, which compares each long bar with the adjacent short bar to show the viewer that, under the various experimental conditions, the long bar is longer:

³T. S. Eliot, "The Function of Criticism," in *Selected Essays 1917-1932* (New York, 1932), p. 18.

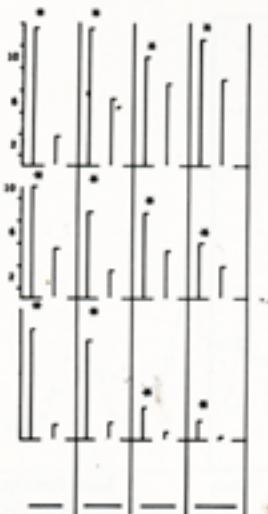


James T. Kuznicki and N. Bruce McCutcheon, "Cross-Enhancement of the Sour Taste on Single Human Taste Papillae," *Journal of Experimental Psychology: General*, 108 (1979), 76.

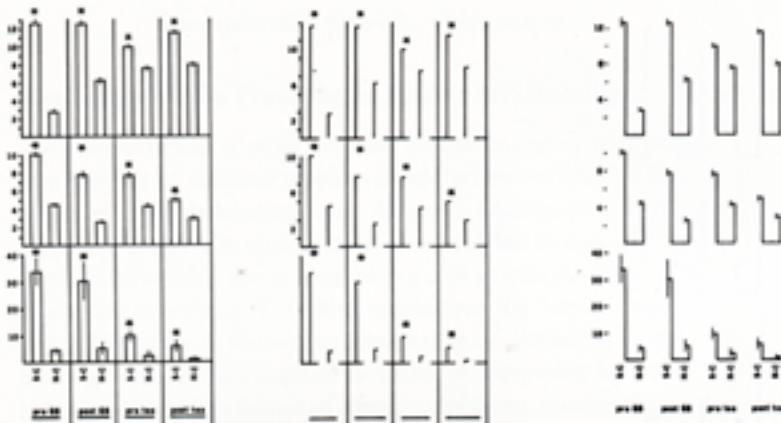
Vigorous pruning improves the graphic immensely, while still retaining all the data of the original. It is remarkable that erasing alone can work such a transformation:



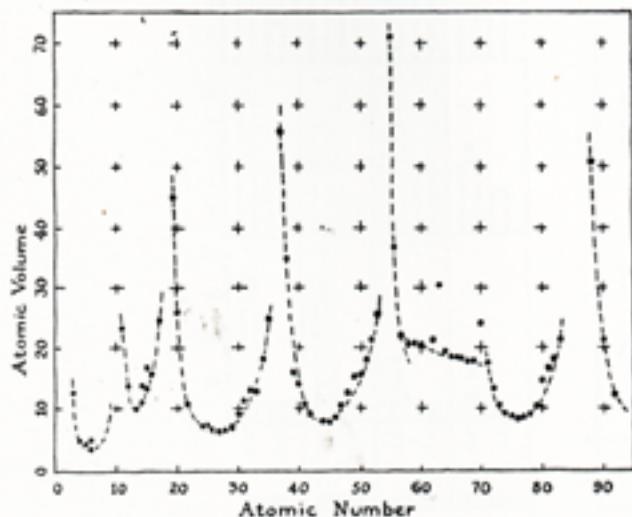
The horizontals indicate the paired comparisons and would change if the experimental design changed—so they count as information-carrying. All the asterisks are out since every paired comparison was statistically significant, a point that the caption can note. Here is the mix of non-data-ink and redundant data-ink that was erased, about 65 percent of the original:



The data graphical arithmetic looks like this—the original design equals the erased part plus the good part:

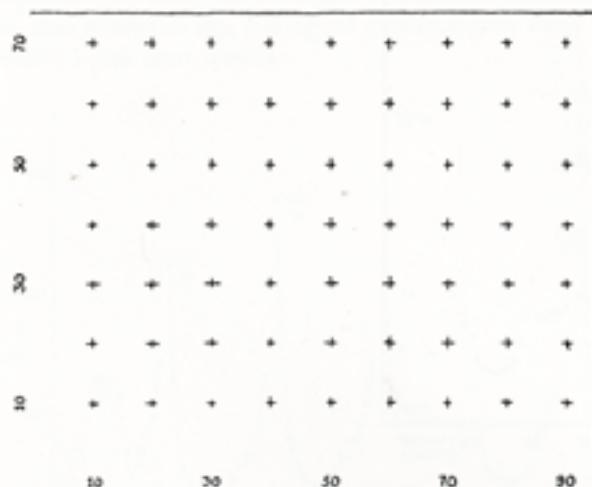


The next graphic, drawn by the distinguished science illustrator Roger Hayward, shows the periodicity of properties of chemical elements, exemplified by atomic volume as a function of atomic number. The data-ink ratio is less than 0.6, lowered because the 76 data points and the reference curves are obscured by the 63 dark grid marks arrayed over the data plane like a precision marching band of 63 mosquitoes:

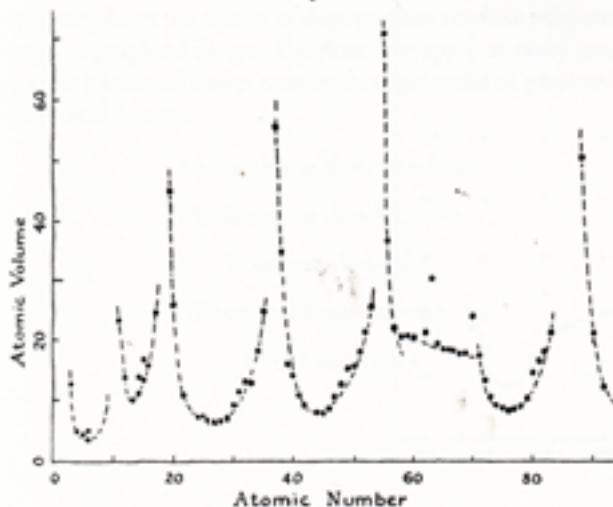


Linus Pauling, *General Chemistry* (San Francisco, 1947), p. 64.

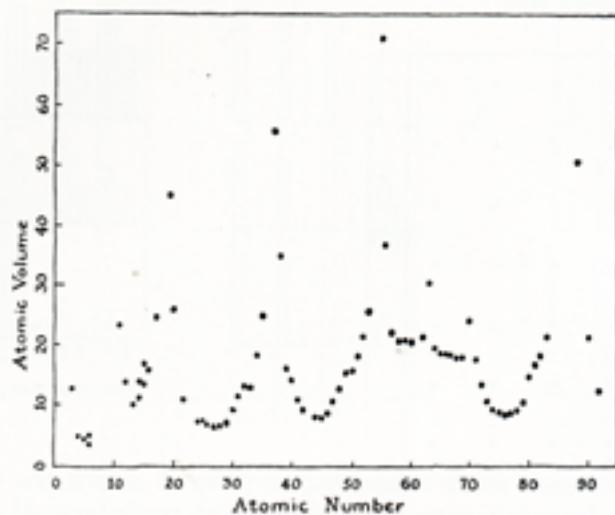
The grid ticks compete with the essential information of the graphic, the curves tracing out the periods and the empirical observations. The little grid marks and part of the frame can be safely erased, removed from the denominator of the data-ink ratio:



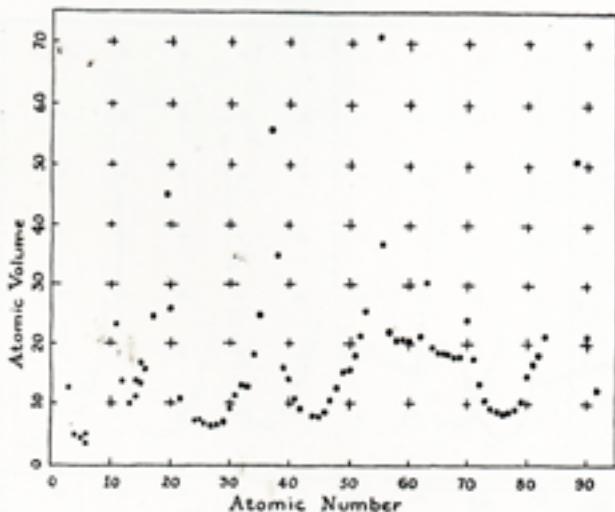
The uncluttered display brings out another aspect of the data: several of the elements do not fit the smooth theoretical curves all that well. The data-ink ratio has increased to about .9, with only the frame lines remaining as pure non-information:



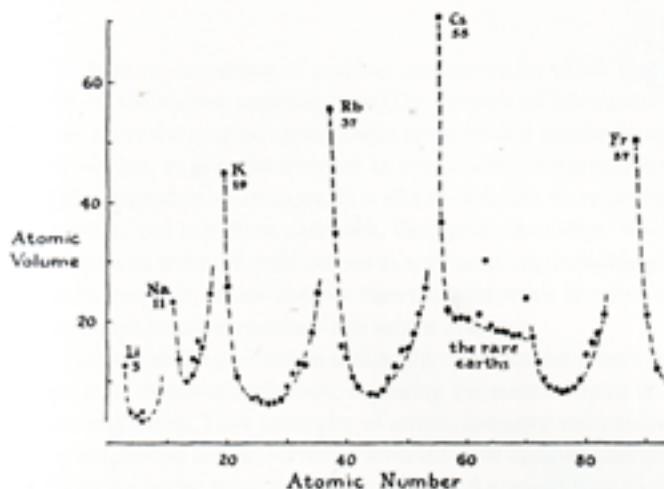
The reference curves prove essential for organizing the data to show the periodicity. The curves create a structure, giving an ordering, a hierarchy, to the flow of information from the page:



Restoring the grid fails to organize the data. The ticks are too powerful, and they also add a disconcerting visual vibration to the graphic. With the ticks, the reference curves become all the more necessary, since the eye needs some guidance through the maze of dots and crosses:



The space opened up by erasing can be effectively used. Labels for the initial elements of each period, an alkali, show the beginning of each cycle in the periodic table of elements—and in the graphic. The unusual rare-earths are indicated. In addition, the label and numbers on the vertical axis are turned to read from left to right rather than bottom to top, making the graphic slightly more accessible, a little more friendly:



Conclusion

Five principles in the theory of data graphics produce substantial changes in graphical design. The principles apply to many graphics and yield a series of design options through cycles of graphical revision and editing.

Above all else show the data.

Maximize the data-ink ratio.

Erase non-data-ink.

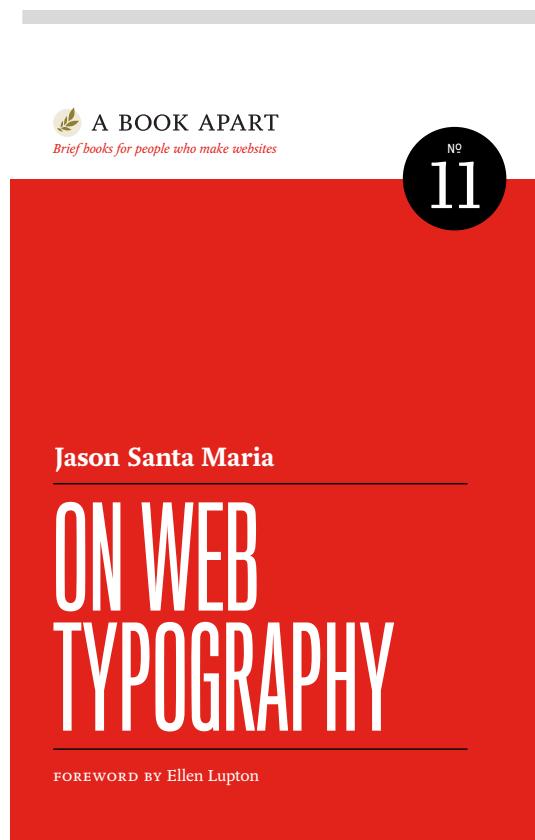
Erase redundant data-ink.

Revise and edit.

On web typography

Jason Santa Maria

“I want to show you that even though this stuff can be difficult to wrap your head around, what you get back is well worth it. Being good at typography makes you a more adept thinker, communicator, and designer. When you immerse yourself in the fine details of text, you not only make yourself aware of those details and how they affect communication, but you also put yourself in your readers’ shoes.”



CLASSIFICATIONS

All typefaces fall into some sort of classification. Unlike the scientific organization of the animal kingdom, however, there has been little consensus on one scheme to rule them all. We struggle with these systems, because typefaces are thoroughly dynamic works diverse in visual structure, intent, influences, and historical context. Defining a classification system that comfortably accommodates typefaces from 500 years ago as well as five months ago—and getting everyone to agree on it—is not an enviable task. Fortunately, a foundation has settled enough for us to build on.

You’re probably familiar with these classifications in a casual capacity. Groupings like *serif*, *sans serif*, and *script* are well known (FIG 3.1). More descriptive subclassifications exist to reference a particular set of physical traits or time periods. For instance,



253

FIG 3.1: Examples of common typeface classifications.

some common subclassifications for serif are *Old Style* (e.g., Bembo) and *Modern* (e.g., Bodoni).

When thinking about typefaces for use in my designs, I mentally sort them into a few common groups: serif, sans serif, slab serif, script, monospace, and decorative (which is mostly made up of anything that doesn't fit into those other categories). Obviously, these groups don't represent all typefaces, but they work for the majority. Understanding the most basic classifications helps you filter the vast number of typefaces out there, and can be handy for searching on the web when you have a rough idea of the look or feel you're after.

While each classification evokes a kind of feeling, it's rarely something you can grab hold of because of how broad the classifications are. Two given typefaces may belong to the same class but spark very different responses, depending on their intended use or when they were made. Look to the typefaces themselves to support a feeling or mood in your design.

While not comprehensive, these working classifications help me sort typefaces against a mix of physical attributes and usage. For example, *decorative* isn't just a synonym for fancy type; it describes the context for using a typeface. A decorative typeface could have serifs or be a script or monospaced, but its overriding characteristics likely prevent it from everyday use.

Thinking of typefaces this way lets me slice my work into smaller chunks. If I'm designing a website that has articles, I'm generally looking at typefaces for running text. Most of what I'm after will fall in the serif and sans serif classifications, as typefaces in other groupings will be too distracting for long

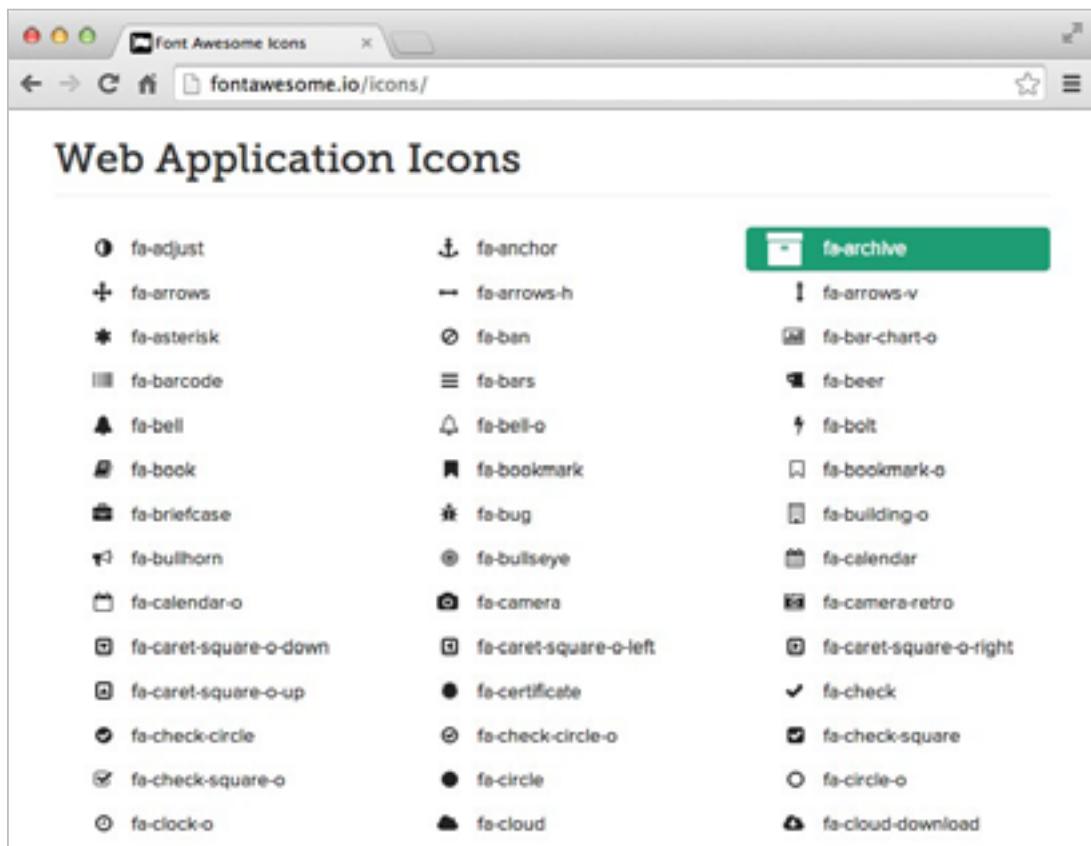


FIG 3.2: An example of an icon font from Font Awesome (<http://bkaprt.com/owt/6/>).

stretches of text. Along those lines, if I'm looking for a typeface to set my new tagline across the top of a website, I may look at more decorative or distinct typefaces first.

Beyond the traditional classifications, there is another category of typefaces: *icon fonts*. Icon fonts have characters filled with symbols other than letters (FIG 3.2). Pictograms in font form—also known as dingbat, symbol, picture, or pi fonts—have been around for a long time, but the urge to use them online is rising. There are clear, practical benefits: a scalable version of your logo or a shopping cart icon that you can size and color with CSS, while keeping the icons bundled as a single loaded asset. The benefits compound with responsive web design, since your icons can always scale up and down to the right size to suit a range of screens and resolutions.

But icon fonts can run into trouble online. Some of them remap individual letters to pictures, leaving little regard for

someone using a screen reader, which may read the letter *e* aloud while displaying an icon of an envelope onscreen. For examples of icon fonts and best practices, check out Filament Group's handy article "Bulletproof Accessible Icon Fonts" (<http://bkaprt.com/owt/7/>). I have found, however, that the cons of icon fonts outweigh the pros, and I tend to agree with Chris Coyier and his article "Inline SVG vs Icon Fonts [Cagematch]" (<http://bkaprt.com/owt/8/>)—SVG is the more flexible solution.

255

Classifications, whether your own personal divisions or ones from an established system, can be a rabbit hole of information and history. And while they're interesting—especially when you want to geek out on type—you don't necessarily need to discern the minutiae of time periods and serif brackets to do your job. Classifications are helpful in the same way knowing about the history of jazz or rock 'n' roll can make you a better musician: they allow you to sort typefaces across criteria and find aesthetic and mental connections to help communicate your design. Sometimes these links are interesting juxtapositions or cultural references. Indra Kupferschmid covers classifications in detail and proposes a more flexible system for the future in "Type Classifications Are Useful, But the Common Ones Are Not" (<http://bkaprt.com/owt/9/>). Knowing about classifications makes you a better designer, because you can traverse these connections and use them to your advantage.

PHYSICAL TRAITS

Classification is only one facet of why a typeface may look and feel a certain way. But what causes some typefaces to appear larger than others when set at the same size? What causes them to feel lighter or heavier? And what makes the same letter look different from one typeface to another? The answers to these questions will help you understand the visual differences between typefaces. Understanding these traits lets you trace the varied ways typefaces approach and solve the same problems.

Typefaces are specialized tools, but they're also the expressive creation of a typeface designer. Some of the choices those designers make when creating a typeface may be based on a personal preference or technical reason. The sum of these choices

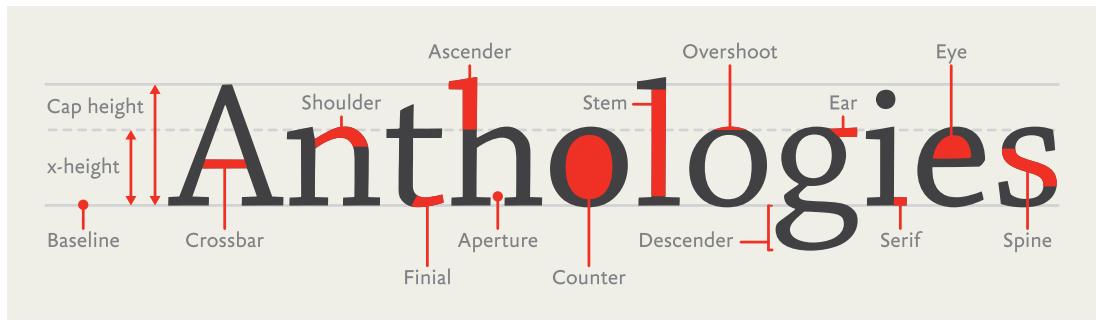


FIG 3.3: Some common parts of typeface anatomy.

influences the features we judge when choosing a typeface: legibility, flexibility, contrast, and more. Similar to classifications, a working knowledge of these traits gives you the power to speak confidently about typography—and helps you make mental connections between typefaces that share traits.

You’re probably familiar with the basics of type: uppercase and lowercase letters, numbers, punctuation, and some special characters. Like any rich visual art form, typography has a vast depth of terminology to describe the diverse parts of letterforms. While you could fill another book discussing type anatomy—like Stephen Coles’s *The Anatomy of Type: A Graphic Guide to 100 Typefaces*—I’d like to point out the most common parts of letters you’ll encounter.

All letterforms are made up of a variety of *strokes*, a general term for most parts of a letter (FIG 3.3). Typefaces whose strokes vary in width to several degrees, from hairline thin to very broad, are known as *high-contrast*. Typefaces whose stroke widths are consistent throughout are *monoline* designs. Some strokes are straight and long, like the stem on a lowercase *h* or the descender on a *p*, while others are short and curved, like the neck and ear on a two-story *g*. Some strokes resolve in serifs, while others, as you sometimes see in the top hook of an *f*, can end in a bulbous shape as a *ball terminal* or a teardrop shape as a *lachrymal terminal*. Some strokes encase whitespace in what’s called a *counter*, like the inside of an *o*.

Understanding the vocabulary of type helps you discover why a typeface looks the way it does, when or where it comes



FIG 3.4: Two typefaces that appear to be different sizes despite being set to the same pixel size.

257

from, and its intended purpose. And greater knowledge of your tools means you’re better equipped to make good decisions. Later in this chapter, we’ll take a closer look at two familiar typefaces, Helvetica and Georgia, to see how we can extend this vocabulary to new typefaces.

In addition, a strong vocabulary for type is excellent when you’re critiquing work. For instance, when you see inconsistent strokes or when something within a design feels off balance, you can point to a specific piece by name, rather than saying something like “that wiggly bit.” That precision results in more productive discussions. For more information on why letters look the way they do, check out Tim Brown’s article “Drawing Letters” (<http://bkapr.com/owt/10/>).



FIG 3.6: Setting the same text in two separate fonts at the same `font-size` can result in two visually different sizes.

TYPEFACE CONTRAST

The *contrast* of a typeface refers to the differences in the thick and thin strokes of its characters. A monoline typeface has the absolute least amount of contrast. A typeface with low contrast has some, but relatively little, variation in the thickness of its strokes. For instance, Helvetica features consistent stroke widths. Compare that to a high-contrast typeface like Bodoni, whose strokes vary from beefy to delicate, all in one letter (**FIG 3.7**).

Higher-contrast typefaces tend to be useful in small bursts or headlines, because the extreme variation in stroke width is burdensome in long text. Our eyes are attracted to the exceptions—the stuff that looks different from everything else. Contrast is not only a duality of thicks and thins in the typeface, but it also involves the whitespace between and inside the letters. That variance adds up. I find I'm more likely to stop or slow down



FIG 3.7: Two typefaces with differing contrasts.

259

and notice the letters, rather than read, when my eye encounters that kind of modulation.

A typeface with less contrast can create a smooth, welcoming rhythm for reading. Most typefaces intended for long-form text have medium to low contrast, which creates less interplay between the individual letters and words. This gives text a steadier visual rhythm as your eyes move across a line, which in turn aids readability. When we aren't distracted by the exceptions, we can focus on the act of reading itself. On the other hand, too little contrast in stroke or distinction between letterforms, as in the case of Helvetica, can be unsuitable for long stretches of text because the letterforms appear too uniform, reducing legibility. Like most things in design, it's about finding the right balance.

WEIGHTS AND STYLES

Many typeface families have at least four basic styles: *regular* (sometimes called *roman* or *book*), *italic*, *bold*, and *bold italic*. A style's *weight* refers to the thickness of its strokes, or their boldness. *Posture* refers to an alteration of the letter's skeleton, like the difference between regular and italic. However, variations in style aren't limited to weight and posture; they can include different optical sizes, numeral sets, and many other kinds of structural and stylistic alternates.

On the web, we commonly employ numerical CSS `font-weight` values. These currently range from `100` to `900`—nine in total, each on the whole hundred value—with the lightest weight at `100` and the heaviest at `900`. Not all fonts include every weight (some may only have a single weight), but this serves as

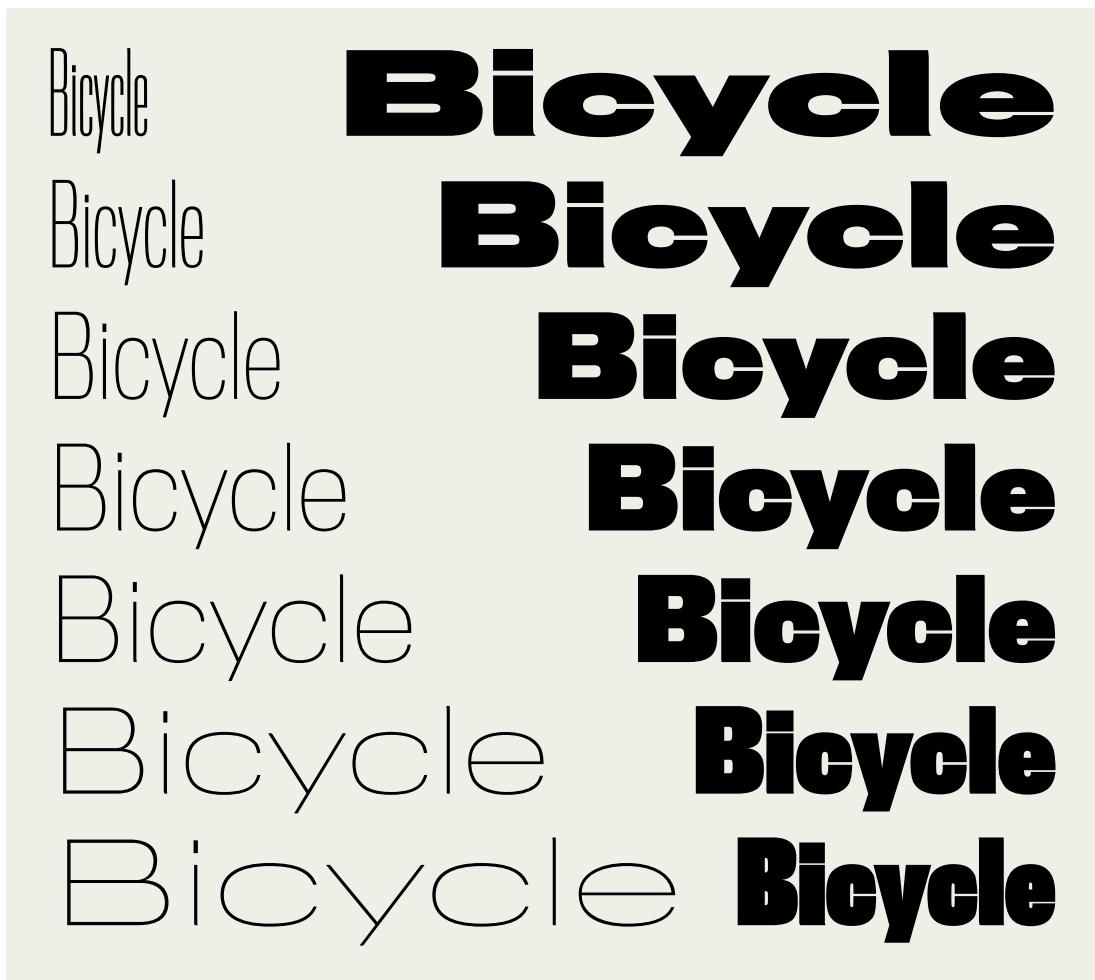


FIG 3.8: Some styles from the expansive Titling Gothic family.

a framework. While some typefaces deviate a little up or down the spectrum, most fonts place their **normal** or book weight at **400** and their **bold** at **700**. Since these names and numbers aren't absolutely prescriptive, it's best to make your judgment visually to be certain you have the weight you want.

The four basics are a standard minimum set of family members, especially for text faces, but some typefaces are part of massive families. Take Titling Gothic by Font Bureau, the condensed sans serif used on the covers and chapter openings of the *A Book Apart* series. The style used for headlines is regular condensed, but it is only one of fifty-eight styles (!) that make up the Titling Gothic family. Family members range from very thin and condensed to very heavy and wide (FIG 3.8).

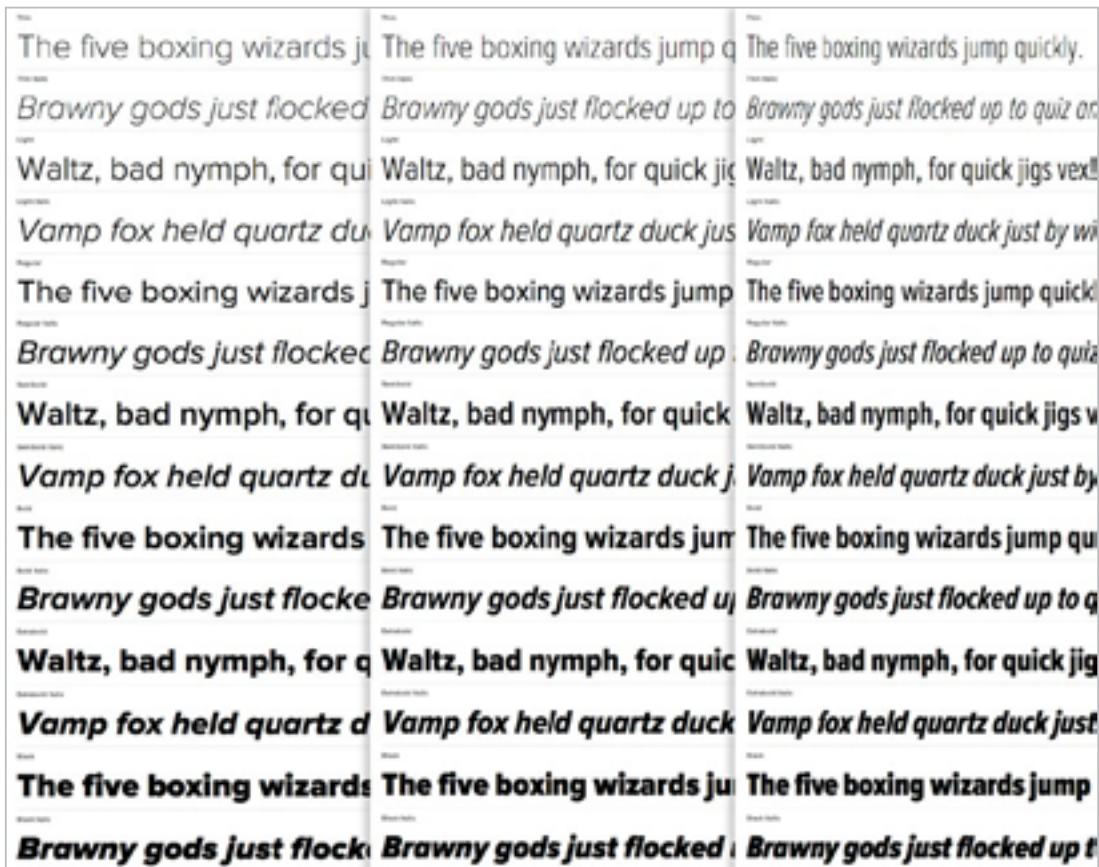


FIG 3.9: Proxima Nova, a very large type family by Mark Simonson.

Many families include condensed and expanded widths, which are what they sound like: *condensed* and *compressed* widths feature narrower letterforms, while *extended* and *expanded* ones have wider letterforms (FIG 3.9).

Why would anyone need so many styles? It depends on what you’re designing. Flexibility is often one of my biggest considerations when choosing a typeface. If I can create a good level of contrast by using different styles within a typeface family, it accomplishes a few important things. For one, it lets me stay stylistically consistent within my design. I can use different styles for headlines, subheads, and maybe even text, and they will all share a common background throughout the piece. That flexibility helps me establish a clear hierarchy while keeping the visual language simple.

FIG 3.10: Liz Danzico's personal site uses a single type family but still achieves design diversity among different kinds of content (<http://bkaprt.com/owt/11/>).

I find that the more typefaces I use in a design, the weaker the design becomes. Many people say not to use more than one or two typefaces at the most. Though obviously not a rule, this can be a good guideline as it puts some constraints on your visual palette (**FIG 3.10**). Having a variety of typefaces can create a cacophony of mixed styles and messages. But when you work from a smaller pool of options, you allow yourself to rely on typographic attributes like size and color to create distinction. And those elements will naturally feel like they belong together, because they come from the same place.

Pay close attention to the members of a type family on the web, especially where browsers may try to fill in for a missing style. When text is set to **bold** or **italic** in CSS, the browser

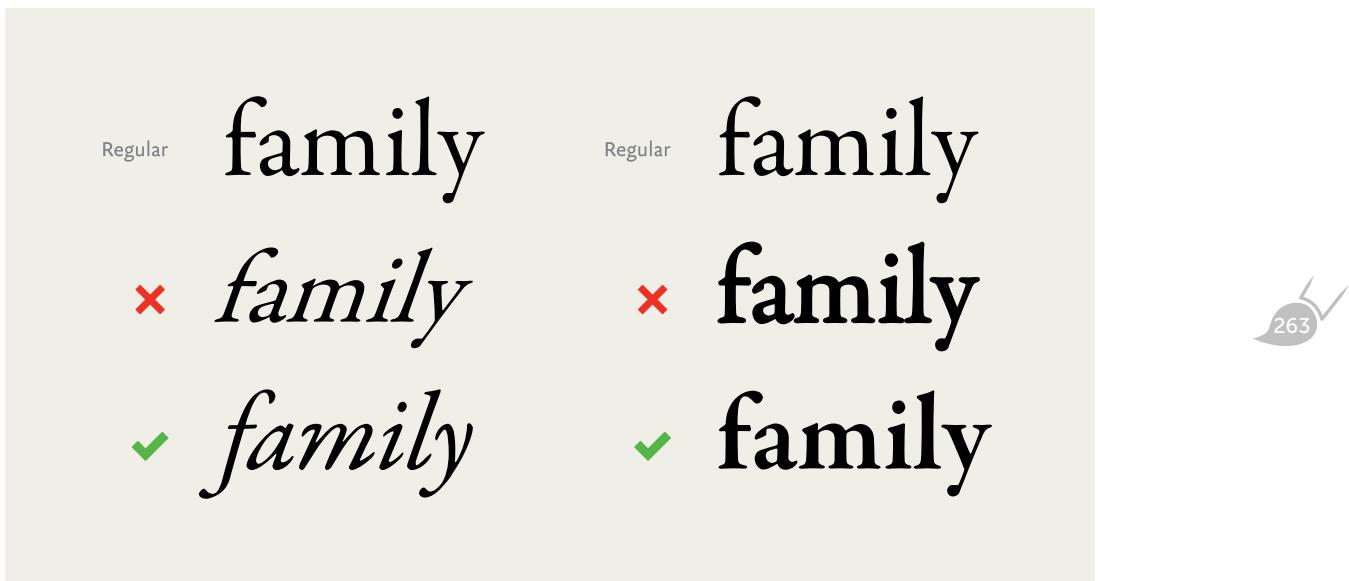


FIG 3.11: Comparing browser-generated pseudo italics and bolds with proper ones.

will first look for the appropriate font to render it. If that font is absent, the browser will try to formulate an italic or bold by artificially skewing or beefing up the letterforms from the existing font.

We call these *pseudo* or *faux* italics and bolds, and they're the typographic equivalent of mistakenly tucking your shirt into your underwear (**FIG 3.11**). Italics are not merely slanted letters, but instead have different shapes adapted from the typeface's normal upright style. Notice how the correct italic and bold seem tailored for their form? In this typeface, the italic letters develop a slant and some trailing strokes, and some letters change drastically, like the lowercase *a* morphing from a two-story to a one-story letter. Now look at the faux italic. It's literally a skewed version of the normal style. Some of the letters' thinner bits look squished, and the counters appear misshapen.

Moving to the faux bold, the letters are a bit blobby, like someone spilled water on paper and the ink started to bleed. The letters' serifs crowd together because the spacing wasn't created with these mutated letterforms in mind. Proper bold weights feel heavier than normal weights, but they may not have a uniform



FIG 3.12: Two typefaces with different x-heights.

increase in their body size. In this example, the correct bold is thicker in the areas that were already heavy, but the thinner strokes and serifs remain largely unchanged. Small details like this allow the visual weight of the typeface style to increase, but keep parts like counters from filling in. Choose typefaces that have the styles your text needs to display properly, or you will be laughed right off the web. Okay, maybe not, but it's an ugly misstep you can easily avoid.

X-HEIGHT

A typeface's *x-height* refers to the height of its lowercase letters from the baseline (the implied line that the letters rest on) to the top of an uppercase letter (FIG 3.12). Just as an em has little to do with the letter *M*, x-height does not specifically refer to the height of the lowercase *x*. But because we're talking about the height of lowercase letters, the two are usually equal.

Like the relative size of a typeface's body, the x-height can be as large or small as the type designer wishes. Some typefaces have very low x-heights, which can communicate elegance, as in the case of some script typefaces. A low x-height can also create an interesting tension between letterforms, as the contrast is more pronounced between upper and lower cases.

When considering text faces, a high x-height is usually ideal; more space for the letterform means more information to help the reader. This is true of typefaces for print or web, but is of utmost importance where interfaces or wayfinding are a concern.

FINDING ALTERNATIVES

The rewards of typographic knowledge are cumulative. If you already know a typeface well, you can build on that knowledge to find other typefaces. To do so, let's look at some specific visual attributes of a typeface. By scrutinizing a typeface, we can quickly determine if it suits our needs. Let's start with something we all recognize as an example: Helvetica.

Now, Helvetica is about as pervasive as a typeface gets; it's used on everything from logos to public signage around the globe. But is Helvetica the best choice for all of these uses? What are the visual attributes that make Helvetica *Helvetica* (FIG 3.27)? Let's take a closer look:

- Helvetica has very little stroke contrast; the lines are basically the same weight.
- It has a generous x-height.
- The letters are based on simple geometric forms.
- The apertures, or openings inside of the letters, are nearly closed. Helvetica hugs that space tightly.
- The terminals, or ends of the strokes, are at right angles.

From these attributes, we can deduce a few things: Helvetica is clear and geometric, but not always very legible. Since the letterforms carry so little variation, it can be easy to confuse some letterforms for others. If we need a typeface for small type or long-form content, we may need to keep looking.



FIG 3.27: A few physical attributes of Helvetica.



FIG 3.28: Some Helvetica alternatives.

Staunch devotees of Helvetica may decry any criticism of the typeface. Some people see it as the ultimate typeface for design, because it's basically a blank slate. They think you can throw anything at Helvetica and it will look just fine, because Helvetica brings little baggage and few connotations.

I feel the opposite. Helvetica is technically a beautiful face, but it's also so overused that I have trouble feeling any response when I see it. To me Helvetica has become a generic default. People use it as a safe choice rather than face the fear of making a bad choice. They'd rather say nothing than risk saying the wrong thing.

Take a stronger stance. Since we already broke down some of Helvetica's attributes, you can seek out typefaces that share

similar traits (FIG 3.28). And don't worry if you aren't sure where to start looking—we'll cover some good resources for finding typefaces in the next chapter.

To start, let's look at FF Dagny. FF Dagny has similar proportions and counters, but it's slightly more compact without the perpendicular terminals. FF Dagny also feels more diminutive than Helvetica, and not so machined. Another option is Pragmatica Slabserif, which shares many of Helvetica's physical attributes but adds serifs, making it feel more academic. Either is a good option if you're familiar with Helvetica but want something a little different.

Let's try another example: Matthew Carter's darling of early web typography, Georgia. Used widely across the internet, Georgia is a modern workhorse for onscreen type. As we saw in the last chapter, Carter not only designed it for the screen, but he built Georgia to stand up to some of the least hospitable rendering environments. Can we find typefaces that embody that same durability but aren't as omnipresent? First, we need to break down what makes Georgia *Georgia* (FIG 3.29).

Let's take a closer look:

- Georgia features a moderate stroke contrast.
- It has a generous x-height, counters, and spacing across letters.
- It has some sharp and pointy angles, as if it were elbowing its way through a crowded room.
- It has beefy, almost slab-like serifs.

We gravitate toward a few key things. Due to its spacing, high x-height, and lower stroke contrast, Georgia is a great candidate for setting long swaths of text. And it's true: Georgia is a comfy typeface to sit back and read with. Because of the lower contrast, the letterforms carry a good rhythm, so text set in Georgia seems to flow. Georgia also has a mild heft to it—it's not quite a slab serif, but not a delicate flower either. With these traits in mind, we can track down typefaces that share similar attributes (FIG 3.30).

Take Chaparral (a personal favorite). It has some of the same traits that make Georgia so legible, like its x-height and openness, but with softer angles that evoke a more restrained elegance. Or



FIG 3.29: A few physical attributes of Georgia.



FIG 3.30: Some Georgia alternatives.

look at FF Tisa—a sure option for running text with more modern serifs and even less stroke contrast. Lastly, Droid Serif takes Georgia's angularity a step further as a pointier, boxier cousin, almost as if it's sucking in a small paunch to impress someone.

Comparing type like this is one of my favorite exercises because you can clearly see the lines between typefaces. We don't need to match every trait of a typeface we like to an alternate. The important thing is to recognize the traits that make the typeface unique. Playing off of that knowledge is extremely useful, because you can apply many of the same typographic methods as you would to the original typeface. Most important, building on that knowledge saves you time and makes you more proficient with type.

KNOW YOUR CONTEXT

As with any creative process, there are many approaches to choosing type, and it's a personal pursuit to find what works best for you and what feels most productive. But you have quantifiable considerations too, like the conditions under which you're going to use a typeface. I'm a minimalist by nature, so I like to simplify the kinds of uses for type into two camps: type for a moment and type to live with. Let me show you what I mean.

Type for a moment

Put simply, *type for a moment* is content that someone should only need a moment to read. This includes small interface copy (like a button or login link), brief asides to an article, and display type for a headline or large marketing copy (FIG 4.1).

As another example, think of it like a sign in an airport. It needs to quickly convey its meaning and let you be on your way. You don't want to spend precious minutes when you're late for a flight trying to decipher the artistic intent behind a sign before you can run off to your gate. And you don't want people to puzzle over where to click to find what they need.

After you get your readers' attention, there may be more to discover, like in the case of a logo or a big, promotional headline, but the key is to be clear up front.

The typeface you start with should suit a purpose. For navigation or interface text, I look for typefaces that hold up well at small sizes and aren't fussy with extra style. More times than not, I narrow it down to a simple sans serif. Let's look at JAF

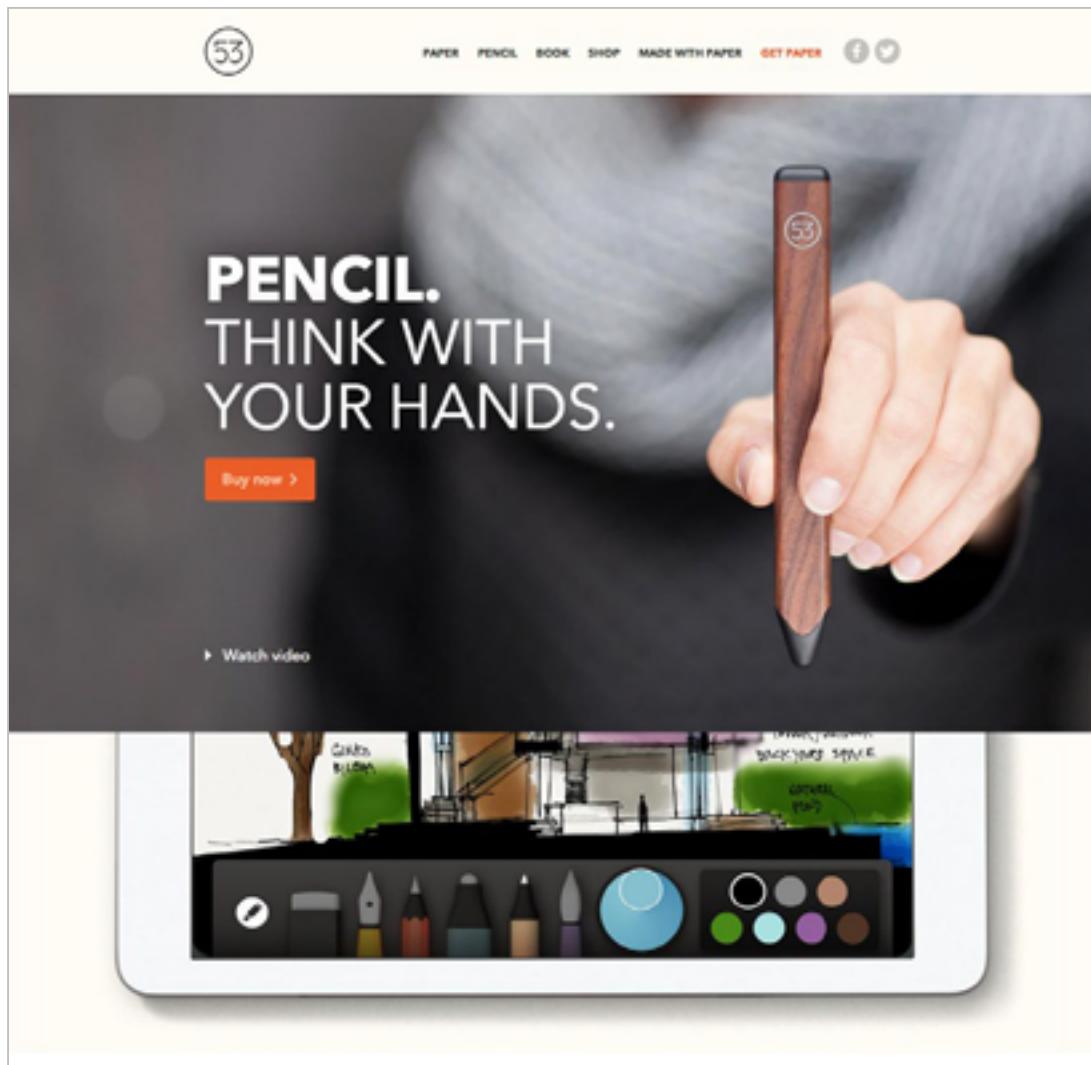
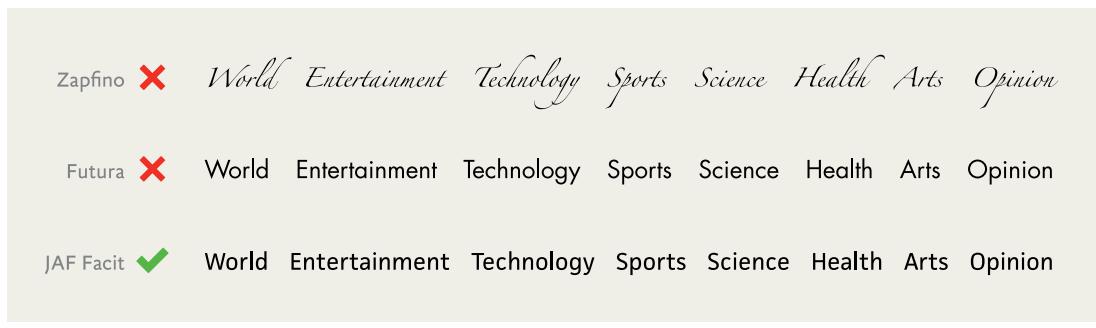


FIG 4.1: An example of a big headline from FiftyThree's Pencil homepage (<http://bkapr.com/owt/29/>).

Facit to see why. It can get very small, but it remains clear and legible because of its large, open counters and simple construction. Compare that with Futura or Zapfino (FIG 4.2). Either could work when used sparingly in a headline, but they both appear fussy as navigational text. They also take up more space. The strict simplicity of Futura's letterforms makes some letters hard to tell apart (for instance, the lowercase *a* and *o* look very similar), not to mention the tight spacing and small x-height. And Zapfino is a bit too decorative at the expense of legibility.



271

FIG 4.2: JAF Facit versus Futura and Zapfino for navigational text.



FIG 4.3: Bello is beautiful, but not suited for longer text.

At larger sizes—especially in print or on a higher-resolution display—even delicate, light weights are easily readable. You can run the gamut of decorative typefaces and settings, as long as there isn’t too much text. For extremely decorative headlines, stick to a few words or less. With these short bursts, type acts more like an image, so the viewer is more forgiving.

As you can see in **FIG 4.3**, using a decorative face like Bello for a headline doesn’t give us much trouble as readers. A headline with a lot of personality immediately sets a mood, which can be a great way to draw a reader in. That said, you wouldn’t want to use Bello for more than a line or two, as it quickly becomes cumbersome to read. Like salt, country music, and in-laws, a little goes a long way.

Type to live with

Type to live with is text we spend a lot more time with, usually long-form text like an article or book. The typefaces you use here can mean the difference between someone reading or not. If a typeface is too loud, too high contrast, or otherwise disruptive, we might lose the reader. And we can't blame them either—reading large swaths of text in all caps or in a decorative face is like yelling at a reader when you really mean to talk in an even tone.

Typefaces for longer reading should give a page an even texture. The texture of flowing text is the sum of the typeface's color (the general combinations of lights and darks in and around letters), the actual color of the text and its contrast with the background, and the size of the setting. You can see what I mean by blurring your eyes while looking at a chunk of text. If you see repeating patterns of weird letterforms that keep sticking out to you, the typeface may not be right. We'll look at typographic color in Chapter 6, but the general guideline is that a reader shouldn't notice the type. They shouldn't stop or stumble over the text, or wonder why something looks the way it does. Because when a reader notices the type, they're taken out of the act of reading and are instead trying to decode why something else is calling attention to itself.

In 1930, renowned typographic scholar Beatrice Warde penned an essay titled “The Crystal Goblet, or Printing Should Be Invisible” on this topic (<http://bkaprt.com/owt/30/>). She compares typographic choices to the difference between drinking wine from a clear crystal goblet—a vessel that lets you fully experience its contents visually alongside your other senses—or a goblet of “solid gold, wrought in the most exquisite patterns” that trades function for form. Warde urges designers to strive for clear presentation of their messages, allowing the contents to speak for themselves, and for designs to be a transparent window to those messages.

You want that clear goblet. Help people forget that they're staring at a screen and instead immerse them in the words and the story you're telling. The type you use should be smooth, removing as much friction as possible between the reader and the text (**FIG 4.4**).



source: Steve Schneider

<http://info3iomatswork.blogspot.nl/2012/04/glitch-art.html>

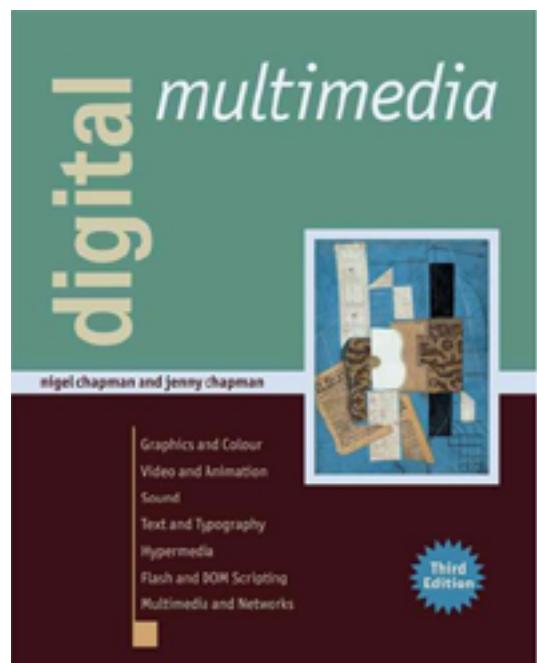
appendix—**multimedia**

Digital multimedia

Nigel Chapman & Jenny Chapman

In order to design effectively, you have to understand your medium. You cannot produce effective print if you don't understand the difference between vector and raster images. You cannot produce web-applications if you don't understand HTML and CSS.

We've included two chapters of this book, on [video](#) and [sound](#), to show the technical depth that a choice of medium can bring with it. Studying these should give you an idea of how deep you have to dive into the technical minutiae of a medium you choose to produce in.



6

Video

■ **Video Standards**

Analogue Broadcast Standards. Digital Video Standards. DV and MPEG. High Definition Formats.

■ **Video Compression**

Spatial Compression. Temporal Compression. MPEG-4 and H.264/AVC. Other Video Codecs. Quality.

■ **Editing and Post-Production**

Traditional Film and Video Editing. Digital Video Editing. Post-Production.

■ **Delivery**

Streaming. Architectures and Formats.

Video is a medium which has been revolutionized by digital technology in a short period of time. In the late 1990s, video cameras were almost exclusively analogue in nature. Importing video footage into a computer system relied on dedicated capture cards to perform the digitization. Digital video editing placed considerable demands on the hardware of the time – much editing was still done on analogue equipment, by copying back and forth between three recording decks. Less than 10 years later, digital video had become the norm. Affordable digital video camcorders are widely available for the consumer market, and higher-end digital equipment is used for professional applications, from news-gathering to feature film-making. Tiny video cameras are built into mobile phones and computers and it is possible to capture activity on a screen directly to video, without even using a camera. Non-linear digital video editing software that runs on modestly powerful systems is used routinely by both amateurs and professionals.

As a result of this explosive spread of digital video technology, coupled with the higher network speeds of broadband Internet access, video has become a prominent feature of the World Wide Web and the Internet. Web sites dedicated to the presentation and sharing of video have proliferated, but video has also become a common element among other media on many sites. News sites often include embedded video clips among textual news items, and support sites for software increasingly rely on video “screencasts” to demonstrate features of programs by showing them in action. Video is also used for communicating over the Internet: any suitably equipped computer can act as a video phone. As well as showing the participants to each other, video chat applications allow them to show each other images and recorded video clips.

Several factors have made these developments possible. First is the rapid increase in processor speeds and memory, disk capacity and network bandwidth. Second is the development of standards for digital video signals and interfaces, which have largely replaced the earlier confusion of incompatible capture cards and proprietary codecs. Finally, the move to digital video has been driven by its convenience and robustness, and the flexibility and relative simplicity of digital video editing compared to its analogue equivalent.

The high-end professional facilities used for making feature films and top-quality broadcast video lie beyond the scope of this book. For multimedia work, there are two broad classes of hardware and software that are in common use.

Where good quality is required, the most widely used combination of hardware for capturing video comprises a digital camcorder or VTR (video tape recorder) using one of the variants of

the **DV** format – mini-DV (often simply called “DV”), DVCAM or DVCPRO – connected to a computer by a **FireWire** interface. (FireWire was formerly known as IEEE 1394, but the more colourful name has now been officially adopted; equipment made by Sony uses the name iLink for the same interface.) These devices capture full-screen video, with frames that are the same size as those used by broadcast TV; they also work at one of the standard television frame rates.

281

The three DV variants use different tape formats and provide differing degrees of error correction and compatibility with analogue studio equipment, but all send digital video as a data stream to a computer in the same format, so software does not need to distinguish between the three types of equipment. Mini-DV is essentially a consumer format, although it is also used for semi-professional video production. The other two formats are more suited for professional use, being especially widely used for news gathering. All DV equipment supports **device control**, the ability for the tape to be stopped, started and moved to a specific position by signals sent from the computer by software.

IN DETAIL

“DV” stands for “digital video”, but that expression is also used in a more general sense, to refer to the storage and manipulation of video data in a digital form, and sometimes it is abbreviated to “DV” when used in this way, too. We will usually use the full term “digital video” in this general sense, and only use “DV” whenever we mean the specific standard we have just introduced.

Some camcorders have an internal hard disk, instead of using tape, while others write directly to DVDs. Such devices may still use the DV format and connect via FireWire, or they may use the MPEG-2 format used on DVDs, and connect via USB. Increasingly, DV equipment employs **High Definition (HD)** standards, which provide higher resolution, but this does not affect the technology in other ways.

Although the subjective quality of DV is very good, it is a compressed format, and as we saw in the case of bitmapped still images in Chapter 4, compression causes artefacts and interferes with subsequent processing and recompression. Figure 6.1 shows a frame of uncompressed video and the same frame compressed as DV. It is hard to see any difference in the full frames, at the top left of each group of images. However, as the blown-up details show, there are visible compression artefacts in the DV. (They are especially noticeable in the water at the bottom of the frame.) As the extreme blow-ups demonstrate, the colour values of the actual pixels have changed considerably in some areas.

The user has no control over the quality of DV. The data stream produced by a digital video camera is required to conform to the appropriate standard, which stipulates the data rate for the



Figure 6.1. Comparison of an uncompressed frame (top) and a DV frame (bottom)

data stream, and thus the amount of compression to be applied. If higher quality is required, it will be necessary to use expensive professional equipment conforming to different standards. High-end equipment does allow uncompressed video to be used, but this places great demands on disk space, as we showed in Chapter 2.

Where quality is much less important than cost and convenience, a completely different set of equipment is common. The cheap video cameras built into mobile phones or laptop computers are not generally DV devices. Usually, the compression and storage format are both defined by the MPEG-4 standard, or a simplified version of it designed for mobile phones, known as 3GP. The frame size is usually small enough to fit a mobile device's screen, and the frame rate is often reduced. All of these factors ensure that the size of the video files is very small, but the result is a substantial loss of quality. When video is transferred from a low-end device of this sort to a computer, it is usually through a USB 2.0 connection, not via FireWire. External cameras that connect in this way can also be obtained. They are generally referred to as Webcams, because they are often used for creating live video feeds for Web sites.

Video Standards

Digital video is often captured from video cameras that are also used to record pictures for playing back on television sets – it isn't currently economically practical to manufacture cameras (other than cheap Webcams) purely for connecting to computers. Therefore, in multimedia production we must deal with signals that correspond to the standards governing television. This means that the newer digital devices must still maintain compatibility with old analogue equipment in essential features such as the size of frames and the frame rate, so in order to understand digital video we need to start by looking at its analogue heritage. (Although **HDTV** uptake is increasing, the original television standards are still in widespread use around the world, and many areas do not have standard definition digital television yet, although this varies from one country to another and will change over time.)

Analogue Broadcast Standards

There are three sets of standards in use for analogue broadcast colour television. The oldest of these is **NTSC**, named after the (US) National Television Systems Committee, which designed it. It is used in North America, Japan, Taiwan and parts of the Caribbean and of South America. In most of Western Europe, Australia, New Zealand and China a standard known as **PAL**, which stands for Phase Alternating Line (referring to the way the signal is encoded) is used, but in France, Eastern Europe and countries of the former Soviet Union **SECAM** (*Séquentiel Couleur avec Mémoire*, a similar reference to the signal encoding) is preferred. The standards used in Africa and Asia tend to follow the pattern of European colonial history. The situation in South America is somewhat confused, with NTSC and local variations of PAL being used in different countries there.

The NTSC, PAL and SECAM standards are concerned with technical details of the way colour television pictures are encoded as broadcast signals, but their names are used loosely to refer to other characteristics associated with them, in particular the frame rate and the number of lines in each frame. To appreciate what these figures refer to, it is necessary to understand how television pictures are displayed.

For over half a century, television sets were based on CRTs (cathode ray tubes) – like older computer monitors – which work on a raster scanning principle. Conceptually, the screen is divided into horizontal lines, like the lines of text on a page. In a CRT set, three electron beams, one for each additive primary colour, are emitted and deflected by a magnetic field so that they sweep across the screen, tracing one line, then moving down to trace the next, and so on. Their intensity is modified according to the incoming signal so that the phosphor dots emit an appropriate amount of light when electrons hit them. The picture you see is thus built up from top to bottom as a sequence of horizontal lines. (You can see the lines if you look closely at a large CRT TV screen.) Once again, persistence of vision comes into play, making this series of lines appear as a single unbroken picture.

As we observed in Chapter 2, the screen must be refreshed about 40 times a second if flickering is to be avoided. Transmitting an entire picture that many times a second requires an amount of

bandwidth that was considered impractical at the time the standards were being developed in the mid-twentieth century. Instead, each frame is therefore divided into two **fields**, one consisting of the odd-numbered lines of each frame, the other of the even lines. These are transmitted one after the other, so that each frame (still picture) is built up by **interlacing** the fields (Figure 6.2). The fields are variously known as odd and even, upper and lower, and field 1 and field 2.

Interlacing may become evident if the two fields are combined into a single frame. This will happen if a frame is exported as a still image. Since fields are actually separated in time, an object that is moving rapidly will change position between the two fields. When the fields are combined into a single frame, the edges of moving objects will have a comb-like appearance where they are displaced between fields, as shown in Figure 6.3. The effect is particularly evident along the bottom edge of the cloak and in the pale patch in its lining. To prevent this combing effect showing when constructing a single frame, it may be necessary to “de-interlace”, by averaging the two fields or discarding one of them and interpolating the missing lines. This, however, is a relatively poor compromise.

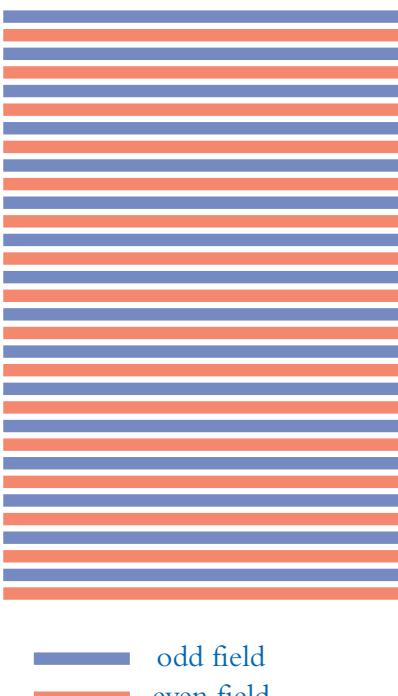


Figure 6.2. *Interlaced fields*



Figure 6.3. Separated fields and combined frame (right) showing combing

Originally, the rate at which fields were transmitted was chosen to match the local AC line frequency, so in Western Europe a field rate of 50 per second – and hence a frame rate of 25 per second – is used for PAL. In North America a field rate of 60 per second was used for black and white transmission, but when a colour signal was added for NTSC it was found to cause interference with the sound, so the field rate was multiplied by a factor of 1000/1001, giving 59.94 fields per second. Although the NTSC frame rate is often quoted as 30 frames per second, it is actually 29.97.

When video is played back on a computer monitor, it is not generally interlaced. Instead, the lines of each frame are written to a frame buffer from top to bottom, in the obvious way. This is known as ***progressive scanning***. Since the whole screen is refreshed from the frame buffer at a high rate, flickering does not occur, and in fact much lower frame rates can be used than those necessary for broadcast. However, if video that originally consisted of interlaced frames is displayed in this way, combing effects may be seen.

Each broadcast standard defines a pattern of signals to indicate the start of each line, and a way of encoding the picture information itself within the line. In addition to the lines we can see on the picture, some extra lines are transmitted in each frame, containing synchronization and other

information. An NTSC frame contains 525 lines, of which 480 are picture; PAL and SECAM use 625 lines, of which 576 are picture. It is common to quote the number of lines and the field rate together to characterize a particular scanning standard; what we usually call NTSC, for example, would be written as 525/59.94.

286

IN DETAIL

It is possible that you might need to digitize material that was originally made on film and has been transferred to video tape. This would be the case if you were making a multimedia film guide, for example. Most film footage is projected at 24 frames per second so there is a mismatch with all the video standards. In order to fit 24 film frames into (nearly) 30 NTSC video frames, a stratagem known as "3-2 pulldown" is employed. The first film frame is recorded for the first three video fields, the second for two, the third for three again, and so on. If you are starting with material that has already had this conversion applied, it is best to remove the 3-2 pulldown after it has been digitized (a straightforward operation with professional video editing software) and revert to the original frame rate of 24 per second. Using PAL, films are simply shown slightly too fast, so it is sufficient to adjust the frame rate.

Digital Video Standards

The standards situation for digital video is no less complex than that for analogue video. This is inevitable, because of the need for backward compatibility with existing equipment – the use of a digital data stream instead of an analogue signal is orthogonal to scanning formats and field rates, so digital video formats must be capable of representing both 625/50 and 525/59.94. The emerging HDTV (high-definition television) standards should also be accommodated. Some attempt has been made to unify the two current formats, but unfortunately, different digital standards for consumer use and for professional use and transmission have been adopted. Only cameras intended exclusively for capturing material to be delivered via computer systems and networks can ignore television broadcast standards.

Like any analogue data, video must be sampled to be converted into a digital form. A standard officially entitled **Rec. ITU-R BT.601** but more often referred to as **CCIR 601**[†] defines sampling of digital video. Since a video frame is two-dimensional, it must be sampled in both directions. The scan lines provide an obvious vertical arrangement; only the lines of the actual picture are relevant, so there are 480 of these for NTSC and 576 for PAL. CCIR 601 defines a horizontal sampling picture format consisting of 720 luminance samples and two sets of 360 colour difference samples per line, irrespective of the scanning standard. Thus, ignoring the colour samples and interlacing for a moment, an NTSC frame sampled according to CCIR 601 will consist of 720×480 pixels, while a PAL frame will consist of 720×576 pixels.

[†] CCIR was the old name of the organization now known as ITU-R.

Observant readers will find this perplexing, in view of our earlier statement that the sizes of PAL and NTSC frames are 768×576 and 640×480 pixels, respectively, so it is necessary to clarify the situation. PAL and NTSC are analogue standards. Frames are divided vertically into lines, but each line is generated by a continuous signal, it is not really broken into pixels in the way that a digital image is. The value for the number of pixels in a line is produced by taking the number of image lines (576 or 480) and multiplying it by the aspect ratio (the ratio of width to height) of the frame. This aspect ratio is 4:3 in both PAL and NTSC systems, which gives the sizes originally quoted. Video capture cards which digitize analogue signals typically produce frames in the form of bitmaps with these dimensions.

The assumption underlying the calculation is that pixels are square. By relaxing this assumption so that there are always 720 pixels in a line, CCIR 601 is able to specify a sampling rate that is identical for both systems. Since there are the same number of pixels in each line for both PAL and NTSC, and $30/25$ is equal to $576/480$, the number of pixels, and hence bytes, transmitted per second is the same for both standards. CCIR 601 pixels, then, are not square: for 625 line systems, they are slightly wider than they are high, for 525 line systems, they are slightly higher than they are wide. Equipment displaying video that has been sampled according to CCIR 601 must be set up to use pixels of the appropriate shape.

IN DETAIL

Most of the time you don't need to be concerned about the shape of the pixels in a video frame. The exceptions are when you mix live-action video with still images prepared in some other way, or export single frames of video to manipulate as still images. By default, bitmapped image editing programs such as Photoshop assume that pixels are square, so that a video frame with non-square pixels will appear to be squashed when you import it into Photoshop. Similarly, a still image will either be stretched when it is treated as a video frame, or it will have black bars down the sides or along the top.

Recent releases of Photoshop are capable of handling images with non-square pixels correctly, but it is necessary to specify the pixel aspect ratio unless the pixels are square.

Video sampled according to CCIR 601 consists of a luminance component and two colour difference components. The colour space is technically $Y'C_B C_R$ (see Chapter 5). It is usually sufficient to consider the three components to be luminance Y , and the differences $B - Y$ and $R - Y$. The values are non-linearly scaled and offset in practice, but this is just a technical detail. The important point to grasp is that the luminance has been separated from the colour differences. As a first step in reducing the size of digital video, this allows fewer samples to be taken for each of the colour difference values as for luminance, a process known as *chrominance sub-sampling*.

As we mentioned in Chapter 5, chrominance sub-sampling is justified by the empirical observation that human eyes are less sensitive to variations in colour than to variations in brightness. The arrangement of samples used in CCIR 601 is called **4:2:2 sampling**; it is illustrated in Figure 6.4. In each line there are twice as many Y samples as there are samples of each of $B - Y$ and $R - Y$.

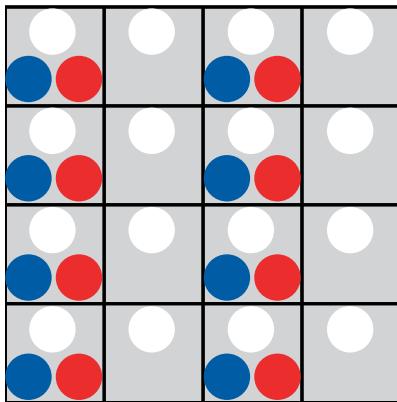


Figure 6.4. 4:2:2 chrominance sub-sampling

The samples are said to be **co-sited**, because both colour differences are sampled at the same points. The resulting data rate for CCIR 601 video, using 8 bits for each component, is 166 Mbits (just over 20 Mbytes) per second, for both PAL and NTSC.

Other sampling arrangements are possible. In particular, as we will see when we consider DV, some standards for digital video employ either 4:1:1 sampling, where only every fourth pixel on each line is sampled for colour, or 4:2:0,[†] where the colour values are not co-sited and are sub-sampled by a factor of 2 in both the horizontal and vertical directions – a somewhat more complex process than it might at first appear, because of interlacing. (4:2:0 is the sub-sampling regime normally used in JPEG compression of still images.)

DV and MPEG

Sampling produces a digital representation of a video signal. This must be compressed and then formed into a data stream for transmission, or stored in a file. Further standards are needed to specify the compression algorithm and the format of the data stream and file. Two separate sets of standards are in use, DV and the **MPEG** family. Both are based on $Y' C_B C_R$ components, scanned according to CCIR 601, but with further chrominance sub-sampling. However, the standards are only part of the story. As we will describe later, codecs and file formats are commonly used which are not defined by official international standards, but are either proprietary or defined by open standards that lack formal status. To complicate matters further, some non-standardized file formats are capable of holding data that has been compressed with standard codecs.

As we remarked earlier, much of the digital video equipment intended for consumer and semi-professional use (such as corporate training video production) and for news-gathering is based on the DV standard, which is relatively limited in its scope. DV and its main variations – DVCAM and DVPRO – all use the same compression algorithm and data stream as DV, which always has a data rate of 25 Mbits (just over 3 Mbytes) per second, corresponding to a compression ratio of 5:1. There are, however, a high-quality DVPRO and a professional Digital-S format, which use 4:2:2 sampling, unlike DV which uses 4:1:1, and offer better quality at correspondingly higher bit rates. These are for professional use. Finally, HDDV is a high-definition version of DV suitable for low-budget film-making.

[†] The notation 4:2:0 is inconsistent; it certainly does not mean that only one of the colour difference values is sampled.

The term “MPEG” encompasses several ISO standards produced by the **ISO/IEC Motion Picture Experts Group**. The earliest standard, MPEG-1, was primarily intended for the Video CD format, but it has provided a basis for subsequent MPEG video standards. Its successor, MPEG-2, is used in the first generation of digital studio equipment, digital broadcast TV and DVD. Subsequent improvements, and a widening of the scope of MPEG, has led to MPEG-4, an ambitious standard designed to support a range of multimedia data at bit rates from as low as 10 kbits per second all the way up to 300 Mbits per second or higher. This allows MPEG-4 to be used in applications ranging from mobile phones to HDTV.

MPEG-4 itself is divided into parts. Some parts are concerned with audio compression, some with delivery of data over a network, some with file formats, and so on. At the time of writing there are 23 parts, although not all of them have been finished and ratified. Parts 2 and 10 deal with video compression. **MPEG-4 Part 2** is what people usually mean when they simply refer to “MPEG-4 video”. It is a refinement of MPEG-2 video, which can achieve better quality at low bit rates (or smaller files of the same quality) by using some extra compression techniques. **MPEG-4 Part 10** describes a further refinement, referred to as **Advanced Video Coding (AVC)**. Because of overlapping areas of responsibility between ISO/IEC and ITU-T, AVC is also an ITU standard, H.264. This has led to a regrettable situation where the same standard is known by four different names: MPEG-4 Part 10, AVC, H.264 and the officially preferred **H.264/AVC**. It has recently emerged as one of the leading compression techniques for Web video and is also used on second generation, high-definition (Blu-Ray) DVDs.

To accommodate a range of requirements, each of the MPEG standards defines a collection of profiles and levels. Each profile defines a set of algorithms that can be used to generate a data stream. In practice, this means that each profile defines a subset of the complete compression technique defined in the standard. Each level defines certain parameters, notably the maximum frame size and data rate, and chrominance sub-sampling. Each profile may be implemented at one or more of the levels, although not every combination of level and profile is defined. For example, the most common combination in MPEG-2 is **Main Profile at Main Level (MP@ML)**, which uses CCIR 601 scanning with 4:2:0 chrominance sub-sampling. This supports a data rate of 15 Mbits per second and allows for the most elaborate representation of compressed data provided by MPEG-2. MP@ML is the format used for digital television broadcasts and for DVD video.

H.264/AVC defines a large and growing set of profiles. Some of these are only of interest for studio and professional use. The profiles most likely to be encountered in multimedia are the **Baseline Profile (BP)**, which is suitable for video-conferencing and mobile devices with limited computing resources; the **Extended Profile (XP)**, which is intended for streaming video; the **Main Profile (MP)**, for general use; and the **High Profile (HIP)**, which is used for HDTV and Blu-Ray. (The Main Profile was originally intended for broadcast use, but has been superseded by HIP.)

The profiles are not subsets of each other: some features supported in the Baseline Profile are not in the Main Profile and vice versa.

290

For each of these profiles, 16 different levels specify the values of parameters such as frame size and bit rate. For example, BP@L1 (level 1 of the Baseline Profile) specifies a bit rate of 64 kbps, for a frame size of 176×144 pixels and frame rate of 15 fps. At the opposite extreme, HP@L5.1 specifies 300 Mbps at 4096×2048 frames and a rate of 30 fps. (The numbering of the levels is not consistent; each level has two or more additional sub-levels, with the sub-level s of level L being written as $L.s$ but level 1 has an additional 1b.)

Although the main contribution of MPEG-4 to digital video lies in its codecs, it also defines a file format, based on the QuickTime format (see below), which can be used to store compressed video data, together with audio and metadata. **MP4** files in this format can be played by many different devices and programs, including the QuickTime and Flash players. The 3GP format used for mobile phones is a simplified version of the MP4 format, which supports video data compressed according to MPEG-4 Part 2 and H.264/AVC, together with audio data.

High Definition Formats

Domestic televisions have been using the same vertical resolution for decades. The first generation of digital video introduced non-square pixels and fixed the number of horizontal samples, but to the viewer, the picture seemed the same size and contained as much (or as little) detail as ever, just less noise. The long-established resolutions for PAL and NTSC frames are referred to as **Standard Definition (SD)** video. HD video is simply anything with larger frames than SD. It was hoped at one time that a global HD standard for broadcast could be agreed, but there are still several to choose from – sometimes different standards are used in a single country. (You may come across “Enhanced Definition”, for example. This generally refers to an SD-sized but progressively scanned frame, written as 480p – see below.)

All the standards agree that the aspect ratio should be 16:9, so the vertical height of the frame is enough to specify the resolution. Two values are in use: 720 and 1080. Each of these might be transmitted at either 25 or (roughly) 30 frames per second, corresponding to the frame rates of the SD standards. Additionally, each HD frame can be transmitted as either a pair of interlaced fields, as we described earlier, or as a single progressively scanned frame. Hence there are eight possible combinations of the different variables. Each one is written as the frame height, followed by the approximate frame rate (for progressive scan) or field rate (for interlaced fields) and a letter i or p, denoting interleaved or progressively scanned, respectively. Thus, for instance, 720 25p would designate a frame size of 1280×720 at a rate of 25 frames per second, progressively scanned, whereas 1080 60i would be a frame size of 1920×1080 , interlaced at 60 fields per second – although in actuality, the field rate would really be 59.94, as in SD NTSC.

HD video requires suitable equipment for capture, transmission, reception, recording and displaying, and it has its own tape formats (including HDCAM, DVCPRO-HD) and optical media (Blu-Ray DVD). However, when it comes to digital processing, the only significant difference between SD and HD video is that the latter uses more bits, so it requires more disk space, bandwidth and processing power. MPEG-2, MPEG-4 Part 2 and H.264/AVC can all have levels at which they can be used to compress HD video. For the most part, therefore, in the rest of this chapter, we will not distinguish between SD and HD.

KEY POINTS

DV camcorders or VTRs connected to computers over FireWire are used for reasonable quality digital video capture.

Cheap video cameras are often built into mobile phones and laptop computers or used as Webcams. They usually use MPEG-4 and USB 2.0.

Digital video standards inherit features from analogue broadcast TV.

Each frame is divided into two fields (odd and even lines), transmitted one after the other and interlaced for display. Interlaced frames may display combing when displayed progressively or exported as still images.

PAL: a frame has 625 lines, of which 576 are picture, displayed at 50 fields (25 frames) per second (625/50). NTSC: a frame has 525 lines, of which 480 are picture, displayed at 59.94 fields (29.97 frames) per second (525/59.94, often treated as 525/60).

CCIR 601 (Rec. ITU-R BT.601) defines standard definition digital video sampling, with 720 luminance samples and 2×360 colour difference samples per line. ($Y'C_B C_R$ with 4:2:2 chrominance sub-sampling.)

PAL frames are 720×576 and NTSC are 720×480. The pixels are not square.

DV applies 4:1:1 chrominance sub-sampling and compresses to a constant data rate of 25 Mbits per second, a compression ratio of 5:1.

MPEG defines a series of standards. MPEG-2 is used on DVDs; MPEG-4 supports a range of multimedia data at bit rates from 10 kbps to 300 Mbps or greater.

MPEG-4 is a multi-part standard. Part 2 defines a video codec; Part 10 (H.264/AVC) is an improved version.

MPEG standards all define a set of profiles (features) and levels (parameters). The Baseline, Extended and Main profiles of H.264/AVC are all used in multimedia.

MPEG-4 defines a file format. 3GP is a simpler version, used in mobile phones.

HD video uses higher resolutions and may be progressively scanned. Frames with widths of 720 and 1080 pixels and an aspect ratio of 16:9 are used.

Video Compression

The input to any video compression algorithm consists of a sequence of bitmapped images (the digitized video). There are two ways in which this sequence can be compressed: each individual image can be compressed in isolation, using the techniques introduced in Chapter 4, or sub-sequences of frames can be compressed by only storing the differences between them. These two techniques are usually called ***spatial compression*** and ***temporal compression***, respectively, although the more accurate terms ***intra-frame*** and ***inter-frame*** compression are also used, especially in the context of MPEG. Spatial and temporal compression are normally used together.

Since spatial compression is just image compression applied to a sequence of bitmapped images, it could in principle use either lossless or lossy methods. Generally, though, lossless methods do not produce sufficiently high compression ratios to reduce video data to manageable proportions, except on synthetically generated material (such as we will consider in Chapter 7), so lossy methods are usually employed. Lossily compressing and recompressing video usually leads to a deterioration in image quality, and should be avoided if possible, but recompression is often unavoidable, since the compressors used for capture are not the most suitable for delivery for multimedia. Furthermore, for post-production work, such as the creation of special effects, or even fairly basic corrections to the footage, it is usually necessary to decompress the video so that changes can be made to the individual pixels of each frame. For this reason it is wise – if you have sufficient disk space – to work with uncompressed video during the post-production phase. That is, once the footage has been captured and selected, decompress it and use uncompressed data while you edit and apply effects, only recompressing the finished product for delivery. (You may have heard that one of the advantages of digital video is that, unlike analogue video, it suffers no “generational loss” when copied, but this is only true for the making of exact copies.)

The principle underlying temporal compression algorithms is simple to grasp. Certain frames in a sequence are designated as ***key frames***. Often, key frames are specified to occur at regular intervals – every sixth frame, for example – which can be chosen when the compressor is invoked. These key frames are either left uncompressed, or more likely, only spatially compressed. Each of the frames between the key frames is replaced by a difference frame, which records only the differences between the frame which was originally in that position and either the most recent key frame or the preceding frame, depending on the sophistication of the decompressor.

For many sequences, the differences will only affect a small part of the frame. For example, Figure 6.5 shows part of two consecutive frames (de-interlaced), and the difference between them, obtained by subtracting corresponding pixel values in each frame. Where the pixels are identical, the result will be zero, which shows as black in the difference frame on the far right. Here, approximately 70% of the frame is black: the land does not move, and although the sea and clouds



Figure 6.5. Frame difference

are in motion, they are not moving fast enough to make a difference between two consecutive frames. Notice also that although the girl's white over-skirt is moving, where part of it moves into a region previously occupied by another part of the same colour, there is no difference between the pixels. The cloak, on the other hand, is not only moving rapidly as she turns, but the shot silk material shimmers as the light on it changes, leading to the complex patterns you see in the corresponding area of the difference frame.

Many types of video footage are composed of large relatively static areas, with just a small proportion of the frame in motion. Each difference frame in a sequence of this character will have much less information in it than a complete frame. This information can therefore be stored in much less space than is required for the complete frame.

IN DETAIL

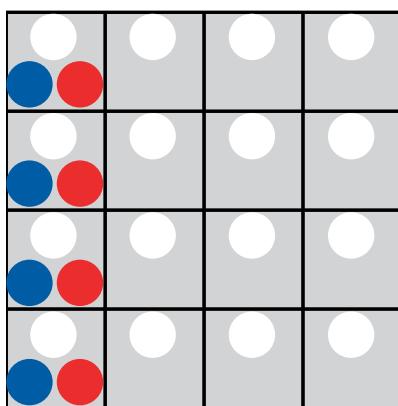
You will notice that we have described these compression techniques in terms of frames. This is because we are normally going to be concerned with video intended for progressively scanned playback on a computer. However, the techniques described can be equally well applied to fields of interlaced video. While this is somewhat more complex, it is conceptually no different.

Compression and decompression of a piece of video need not take the same time. If they do, the codec is said to be *symmetrical*, otherwise it is *asymmetrical*. In theory, this asymmetry could be in either direction, but generally it is taken to mean that compression takes longer – sometimes much longer – than decompression. This is acceptable, except during capture, but since playback must take place at a reasonably fast frame rate, codecs which take much longer to decompress video than to compress it are essentially useless.

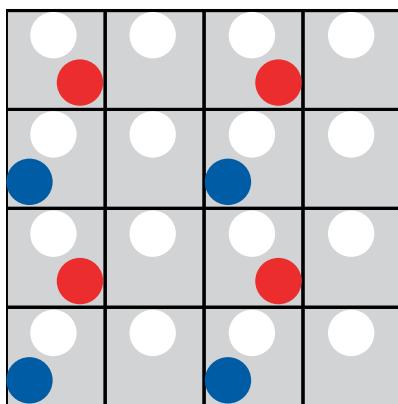
Spatial Compression

The spatial element of many video compression schemes is based, like JPEG image compression, on the use of the Discrete Cosine Transform. The most straightforward approach is to apply JPEG compression to each frame, with no temporal compression. JPEG compression is applied to the three components of a colour image separately, and works the same way irrespective of the colour space used to store image data. Video data is usually stored using $Y'C_BC_R$ colour, with chrominance sub-sampling, as we have seen. JPEG compression can be applied directly to this data, taking advantage of the compression already achieved by this sub-sampling.

The technique of compressing video sequences by applying JPEG compression to each frame is referred to as ***motion JPEG*** or ***MJPEG*** (not to be confused with MPEG) compression, although you should be aware that, whereas JPEG is a standard, MJPEG is only a loosely defined way of referring to this type of video compression. MJPEG was formerly the most common way of compressing video while capturing it from an analogue source, and used to be popular in digital still image cameras that included primitive facilities for capturing video.



Now that analogue video capture is rarely needed, the most important technology that uses spatial compression exclusively is DV. Like MJPEG, DV compression uses the DCT and subsequent quantization to reduce the amount of data in a video stream, but it adds some clever tricks to achieve higher picture quality within a constant data rate of 25 Mbits (3.25 Mbytes) per second than MJPEG would produce at that rate.



DV compression begins with chrominance sub-sampling of a frame with the same dimensions as CCIR 601. Oddly, the sub-sampling regime depends on the video standard (PAL or NTSC) being used. For NTSC (and DVCPRO PAL), 4:1:1 sub-sampling with co-sited sampling is used, but for other PAL DV formats 4:2:0 is used instead. As Figure 6.6 shows, the number of samples of each component in each 4×2 block of pixels is the same. As in still-image JPEG compression, blocks of 8×8 pixels from each frame are transformed using the DCT, and then quantized (with some loss of information) and run-length and Huffman encoded along a zig-zag sequence. There are, however, a couple of additional embellishments to the process.

First, the DCT may be applied to the 64 pixels in each block in one of two ways. If the frame is static, or almost so, with no difference between the picture in each field, the transform is applied to the entire 8×8 block, which comprises alternate lines from the odd and even fields.

Figure 6.6. 4:1:1 (top) and 4:2:0 chrominance sub-sampling

However, if there is a lot of motion, so that the fields differ, the block is split into two 8×4 blocks, each of which is transformed independently. This leads to more efficient compression of frames with motion. The compressor may determine whether there is motion between the frames by using motion compensation (described below under MPEG), or it may compute both versions of the DCT and choose the one with the smaller result. The DV standard does not stipulate how the choice is to be made.

Second, an elaborate process of rearrangement is applied to the blocks making up a complete frame, in order to make best use of the space available for storing coefficients. A DV stream must use exactly 25 Mbits for each second of video; 14 bytes are available for each 8×8 pixel block. For some blocks, whose transformed representation has many zero coefficients, this may be too much, while for others it may be insufficient, requiring data to be discarded. In order to allow the available bytes to be shared between parts of the frame, the coefficients are allocated to bytes, not on a block-by-block basis, but within a larger “video segment”. Each video segment is constructed by systematically taking 8×8 blocks from five different areas of the frame, a process called *shuffling*. The effect of shuffling is to average the amount of detail in each video segment. Without shuffling, parts of the picture with fine detail would have to be compressed more highly than parts with less detail, in order to maintain the uniform bit rate. With shuffling, the detail is, as it were, spread about among the video segments, making efficient compression over the whole picture easier.

As a result of these additional steps in the compression process, DV is able to achieve better picture quality at 25 Mbits per second than MJPEG can achieve at the same data rate.

Temporal Compression

All modern video codecs use temporal compression to achieve either much higher compression ratios, or better quality at the same ratio, relative to DV or MJPEG. Windows Media 9, the Flash Video codecs and the relevant parts of MPEG-4 all employ the same broad principles, which were first expressed systematically in the MPEG-1 standard. Although MPEG-1 has been largely superseded, it still provides a good starting point for understanding the principles of temporal compression which are used in the later standards that have improved on it, so we will begin by describing MPEG-1 compression in some detail, and then indicate how H.264/AVC and other important codecs have enhanced it.

The MPEG-1 standard[†] doesn't actually define a compression algorithm: it defines a data stream syntax and a decompressor, allowing manufacturers to develop different compressors, thereby leaving scope for “competitive advantage in the marketplace”. In practice, the compressor is fairly thoroughly defined implicitly, so we can describe MPEG-1 compression, which combines

[†] ISO/IEC 11172: “Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s.”

temporal compression based on motion compensation with spatial compression based, like JPEG and DV, on quantization and coding of frequency coefficients produced by a discrete cosine transformation of the data.

296

A naïve approach to temporal compression consists of subtracting the value of each pixel in a frame from the corresponding pixel in the previous frame, producing a difference frame, as we did in Figure 6.5. In areas of the picture where there is no change between frames, the result of this subtraction will be zero. If change is localized, difference frames will contain large numbers of zero pixels, and so they will compress well – much better than a complete frame.

This frame differencing has to start somewhere, with frames that are purely spatially (intra-frame) compressed, so they can be used as the basis for subsequent difference frames. In MPEG terminology, such frames are called ***I-pictures***, where I stands for “intra”. Difference frames that use previous frames are called ***P-pictures***, or “predictive pictures”. P-pictures can be based on an earlier I-picture or P-picture – that is, differences can be cumulative.

Often, though, we may be able to do better, because pictures are composed of objects that move as a whole: a person might walk along a street, a football might be kicked, or the camera might pan across a landscape with trees. Figure 6.7 is a schematic illustration of this sort of motion, to demonstrate how it affects compression. In the two frames shown here, the fish swims from left to right. Pixels therefore change in the region originally occupied by the fish – where the background becomes visible in the second frame – and in the region to which the fish moves. The black area in the picture at the bottom left of Figure 6.7 shows the changed area which would have to be stored explicitly in a difference frame.

However, the values for the pixels in the area occupied by the fish in the second frame are all there in the first frame, in the fish’s old position. If we could somehow identify the coherent area corresponding to the fish, we would only need to record its displacement together with the changed pixels in the smaller area shown at the bottom right of Figure 6.7. (The bits of weed and background in this region are not present in the first frame anywhere, unlike the fish.) This technique of incorporating a record of the relative displacement of objects in the difference frames is called ***motion compensation*** (also known as ***motion estimation***). Of course, it is now necessary to store the displacement as part of the compressed file. This information can be recorded as a ***displacement vector***, giving the number of pixels the object has moved in each direction.

If we were considering some frames of video shot under water showing a real fish swimming among weeds (or a realistic animation of such a scene) instead of these schematic pictures, the objects and their movements would be less simple than they appear in Figure 6.7. The fish’s body would change shape as it propelled itself, the lighting would alter, the weeds would not stay still.

Attempting to identify the objects in a real scene and apply motion compensation to them would not work, therefore (even if it were practical to identify objects in such a scene).

MPEG-1 compressors do not attempt to identify discrete objects in the way that a human viewer would. Instead, they divide each frame into blocks of 16×16 pixels known as **macroblocks** (to distinguish them from the smaller blocks used in the DCT phase of compression), and attempt to predict the whereabouts of the corresponding macroblock in the next frame. No high-powered artificial intelligence is used in this prediction: all possible displacements within a limited range are tried, and the best match is chosen. The difference frame is then constructed by subtracting each macroblock from its predicted counterpart, which should result in fewer non-zero pixels, and a smaller difference frame after spatial compression.

The price to be paid for the additional compression resulting from the use of motion compensation is that, in addition to the difference frame, we now have to keep a record of the motion vectors describing the predicted displacement of macroblocks between frames. These can be stored relatively efficiently, however. The motion vector for a macroblock is likely to be similar

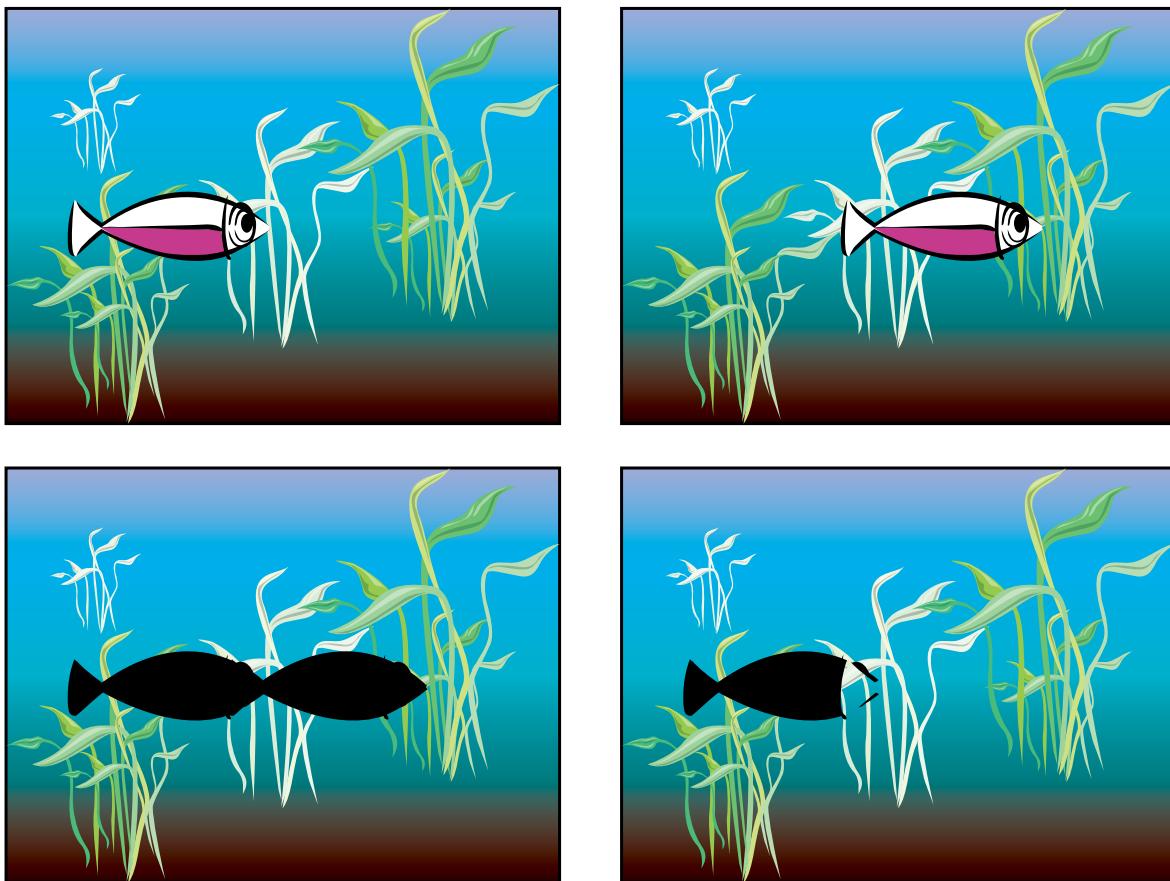


Figure 6.7. Motion compensation

or identical to the motion vector for adjoining macroblocks (since these will often be parts of the same object), so, by storing the differences between motion vectors, additional compression, analogous to inter-frame compression, is achieved.

298

Although basing difference frames on preceding frames probably seems the obvious thing to do, it can be more effective to base them on following frames. Figure 6.8 shows why such backward

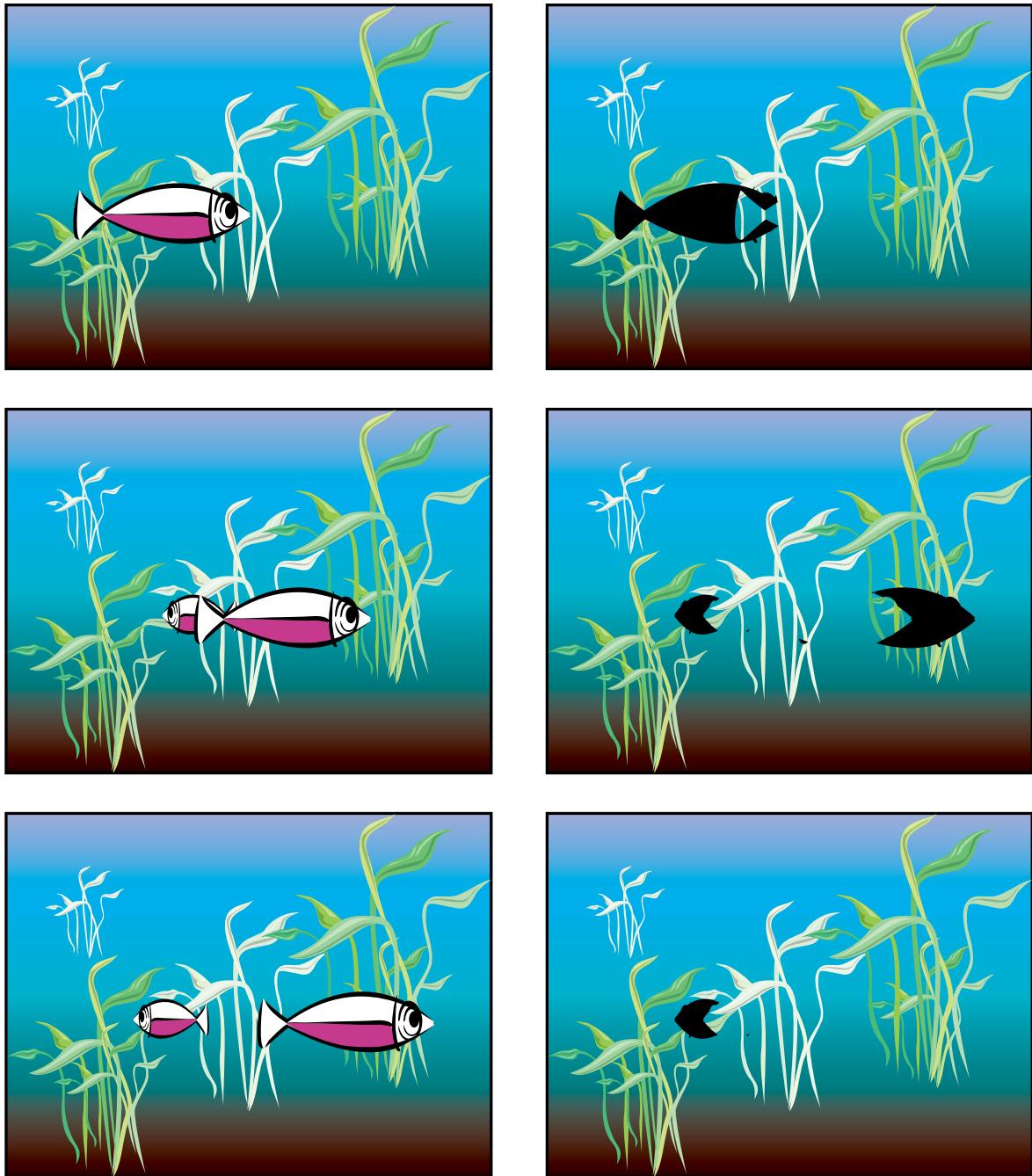


Figure 6.8. *Bi-directional prediction*

prediction can be useful. In the top frame, the smaller fish that is partially revealed in the middle frame is hidden, but it is fully visible in the bottom frame. If we construct an I-picture from the first two frames, it must explicitly record the area covered by the fish in the first frame but not the second, as before. If we construct the I-picture by working backwards from the third frame instead, the area that must be recorded consists of the parts of the frame covered up by either of the fish in the third frame but not in the second. Motion compensation allows us to fill in the bodies of both fish in the I-picture. The resulting area, shown in the middle of the right-hand column of Figure 6.8, is slightly smaller than the one shown at the top right. If we could use information from both the first and third frames in constructing the I-picture for the middle frame, almost no pixels would need to be represented explicitly, as shown at the bottom right. This comprises the small area of background that is covered by the big fish in the first frame and the small fish in the last frame, excluding the small fish in the middle frame, which is represented by motion compensation from the following frame. To take advantage of information in both preceding and following frames, MPEG compression allows for **B-pictures**, which can use motion compensation from the previous or next I- or P-pictures, or both, hence their full name “bi-directionally predictive” pictures.

A video sequence can be encoded in compressed form as a sequence of I-, P- and B-pictures. It is not a requirement that this sequence be regular, but encoders typically use a repeating sequence, known as a **Group of Pictures** or **GOP**, which always begins with an I-picture. Figure 6.9 shows a typical example. (You should read it from left to right.) The GOP sequence is IBBPBB. The diagram shows two such groups: frames 01 to 06 and frames 11 to 16. The arrows indicate the forward and bi-directional prediction. For example, the P-picture 04 depends on the I-picture 01 at the start of its GOP; the B-pictures 05 and 06 depend on the preceding P-picture 04 and the following I-picture 11.

All three types of picture are compressed using the MPEG-1 DCT-based compression method. Published measurements indicate that, typically, P-pictures compress three times as much as I-pictures, and B-pictures one and a half times as much as P-pictures. However, reconstructing B-pictures is more complex than reconstructing the other types, so there is a trade-off to be made between compression and computational complexity when choosing the pattern of a GOP.

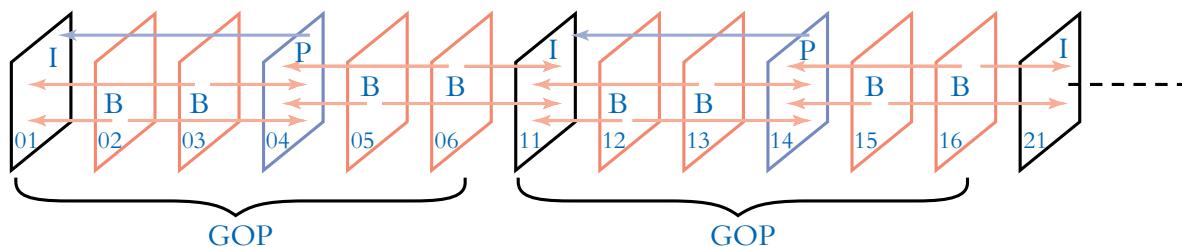


Figure 6.9. An MPEG sequence in display order

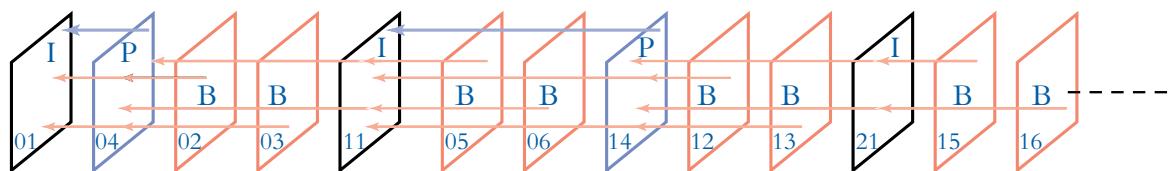


Figure 6.10. An MPEG sequence in bitstream order

An additional factor is that random access to frames corresponding to B- and P-pictures is difficult, so it is customary to include I-pictures sufficiently often to allow random access to several frames each second. Popular GOP patterns include **IBBPBBPBB** and **IBBPBBPBBPBB**. However, as we remarked, the MPEG-1 specification does not require the sequence of pictures to form a regular pattern, and sophisticated encoders will adjust the frequency of I-pictures in response to the nature of the video stream being compressed.

For the decoder, there is an obvious problem with B-pictures: some of the information required to reconstruct the corresponding frame is contained in an I- or P-picture that comes later in the sequence. This problem is solved by reordering the sequence. The sequence of pictures corresponding to the actual order of frames is said to be in “display order”; it must be rearranged into a suitable “bitstream order” for transmission. Figure 6.10 shows the bitstream order of the sequence shown in display order in Figure 6.9. All the arrows showing prediction now run from right to left, i.e. every predicted frame comes later in the sequence than the pictures it depends on.[†] You will notice that the first GOP is reordered differently from the second; any subsequent groups will extend the pattern established by the second.

Before any of this compression is done, MPEG-1 video data is chroma sub-sampled to 4:2:0. If, in addition to this, the frame size is restricted to 352×240 , video at a frame rate of 30 fps can be compressed to a data rate of 1.86 Mbits per second – the data rate specified for compact disc video. 4:2:0 video of this size is said to be in **Source Input Format (SIF)**. SIF is the typical format for MPEG-1 video, although it can be used with larger frame sizes and other frame rates. MPEG-1 cannot, however, handle interlacing or HDTV formats, hence the need for MPEG-2 for broadcasting and studio work.

The preceding description should have made it clear that MPEG compression and decompression are computationally expensive tasks – and there are further complications which we have glossed over. Initially, MPEG video could only be played back using dedicated hardware. Indeed, the parameters used for CD video were chosen largely so that MPEG decoders could be accommodated in VLSI chips at the time the standard was drawn up (1993). Advances in processor speed

[†] For the B-pictures, we have run the arrows to the relevant P- and I-pictures together, with an intermediate arrowhead, in an attempt to keep the diagram less cluttered.

mean that it has since become feasible to play back MPEG-1 video using software only. File sizes are by no means small, however. A 650 Mbyte CD-ROM will only hold just over 40 minutes of video at that rate; an 8.75 Gbyte DVD has room for over nine hours. (You would only use MPEG-1 on DVD if you were just using the disk as a storage medium, though. DVDs employ MPEG-2 when they are Digital Video Disks, for playing in domestic DVD players.)

MPEG-4 and H.264/AVC

MPEG-4 is an ambitious standard, which defines an encoding for multimedia streams made up of different types of object – video, still images, animation, textures, 3-D models, and more – and provides a way of composing scenes at the receiving end from separately transmitted representations of objects. The idea is that each type of object will be represented in an optimal fashion, rather than all being composited into a sequence of video frames. Not only should this allow greater compression to be achieved, it also makes interaction with the resulting scene easier, since the objects retain their own identities.

At the time of writing, however, it is the video and audio codecs described in the MPEG-4 standard which have received the most attention, and for which commercial implementations exist. We will look at audio compression in Chapter 8, and only consider video here, beginning with the older MPEG-4 Part 2.

As we remarked earlier, MPEG standards define a collection of profiles for video data. The higher profiles of MPEG-4 Part 2 employ a method of dividing a scene into arbitrarily shaped video objects – for example a singer and the backdrop against which she is performing – which can be compressed separately. The best method of compressing the background may not be the same as the best method of compressing the figure, so by separating the two, the overall compression efficiency can be increased. However, dividing a scene into objects is a non-trivial exercise, so the lower profiles – *Simple Profile* and *Advanced Simple Profile* – are restricted to rectangular objects, in particular complete frames, and it is these profiles which have been implemented in widely used systems such as QuickTime and DivX (see below). For practical purposes, therefore, MPEG-4 Part 2 video compression is a conventional, frame-based codec, which is a refinement of the MPEG-1 codec just described. I-pictures are compressed by quantizing and Huffman coding DCT coefficients, but some improvements to the motion compensation phase used to generate P- and B-pictures provide better picture quality at the same bit rates, or the same quality at lower bit rates, as MPEG-1.

The Simple Profile uses only P-pictures (those that depend only on earlier pictures) for inter-frame compression. This means that decompression can be more efficient than with the more elaborate schemes that use B-pictures (which may depend on following pictures), so the Simple

Profile is suitable for implementation in devices such as PDAs and portable video players. The Advanced Simple Profile adds B-pictures and a couple of other features.

302

Global Motion Compensation is an additional technique that is effective for compressing static scenes with conventional camera movements, such as pans and zooms. The movement can be modelled as a vector transformation of the original scene, and represented by the values of just a few parameters. **Sub-pixel motion compensation** means that the displacement vectors record movement to an accuracy finer than a single pixel – in the case of Simple Profile, half a pixel, and for the Advanced Simple Profile, a quarter of a pixel. This prevents errors accumulating, resulting in better picture quality with little additional overhead.

H.264/AVC is an aggressively optimized version of MPEG-4 Part 2. It is one of three codecs which all Blu-Ray players must implement. (The others are MPEG-2, for compatibility with older DVDs, and VC-1, discussed below.) It is routinely claimed that “H.264 can match the best possible MPEG-2 quality at up to half the data rate”. Among other refinements contributing to this improved performance, H.264/AVC allows the use of different-sized blocks for motion compensation, so that areas with little change can be encoded efficiently using large blocks (up to 16×16 pixels), but areas that do change can be broken into smaller blocks (down to 4×4 pixels), which is more likely to result in compression, while preserving the picture quality in fast-moving parts of the frame. Additionally, whereas MPEG-4 Part 2, like MPEG-1, only allows difference frames to depend on at most one preceding and one following frame, H.264/AVC allows data from a stack of frames anywhere in a movie to be used. (The whole movie thus becomes a source of blocks of pixels, which can be reused. This is somewhat similar to the dictionary-based approach to compression found in the LZ algorithms we mentioned in Chapter 4.) B-frames may even depend on other B-frames.

H.264/AVC takes the same approach as JPEG and the other MPEG video codecs to compressing the individual I-, P- and B-frames – transforming them to the frequency domain, quantizing and compressing the coefficients losslessly – but it improves all three elements of the process. It uses a better transform than DCT, with a choice of 8×8 or 4×4 blocks, logarithmic quantization, and employs a mixture of lossless algorithms for compressing the coefficients, which can take account of context, and between them work more efficiently than Huffman coding. H.264/AVC also incorporates filters for removing some compression artefacts, which result in better picture quality. In particular, a “de-blocking filter” is used to smooth the characteristic discontinuities between the blocks of pixels that are transformed separately.

Some aspects of H.264/AVC compression require more than one pass to be made over the data. This is not practical for live video, and may be too slow for creating rough previews, so codecs typically offer a single-pass mode for occasions when the video has to be compressed as quickly

as possible. Single-pass coding is faster but does not produce such good results as the multi-pass mode, which is required if the best results are to be obtained.

Other Video Codecs

Two other video codecs are of considerable practical importance: Windows Media 9 and the On2 V6 codec used for Flash Video.

303

Windows Media is a proprietary technology, developed by Microsoft. Its video codec has evolved over the years, with the latest version, WMV 9, incorporating many of the same ideas as H.264/AVC, including bi-directional prediction (B-pictures), motion compensation and a de-blocking filter. A significant difference is that WMV 9 supports “differential quantization”, which means that different quantization matrices can be used on different parts of a frame. Generally, only two matrices are used, one for simple areas and another for more complex ones. WMV 9 can also apply its DCT to each 8×8 block of pixels as a whole in the conventional way, or break it into two 8×4 blocks, two 4×8 blocks, or four 4×4 transforms. These smaller transform blocks can reduce the visible artefacts at block edges that are typical of DCT-based compression.

A somewhat specialized optimization is that fade transitions (see below) are treated specially. Normally, these transitions are difficult to compress, because every single pixel will change in each frame over the duration of the fade. By detecting fades and treating them as a special case, WMV 9 is able to achieve extra compression. Fades are probably the most common transitions after straight cuts, so this will often be a worthwhile optimization.

The WMV-9 codec has been standardized by the Society of Motion Picture Engineers (SMPTE), under the name **VC-1**. In this guise, it is mandatory for Blu-Ray players. Like the MPEG codecs, VC-1 has several profiles and levels, which cover applications ranging from low bit-rate network video up to 1080p HD video. Subjectively, the quality of VC-1 is at least as good as H.264/AVC, as you would expect given the similarities between the two.

The On2 VP6 codec achieved widespread use when it was adopted for use in Flash Video at the time that format became popular on the Web. Unlike the other codecs we have looked at, On2 VP6 is purely proprietary, and is not defined by an official standard. Instead, it is protected by copyright, and technical details are scarce. It appears to be another DCT-based technique, with inter-frame compression and motion compensation. Unlike the other codecs, it does not support bi-directional prediction: P-pictures can only depend on P- and I-pictures that precede them.

One advantage that is claimed for the On2 VP6 codec is that it is said to be relatively simple to decompress video that has been compressed with it.

On2 VP6 is one of a series of VPx codecs created by On2 Technologies. On2 VP3 has special significance: On2 Technologies granted a licence to an organization called the Xiph Foundation for its free use for any purpose. Xiph Foundation used VP3 as the basis of the Open Source **Ogg Theora** codec, which is free to use for any purpose, unlike all the other codecs described, which are subject to licence fees for some purposes. As a result, Ogg Theora is extensively documented.

Like all the codecs we have described, Theora uses a JPEG-like lossy compression algorithm based on a Discrete Cosine Transform followed by quantization, coupled with inter-frame compression with motion compensation. The DCT is applied to 8×8 blocks of pixels, as usual. Only I- and P-pictures are supported; there is no bi-directional prediction. In other words, Theora lacks most of the refinements present in other popular codecs. The present version cannot handle interlaced video either. Its main interest lies in its Open Source status, not in its technology.

IN DETAIL

Video compression is presently dominated by DCT-based methods. Some work is being done on applying wavelet compression to video. The only standardized wavelet-based format in use is Motion JPEG 2000, which is simply JPEG 2000, as described in Chapter 4, applied to sequences of frames, with no inter-frame compression. It is therefore only suitable for specialized applications, the most important of which is digital cinema. Apple's Pixlet codec is similar: it too does no inter-frame compression and is intended for use by film-makers.

Dirac is an Open Source codec, originally developed by the BBC's R&D department, which does combine wavelet compression with inter-frame compression and motion compensation. It is still at an early stage of development, but it seems likely that it will grow into a significant alternative to H.264/AVC and other DCT-based codecs.

Quality

It is natural to ask “Which codec is best?”, but the question does not admit a simple answer. Usually, “best” means producing the best picture quality at a particular bit rate (or the highest compression ratio for the same quality). However, sometimes the speed of compression, the complexity of decompression, or the availability of software capable of playing back video compressed with a particular codec may be of more practical importance than its compression performance.

The parameters which each codec provides for varying the quality are not the same, so it is not easy to compare codecs directly. Some restrict you to particular sets of parameters, others let you specify maximum bit rates, others provide a numerical quality setting, some allow you to select a profile, while others allow you control over all these values. The way in which they interact is not always clear.



Figure 6.11. Compressed video at high quality

The quality of compressed video at a particular bit rate produced by each codec will vary with the nature of the source video as well as with the parameters to the compression. In any case, judgements of quality are subjective.

Despite these reservations, Figure 6.11 demonstrates that all of the leading codecs are capable of producing compressed video which is barely distinguishable from a DV original when their parameters are set to produce full-frame video at a bit rate of roughly 2 Mbps. As we showed earlier in the chapter, the DV frame already shows some compression artefacts, but it serves as an appropriate reference point, since it was the format in which the footage was captured, and is thus the best quality attainable in this case. There is a fairly subtle colour shift on the H.264/AVC sample, but otherwise even the inset details, which are considerably blown up, are hard to distinguish from one another. Only the On2 VP6 sample shows any appreciable artefaction.



Figure 6.12. Over-compression with H.264/AVC (top) and On2 VP6 (bottom)

For studio-quality source material you would use higher rates, but 2 Mbps will be a reasonable bit rate for multimedia video, so the choice of codec will depend on the other factors just outlined. For instance, despite its excellent quality, WMV 9 can be problematic on systems other than Windows, so to maximize compatibility you might prefer to use H.264/AVC, which can be played on any platform.

It can be instructive to look at what happens if the compression ratio is driven to unreasonable extremes. The top set of illustrations in Figure 6.12 show our example frame as it appears in a version of the clip compressed with H.264/AVC to a rate of only 256 kbps, at its full size and frame rate. The parameters lie outside any level of the standard, so this is not something you would normally do – it should be obvious why not. What is interesting is the way in which the moving figure has broken up very badly, while the relatively static background still retains much of its original quality. In the inset detail of the figure, notice the blurry appearance, presumably caused by the de-blocking filter. In contrast, the version below, compressed to roughly the same size with

On2 VP6, is characterized by a “blocky” over-sharpened appearance, in both the moving figure and the static background. When the movies are actually played, there are more intrusive sudden changes in the background of the On2 VP6 version, but a much greater loss of detail in the H.264/AVC version. Neither is acceptable. If this sort of distortion is occurring you should either increase the target bit rate, if your codec permits it, or reduce the frame size, frame rate or both.

307

KEY POINTS

Spatial (intra-frame) compression and temporal (inter-frame) compression are used together in most contemporary video codecs.

Chrominance sub-sampling is nearly always applied before any compression.

Spatial compression of individual video frames is usually based on a Discrete Cosine Transformation, like JPEG.

DV compression is purely spatial. It extends the JPEG technique by using a choice of sizes for transform blocks, and by shuffling, to even out change across a frame.

Temporal compression works by computing the difference between frames instead of storing every one in full.

In MPEG terminology, I-pictures are only spatially compressed. P-pictures are computed from a preceding I- or P-picture.

Motion compensation is the technique of incorporating a record of the relative displacement of objects in the difference frames, as a motion vector.

In existing codecs, motion compensation is applied to macroblocks, since coherent objects cannot usually be identified.

B-pictures use following pictures as well as preceding ones as the basis of frame differences and motion compensation.

A video sequence is encoded as a Group of Pictures (GOP). If B-pictures are used, a GOP may have to be reordered into display order for decoding.

MPEG-4 Part 2 uses global motion compensation and sub-pixel motion compensation to improve on the quality of MPEG-1 and MPEG-2.

H.264/AVC adds several extra techniques, including variable-sized transform blocks and macroblocks, and a de-blocking filter, to make further improvements.

Windows Media 9 (standardized as VC-1) incorporates similar improvements.

On2 VP6 and Ogg Theora are less powerful, but widely or freely available.

All modern codecs produce excellent quality at 2 Mbps and higher.

Editing and Post-Production

Any video production must begin with the shooting of some footage. It is not the purpose of this book to teach you how to be a film director, so we won't offer any advice about the shooting, composition, lighting, camera work or any other part of the production. We will assume that you have already shot or acquired some properly lit action taking place in front of a camera, which has been recorded on tape (or even DVD), or on the internal disk of a video camera.

With modern equipment, capturing video from a camera or tape deck is simple. (If you are working from tape it is best to use a tape deck for this process if possible – tape transports in camcorders don't always withstand much winding and rewinding.) Recording to computer disk from a DV device is usually just a matter of connecting the device to the computer using a FireWire cable, starting up some software that can perform capture, selecting the standard to be used (PAL or NTSC) and clicking a button. The software in question can be a simple utility that does nothing but capture video, a consumer-oriented video application which also provides rudimentary editing facilities, such as iMovie or Windows Movie Maker, or a professional or semi-professional program, such as Final Cut Pro or Premiere, which provide capture as part of a comprehensive set of editing and post-production facilities. In each case, the operation is broadly similar. The more sophisticated programs will take advantage of the device control facilities of DV to allow you to start and stop the tape or move to a specific point before beginning the capture.

Shooting and recording video only provides raw material. Creating a finished video movie – whether it is a feature film or a small clip for a Web site – requires additional work. Editing is the process of constructing a whole movie from a collection of parts or clips. It comprises the selection, trimming and organization of the raw footage and – where sound is used – the synchronization of sound with picture. Transitions, such as dissolves, may be applied between shots, but – at the editing stage – no changes are made to the footage itself. We contrast this with post-production, which is concerned with altering or adding to the original material. Many of the changes made at this stage are generalizations of the image manipulation operations we described in Chapter 4, such as colour and contrast corrections, blurring or sharpening, and so on. Compositing – the combination or overlaying of elements from different shots into one composite sequence – is often carried out during post-production. Figures may be inserted into background scenes that were shot separately, for example. Elements may be animated during post-production, and animation may be combined with live action in order to create special effects.

Even if nobody wanted to display it on a computer, send it over a network or broadcast it digitally, video would still be digitized, because the advantages of digital non-linear editing are too compelling to resist. To appreciate this, and to understand the metaphors commonly used by digital editing programs, we have briefly to consider traditional methods of film and video editing.

Traditional Film and Video Editing

Editing film is a physical process. The easiest way to rearrange film is by actually cutting it – that is, physically dividing a strip of film into two clips which may then be spliced together with other clips to compose a scene. When the film is projected, the resulting transition between shots or scenes is the familiar “cut” (the splice itself does not show). A cut produces an abrupt discontinuity in the action on screen, but film audiences have become so accustomed to such jumps that they are accepted as part of the story-telling process in the medium.

Although making straight cuts in film is straightforward, creating other types of transition between clips – such as dissolves and wipes – is much less so, and before the digital era it usually required the use of a device called an *optical printer*. There are several types of optical printer; the simplest to understand comprises a rig that directs the light from a pair of projectors into a camera. Optical filters and masks can be interposed to control the amount of light from each projector reaching the camera. The picture which the camera records can thus be a combination of the pictures on the two original clips, with the filters and so on applied, as shown schematically in Figure 6.13. The result of creating an effect in the optical printer is a new piece of film which can then be spliced into the whole.

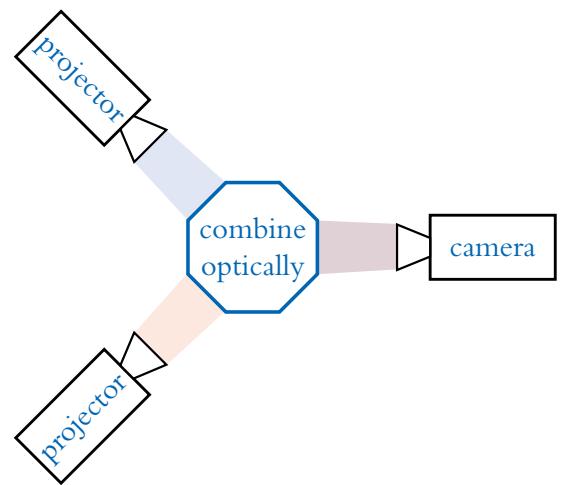


Figure 6.13. *Optical printing*

Despite the apparent simplicity of the set-up, exceptionally sophisticated effects can be achieved using such “opticals”, in conjunction with techniques such as matte painting or the use of models. Many famous films of the twentieth century used optical printing to achieve magical special effects. One drawback is that opticals are usually done by a specialist laboratory, so the film editor and director cannot actually see what the transition looks like until the resulting film has been developed. This leaves little room for experimentation. It is no coincidence that the straight cut formed the basis of most films’ structure, especially when the budget was limited.

Traditional analogue video editing, although the same as film editing in principle, was quite different in practice. It is virtually impossible to cut video tape accurately, or splice it together, without destroying it. Before digital video, therefore, the only way to rearrange pictures recorded on analogue video tape was to use more than one tape deck and copy selected parts of a tape from one machine onto a new tape on another, in the desired order. It was necessary to wind and rewind the source tape to find the beginning and end points of scenes to be included. Very simple editing could be carried out with just two tape decks, but a more powerful (and more common) arrangement was to use three machines, so that scenes on two separate tapes could be combined

onto a third. (This setup was known as a three-machine edit suite.) This arrangement closely resembles an optical printer, but electronic signals are combined instead of light, so only effects that can easily be achieved using electronic circuits can be used. A rich variety of transitions could be produced this way, and – unlike film transitions – they could be reviewed straight away, and parameters such as the speed of a dissolve could be controlled in real time. With this arrangement, straight cuts were not significantly easier to make than any other transition, but they were still the predominant transition because of established film-making convention.

This method of editing required some means of accurately identifying positions on tapes. **Timecode** was devised for this purpose. There are several timecode standards in use, but the only one of any importance is **SMPTE timecode**. A timecode value consists of four pairs of digits separated by colons – such as 01:14:35:06 – representing hours, minutes, seconds and frames, so that the complete value identifies a precise frame. It might seem like a trivially obvious scheme, but the tricky bit was writing the code onto the video tape so that its current frame could be read by a machine. Standards for doing so were developed, and so “frame-accurate” positioning of tape was made possible.

IN DETAIL

Timecode behaves differently depending on the frame rate. For a PAL system, the final component (which identifies the frame number) ranges from 0 to 24, for NTSC it ranges from 0 to 29, but not in the obvious way, because the NTSC frame rate is 29.97.

Since there is not an exact number of NTSC frames in a second, SMPTE timecode, which must use exactly 30, drifts with respect to the elapsed time. The expedient adopted to work round this is called “drop frame timecode”, in which frames 00:00 and 00:01 are omitted at the start of every minute except the tenth. (It’s a bit like a leap year.) So your count jumps from, say, 00:00:59:29 to 00:01:00:02, but runs smoothly from 00:09:59:29 through 00:10:00:00 to 00:10:00:01. The correct handling of drop frame timecode is one measure of how professional a digital video editing program is.

Digital Video Editing

Now that digital video is widely used, almost all video editing is being done on computers, where the non-linear working mode of film editing can be applied to the digital data representing video sequences. Video editing is therefore now closer in kind to film editing, but without the physically destructive process. An imperfect (but useful) analogy of the difference between linear analogue and non-linear digital video editing is the difference between writing with a typewriter and using a word processor. On a traditional typewriter, words have to be written in their final order, with the potential for corrections limited to what can be achieved with correction fluid. When things

go wrong or sections need rewriting, entire sheets of paper have to be thrown away and retyped – which may upset subsequent pagination, in turn requiring even more retyping. Similarly, when analogue video tape was edited, the signals had to be recorded in their final order, and the order could only be changed by rewriting to a new tape. Once the edit was written to the new tape it couldn't be changed – except by over-writing or discarding the tape and starting again.

When you use a word processor instead of a typewriter, however, a potentially infinite number of corrections can be made anywhere in the text at any time, and composition can be written in any order, without regard to pagination or layout – and without throwing anything away and starting again. In the same way, digital video editing software allows scenes to be rearranged and changed just by dragging a representation of the video in an editing window and applying some instructions. Most importantly, it is non-destructive – a huge advantage over pre-digital editing techniques. In film editing the film itself had to be cut up and much of the footage was literally thrown away (some valuable scenes were lost on the cutting room floor), and in analogue video editing the picture had to be copied onto new tape and the original tapes played over and over again. This resulted in degradation of picture quality and eventually of the physical material of the source tape itself. In digital video editing, however, the source clips need never be altered or damaged. It is possible to cut and recut, potentially forever, as the editor changes his or her mind, without any alteration to the original material.

Furthermore – in stark contrast to film – edited digital video can be played back as soon as the hardware on which it is being edited allows. With top-end equipment, playback is instantaneous. On desktop machines there may be some delay, but the delays are measured in minutes – or hours at worst – not the days that it may take for film to be processed. Recent advances in hardware and software mean that now even desktop editing systems often provide instant playback of edited digital video.

Generally, digital video formats are designed to facilitate editing and minimize the need for recompression. For instance, the QuickTime file format (and hence the MPEG-4 file format) separates the media data – the bits representing the actual pictures – from track data – descriptions of how the media data should be played back. Some editing operations can be implemented by changing the track data without altering the media data. For example, a video clip can be “trimmed” by changing the track data to record the point in the clip where it should start to play. In these cases, when the edited video is exported as a complete movie it need not be recompressed (unless it is being exported to a different format, for example for the Web). This means that there will be no loss of picture quality at all.

However, where transitions are used which depend on combining data from two or more video clips, it is necessary to create new frames – in the same way as it is in an optical printer – so that

although the source clips themselves are not destroyed, the new frames will not be of quite the same quality as the original source material. Creating composited frames requires decompression before they are combined and recompression when they are exported.

312

People develop their own methods of working with a particular program, but the facilities provided by different editing applications are basically the same. One simple, idealized procedure for editing with a desktop application would begin with assembling all the clips for a project – capturing them where necessary, and importing them into a library, where they may be arranged for convenient access.

Next, each clip is opened within the application, and roughly trimmed to remove such extraneous matter as the clapper board or obviously excess footage. A frame is designated as the clip's **in point**, that is, the frame where it should begin, and another as its **out point**, the frame where it should end. Trimming digital video does not discard any frames, it merely suppresses those before the in point and after the out point by adjusting track data. If necessary, the in and out points can be readjusted later. If the out point is subsequently moved to a later frame in the clip, or the in point is moved to an earlier one, frames between the old and new points will reappear.

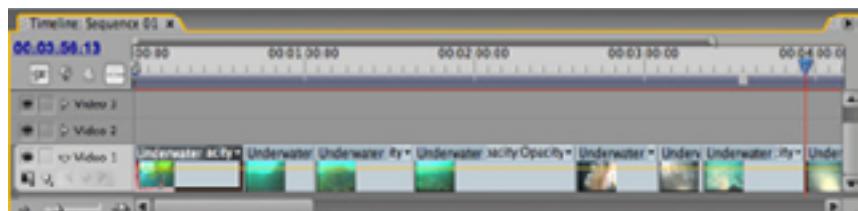


Figure 6.14. The timeline in Premiere

The next step is to arrange clips in the desired order on a **timeline**, as shown in Figure 6.14. The timeline provides a convenient spatial representation of the way frames are arranged in time. (The timeline reads from left to right.)

Still images can also be placed on

the timeline and assigned an arbitrary duration; they will behave as clips with no motion. If the movie is to have a soundtrack, the picture and sound can be combined on the timeline. Often, adjustments will have to be made, particularly if it is necessary to synchronize the sound with the picture. Clips may need to be trimmed again, or more drastic changes may be required, such as the substitution of completely different material when ideas fail to work out. For some basic projects, editing will then be complete at this stage, but more extended or elaborate movies will probably require some more complex transitions, as well as corrections or compositing.



Figure 6.15. A dissolve

Using other types of transition changes the style, rhythm and mood of a piece. A dissolve, for example – in which one clip fades into another – is less emphatic than a cut, and tends to convey a sense of gradual change or smooth flow from one thing to another. It may be used to change location between scenes, or in a more imaginative way – for example, extended dissolves are sometimes used to introduce “dream sequences” in movies. In Figure 6.15 the picture dissolves from the shot looking over the outside of a house to the figure standing by the sea, which in the context of the movie also conveys a more subtle change of circumstance. A dissolve to black (a “fade-out”) and then back from black into a new scene (a “fade-in”) is frequently used to indicate that time has elapsed between the end of the first scene and the beginning of the second.

As most transitions can be described relatively easily in terms of mathematical operations on the two clips involved, digital video editing software usually offers a vast range of possibilities – some video editing applications have well over 50 transitions built in – but many of them are showy gimmicks which are usually best avoided. The more fanciful transitions, such as wipes, spins and page turns, draw attention to themselves and therefore function almost as decoration.

There are two important practical differences between cuts and other transitions. Firstly, in a cut the two clips are butted, whereas in all other transitions they overlap, so that some part of each clip contributes to the resulting picture, as illustrated in Figure 6.16. (Some editing software will display the clips overlapping in this way on the timeline, but other programs will not.) It is therefore necessary to ensure that each clip is shot with enough frames to cover the full duration of the transition in addition to the time it plays on its own.

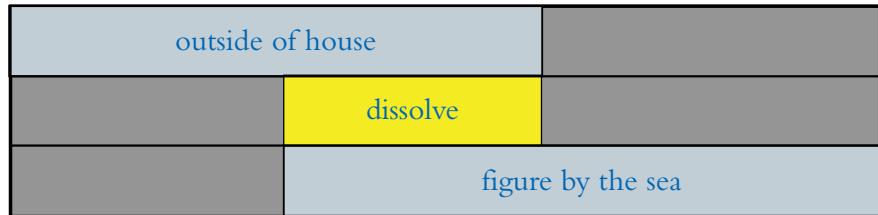
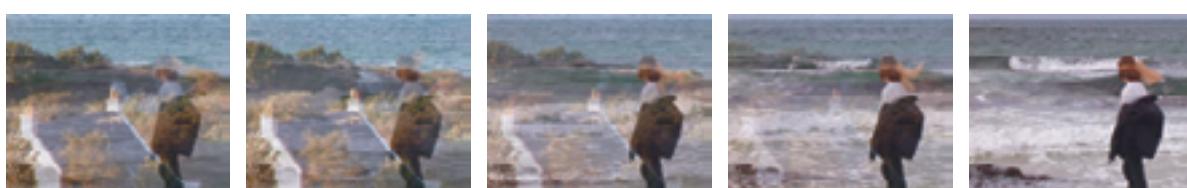


Figure 6.16. Overlapping clips for a transition

Secondly, because image processing is required to construct the transitional frames, transitions must be rendered, unlike cuts, which can be implemented simply by copying. Hence, as we mentioned before, there will inevitably be some loss of image quality where dissolves and other transitions are used instead of straight cuts, though in practice this may not be readily perceptible by the viewer.



Post-Production

Most digital video post-production tasks can be seen as applications of the image manipulation operations we described in Chapter 4 to the bitmapped images that make up a video sequence. Contemporary video editing applications which include post-production facilities normally describe them in the same terms as those used when dealing with single bitmapped still images.

As the raw footage of a video sequence is just a series of photographs, it may suffer from the same defects as a single photograph. For example, it may be incorrectly exposed or out of focus, it may have a colour cast, or it may display unacceptable digitization artefacts. Each of these problems can be remedied in the same way as we would correct a bitmapped image in an application such as Photoshop – for example, we may adjust the levels, sharpen the image, or apply a Gaussian blur (see Chapter 4). Post-production systems therefore provide the same set of adjustments as image manipulation programs – some even support the use of Photoshop plug-ins – but they allow these adjustments to be applied to whole sequences of images. Like Photoshop effects, video effects can be used to create artificial images as well as to correct faulty real ones, and the added time dimension also allows some special effects to be created.

Most adjustments have parameters, such as the slider positions for levels controls (see Chapter 4). When adjustments are made to video sequences, it is possible to choose whether to use the same parameter values for each image in the sequence, or to vary the values in order to create adjustments or effects which change over time.

If, for example, a complete sequence has been shot under incorrect lighting, the same correction will probably be needed for every frame, so the levels can be set for the first, and the adjustment will be applied to as many frames as the user specifies. However, if the light fades during a sequence when it was intended to remain constant, it would be necessary to increase the brightness gradually over time to compensate. It is possible to apply a suitable correction to each frame individually, and this may occasionally be necessary, but often it is adequate to specify parameters at a few **key frames** and allow their values at intermediate frames to be interpolated.

Figure 6.17 shows the simplest form of temporal variation being applied to achieve a special effect. Final Cut Pro's colour offset effect is similar to the hue component of Photoshop's hue and saturation adjustment, applied to each colour channel individually. The interface shown at the bottom of the figure is typical of the way in which such effects are applied, though they vary from one application (and version) to another. At the left you can see the controls for the effect – a set of sliders for adjusting the hue in the R, G and B channels. To the right of these are three small timelines, one for each channel. Key frames are added to the timeline, and the desired values of the effect's parameter at the corresponding time are set using the slider. You can add as many key frames as

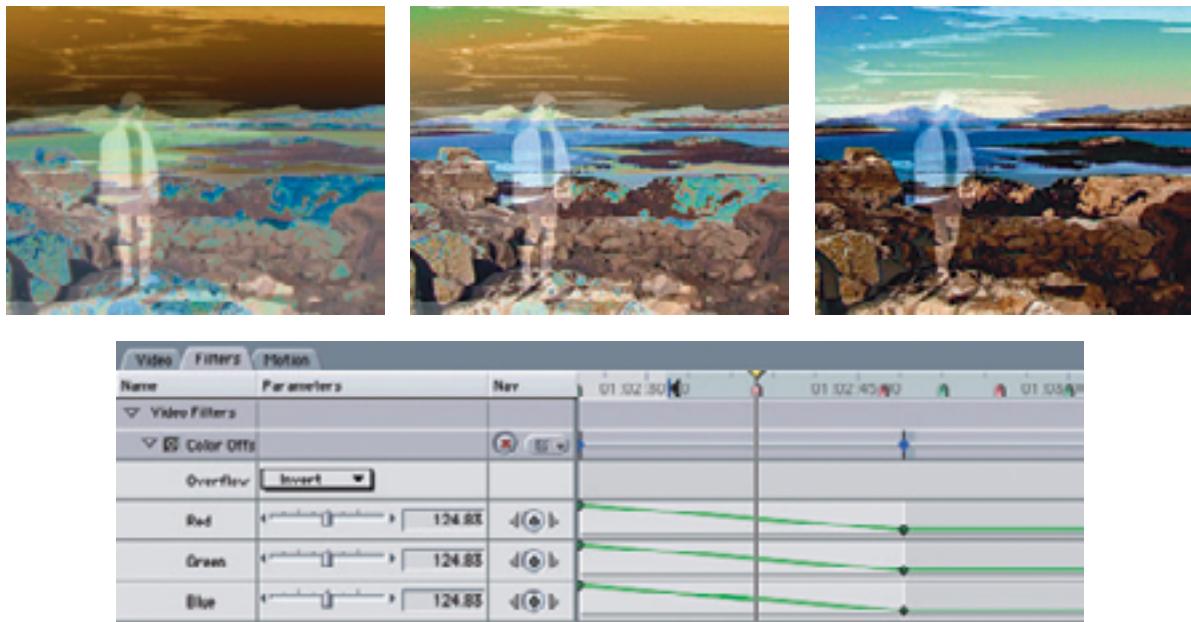


Figure 6.17. A colour offset filter applied to video over time

necessary; the parameters will be interpolated between them. For this simple colour change, a straightforward linear interpolation was used, with all three colours changing in parallel.

Some post-production programs allow you to use Bézier curves to control the interpolation as well as straight lines. Figure 6.18 shows this being done in the creation of a complex special effect that was achieved by varying the parameters of several filters over time (only a few frames of the video sequence are shown here). In this example, abrupt changes in the brightness and colour were used to convey the impression of a sudden intense blast and the disintegration of the scene and figure. The whole sequence is thus created entirely by altering the original bitmapped images of the footage.

Just as some of the image manipulation operations we described in Chapter 4 combined separate layers into a composite result, so some post-production operations combine separate video tracks into a composite. As with still images, some parts of the superimposed tracks must be transparent for superimposition to achieve anything useful. In video, selecting transparent areas is called **keying**. Good video editing and post-production software will offer several different keying methods.

A long-established use of keying in traditional film-making is “blue screening”, which is typically used for inserting isolated elements or figures into scenes artificially – for example, to add models to live footage, or to place live actors in seemingly impossible or dangerous situations. Digital post-production systems support both traditional blue screening – where the actor or model is shot in front of a screen that is a particular shade of blue and then the blue channel is removed – and a

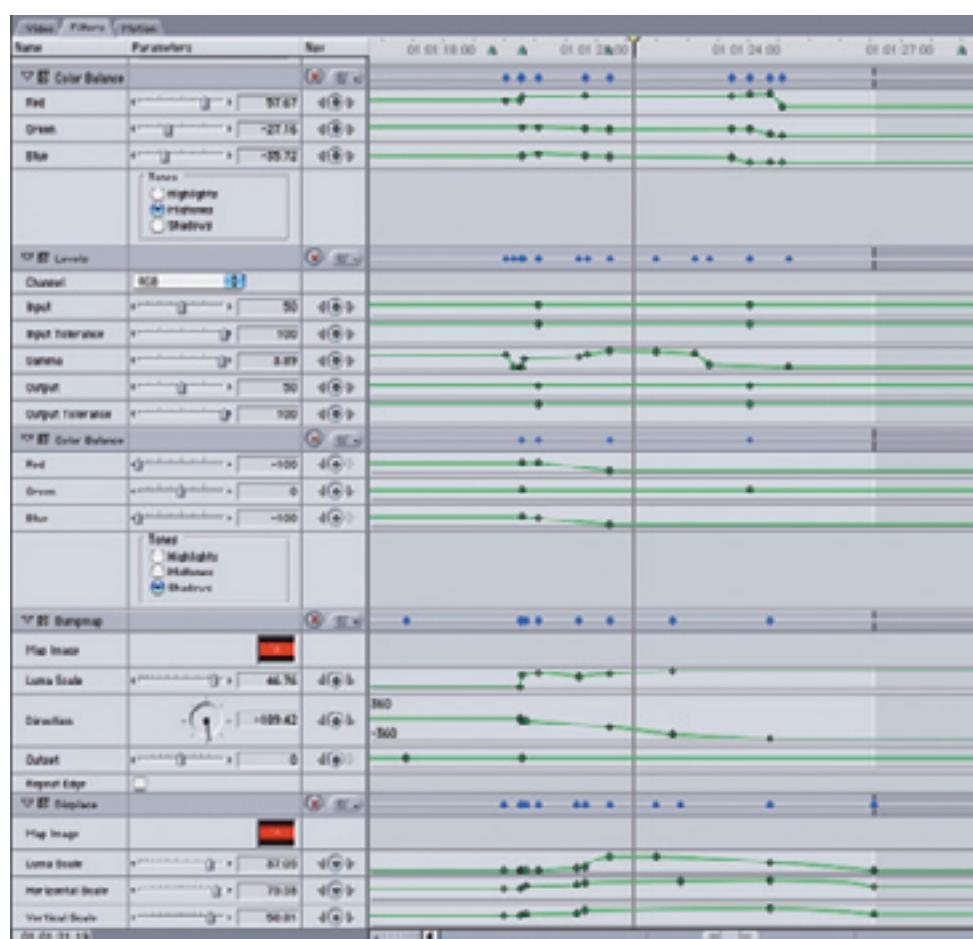


Figure 6.18. A complex set of filters applied to video over time to achieve a special effect

more general form of **chroma keying**, where any colour in a scene can be selected and designated as transparent. Chroma keying is essentially the same as building an alpha channel from a selection made using a magic wand tool. An alternative is **luma keying**, where a brightness threshold is used to determine which areas are transparent.

In some video editing applications it is possible to use selection tools to create a mask. In film and video, a mask used for compositing is called a **matte**. Mattes are frequently used for removing

unwanted elements (such as microphone booms or lights) from a scene, or for allowing live footage to be combined with a still image to convey a special impression. Throughout the history of film-making, mattes have been used to create the impression of a scene that was not really there (not even as a set). Actors can be filmed on a relatively cheap set containing just a few foreground props, with the set itself constructed to a height not much above their heads. The top of the frame is matted out when the action is being filmed, and subsequently replaced with a painting of a large building, landscape or whatever – designed to blend in with the set itself – in order to create the illusion of a much larger, more elaborate or more fantastic environment for the action. Mattes can also be used for split-screen effects. As with many other editing and post-production operations, digital video applications make matting and keying not only much simpler than they were using traditional techniques, but available at low cost on desktop editing systems.

KEY POINTS

Video editing is the process of constructing a complete movie from a set of video clips or scenes, combining them with sound where required.

Post-production is concerned with making changes or compositing the material, using operations that are similar to bitmapped image manipulations.

SMPTE timecode is used to identify frames by their time coordinates.

During editing, clips are imported, trimmed and assembled on a timeline. Transitions, such as dissolves, may be added between overlapping clips.

In post-production, the values of effects' parameters may vary over time.

Chroma keying, luma keying and mattes are used when combining tracks.

Delivery

The traditional way of delivering video is by way of a dedicated service, usually a broadcast television signal, or on a special medium, for instance a DVD, which is played in a dedicated device, such as a DVD player. However, because digital video can be treated as data by programs, it offers other possibilities.

It is usual to call a self-contained piece of video a “movie”, whatever its content may be. A movie can be stored in a file, just as an image can, which raises the usual questions of file formats. It can also be sent over a network, and this raises new issues.

Streaming

A movie stored in a file could be downloaded from a remote server, saved to a local disk and played from there, in much the same way as an image can be downloaded and displayed. Because

of the size of video data this would be a slow process, and it would require large files to be stored locally. A better alternative is to deliver a video data stream from the server, to be displayed as it arrives, without storing it on disk first. Such **streamed video** resembles broadcast television, in that the source video is held on the server, which acts like a TV transmitter sending out the signal, which is played back straight away on a client machine. (In contrast, downloading the entire video would be like having the TV company send a courier round with a DVD whenever you wanted to watch a programme.)

Streamed video opens up the possibility of delivering live video, bringing one of the modes of conventional broadcasting to video on computers. It goes beyond conventional broadcast TV in this area, though, because it is not restricted to a single transmitter broadcasting to many consumers. Any suitably equipped computer can act as both receiver and transmitter, so users on several machines can communicate visually, taking part in what is usually called a video conference.

Until recently, a fundamental obstacle to streamed video has been bandwidth. Decent quality streamed video is restricted to broadband connections (or local area networks); dial-up Internet connections cannot handle the required data rate. Even where the bandwidth is available, the network has to be capable of delivering data with the minimum of delay, and without undue “jitter” – a variation in the delay that can cause independently delivered video and audio streams to lose synchronization. We will return to this subject in Chapter 16.

It may help you to understand the nature of what we will sometimes call “true streaming” by contrasting it with alternative methods of video delivery you may meet on the World Wide Web. The simplest method, which we already mentioned, is **embedded video**, where a movie file is transferred from a server to the user’s machine, and not played back from the user’s disk until the entire file has arrived.

A refinement of this method is called **progressive download** or **HTTP streaming**. With this mode of delivery the file is still transferred to the user’s disk, but it starts playing as soon as enough of it has arrived to make further waiting unnecessary. This will be when the time it will take for the remainder to be downloaded is equal to the duration of the entire movie.

For instance, suppose a 30-second movie has been compressed to a data rate of 2 Mbps. The file would take 120 seconds to download over a slow broadband connection operating at 512 kbps – that is, the user would have to wait two minutes before seeing any of the movie. After 90 seconds have elapsed, though, three-quarters of the frames will have been received by the browser. If the movie starts playing at that point, by the time these frames have been used up new ones will have arrived. In fact, since the movie lasts for 30 seconds, and the remaining quarter of it will take that

amount of time to arrive, it should be possible to play the entire movie starting from the 90-second point. This is illustrated in Figure 6.19.

There is usually an appreciable delay before playback starts, since progressively downloaded movies are typically made with a data rate that exceeds the bandwidth of the network connection. In such a case, this will be the best that can be achieved if dropped frames are to be avoided. The movie file usually remains on the user's hard disk – at least in their Web browser's cache – after playback is completed. Enough disk space to store the whole movie must be available, so progressive download cannot be used often for huge files, such as complete feature films. Also, since an entire file is downloaded, this method of delivery cannot be used for live video, nor does it allow the user to skip over parts of the file without downloading them.

In contrast, true streaming video is never stored on the user's disk. A small buffer may be used to smooth out jitter, but effectively each frame in the stream is played as soon as it arrives over the network. This means that streams can be open-ended, so true streaming can be used for live video, and the length of a pre-recorded movie that is streamed is limited only by the amount of storage available at the server, not by the disk space on the user's machine. Random access to specific points in a stream is possible, except for live streams. True streaming is thus suitable for "video on demand" applications and video-conferencing. It is also more acceptable to copyright holders, because there is normally no copy of the complete movie left on the user's machine for potential copying and redistribution.

If streaming is to work properly, the network must be able to deliver the data stream fast enough for playback. Looked at from the other side, this means that the movie's data rate, and thus its quality, is restricted to what the network can deliver. With modern codecs and broadband connections, full-frame, full-speed playback can be achieved at acceptable quality, although it is advisable to use smaller frames, to allow for connections at the lower end of the broadband range.

The main drawback of true streaming is that it requires a special server, whereas progressive download can be carried out using an ordinary Web server. (Video podcasts can be delivered in either way, since the podcast is just a wrapper containing metadata, which points to the location of the actual video movie.) We will explain the requirements for streaming in more detail

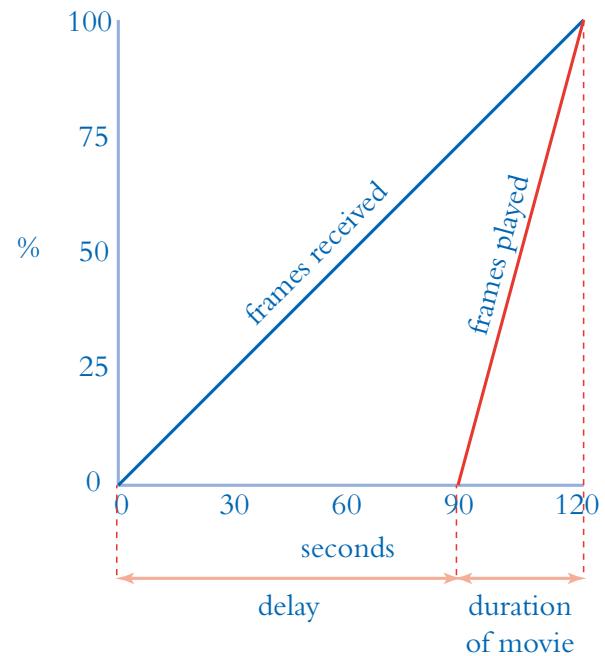


Figure 6.19. *Progressive download*

in Chapter 16. Streaming servers often require the payment of licence fees, and are not usually available on shared hosts, so a restricted budget may necessitate the use of progressive download, even where streaming would be technically feasible.

320

Architectures and Formats

Video formats are more complex than image formats. As we have shown in preceding sections, there are many digital video compression schemes. Each of these schemes requires the information to be encoded in a different way, but does not in itself define a file format. Some standards, such as MPEG-4, define a file format, but the data compressed by the codecs defined in the same standard may be stored in files with other formats, too. Most file formats used for video have been devised to contain data compressed using different codecs. Video is usually accompanied by sound, and as we will see in Chapter 8, audio uses many different codecs and formats too.

To accommodate the multitude of possible combinations of video and audio codecs and formats, the major platforms each provide what is vaguely referred to as a ***multimedia architecture***. The term is not clearly defined, but a multimedia architecture usually features the following:

- An API (Application Programming Interface) that provides facilities for media capture, compression and playback, which can be incorporated into multimedia software.
- One or more codecs.
- A container format for storing media data.
- A streaming server (see Chapter 16).
- Software tools for playback and possibly capture, compression and simple editing. Most multimedia architectures include a Web browser plug-in as well as desktop programs.

Multimedia architectures are component-based, with a mechanism for incorporating additional third-party components. This means, for example, that if some codec is not supported it can be added by way of a component – a new version of the architecture is not required for every new codec.

The container format will usually be able to hold data that has been compressed by many different codecs, not just the native codecs of the architecture. In most cases, the playback functions will be able to cope with many file formats in addition to the architecture's own container. Although this may sound complicated, what it means in practice is that a good multimedia architecture will make it possible to play almost any movie it is presented with, so that users should not have to worry about formats and codecs.

The first multimedia architecture was **QuickTime**, which was introduced by Apple in 1991 and has been extended through new versions ever since. QuickTime's container format is the movie

file, often called a MOV file, as they usually have the extension .mov. QuickTime movies are extremely flexible containers, which can accommodate still images, text and animation as well as many different video and audio formats and compression schemes. Among the video codecs included with current versions of QuickTime are H.264/AVC, MPEG-4 Part 2, Pixlet, and several others developed for specific tasks such as compressing computer-generated animation.

The QuickTime movie file format has been used as the basis of the MPEG-4 file format. To enable application software based on QuickTime to access other types of file, components have been added to make it possible to manipulate files in other formats as if they were native QuickTime. Formats supported in this way include MPEG-1 and MPEG-2, DV, OMF (a high-end professional format), Microsoft's AVI and (using a third-party extension) WMV. As this demonstrates, QuickTime's component architecture makes it easily extensible.

The QuickTime Player is a program that uses the playback components to provide a standard video player that can display movies in any supported format. (Many of its functions have also been incorporated into the popular iTunes program.) A Pro version exists, which adds simple capture, editing and export functions. A fee is charged for activation of the Pro features.

QuickTime is available on both Macintosh and Windows systems. Windows' own multimedia architecture is called **DirectShow**.[†] Although it is organized differently from a programmer's point of view, DirectShow is functionally very similar to QuickTime. It provides a similar set of facilities for creating and manipulating media, including video. Strictly speaking, its container format is **ASF (Advanced Systems Format)** but ASF files containing video data are most often called **Windows Media** or **WMV** files. The Windows Media Video codec which we described earlier can be considered part of DirectShow. Like QuickTime, though, DirectShow allows additional codecs to be added by way of components, so although the number of codecs provided with DirectShow itself is small, many others can be added.

The Windows Media Player is similar to the QuickTime Player. The Windows Media Encoder can be used for capturing and compressing video and converting it between different formats. WMV is the usual format for video data but DirectShow also supports the older AVI format as well as MPEG video and some very early versions of the QuickTime movie format.

The prevalence of the Windows operating system has led to Windows Media and DirectShow being installed on a large proportion of the world's consumer-level computers, but it does not run on any operating system apart from recent versions of Windows.

[†] At least, it is at the time of writing: its name keeps changing.

IN DETAIL

Video for Windows was the predecessor of DirectShow; its associated file format was AVI (Audio-Video Interleaved). AVI is generally considered to be an outdated format. In particular, it provides no means of identifying B-pictures in a stream, so data compressed by any codec that uses bi-directional prediction can only be stored in an AVI file by the use of an additional coding hack.

Nevertheless, there are many AVI files in existence, and the format has been adapted to accommodate modern codecs. In particular, the DivX format is a version of AVI that has been adapted to hold video data compressed with MPEG-4 Part 2. When video file-sharing first became popular, DivX was a popular format for distributing movies. Consequently, many domestic DVD players can play movies in DivX format. However, the DivX project has become fragmented, with the appearance of a competing XviD codec, and the development of a new container file format, DivX Media Format. As support for the standard MP4 format becomes more widespread DivX is becoming less relevant.

Although QuickTime is available for Windows, it is not installed by default. Similarly, although there is a third-party set of QuickTime components that allow Windows Media files to be played back and created on Mac systems, these are not part of the standard QuickTime distribution. Because of licensing issues, Open Source Linux distributions do not support QuickTime or WMV, so again a third-party program must be installed on such systems to play those formats. (And doing so may be illegal in some countries.) Hence, it is certainly not safe to assume that either WMV or QuickTime movies will be playable on all systems. This becomes a problem when video is distributed through the World Wide Web, since there is no way of predicting which computer system visitors to a site will be using.

A surprising solution to the problem of cross-platform video emerged in the form of Adobe's Flash Player. Although it is not built in to any operating system, the Flash Player is among the most widely installed pieces of software in the world. Originally, it was intended purely for playing vector animations in SWF format, as we will describe in Chapter 7. The only way to incorporate video in a SWF movie was by importing each frame as a bitmapped image. This was often done – and still is, for specialized applications – but it provided no means of applying inter-frame compression, so the resulting movies were large. The *Flash Video (FLV)* format was devised to overcome this difficulty.

Flash Video playback in the Flash Player works in a slightly odd way. You can't simply stream an FLV file to the Flash Player. Instead, you must create a SWF (Flash movie) as if you were making an animation. This SWF need only contain a single frame: a script can load and play an FLV movie within that frame. The video can be streamed or progressively downloaded. Typically, the SWF frame holds player controls, which allow the user to start, stop, pause and rewind the video. This

way of working is somewhat inconvenient but it makes it simple to customize the appearance of playback controls and other aspects of the player, because they are just part of the SWF you create to load the video. A selection of standard player controls and a “wizard” for importing video are available in Flash, so the process is not onerous. (See Chapter 6 of *Digital Media Tools* for more practical information about creating and playing Flash Video.)

IN DETAIL

There are third-party media players which can simply play a FLV movie the way the QuickTime Player plays a QuickTime movie, but these are not installed with nearly every Web browser in the way that the Flash Player is.

The near-ubiquity of the Flash Player made FLV into a suitable format for video on the Web. The growth of video-sharing sites – especially YouTube, which used Flash Video – made the format hugely popular in a short period of time. Nevertheless, the quality that could be obtained at suitable bit rates with the original Flash Video codec, or even with the improved On2VP6 codec that we described earlier, was inferior to what was possible with H.264/AVC. Subsequently, though, Adobe added support for MP4 files using H.264/AVC. It is still necessary to make a SWF to load the video, but instead of generating an FLV file – which can only be done with Flash or the Flash Video Encoder – you can load any MPEG-4 file that uses the H.264/AVC codec, and these can be created with any standard video software. (However, both QuickTime and DirectShow support MP4 playback, so the case for going through the extra step of using Flash is less compelling.)

The file formats most often encountered for video are the container formats associated with the QuickTime and DirectShow frameworks (i.e. MOV and WMV files), Flash Video and the MPEG-4 container format. You may also come across **Ogg** files, Ogg being another container format, associated with the same project as the Theora codec we mentioned earlier. The format is a free open standard so there are no restrictions on its use. It is only normally used with video that has been compressed with open codecs, in particular, Ogg Theora. If you want or need to use purely open technology, Ogg and its associated video and audio codecs are probably the only option. All others are either proprietary or subject to licence fees, in theory if not in practice.

The nearest thing to a multimedia architecture in the Linux/Open Source world is **ffmpeg**, which is a command-line tool for video capture, compression and format conversion. It is supported by a collection of libraries providing codecs and some post-production facilities, which can be used like the APIs in the mainstream architectures to provide these functions in many programs. Various media players have been built on top of these libraries. There is also an associated streaming server. Many codecs are supported, including H.264/AVC and Theora. File formats that can be read and written by ffmpeg include WMV, AVI and FLV.



Figure 6.20. *Export settings*

Some platforms demand the use of certain formats and codecs. Video intended for mobile phones must usually be in 3GP format; video for Apple's iPod players must be QuickTime, compressed with H.264/AVC. For playing on computers, there is generally a wider choice, but if you wish to cater for as many people as possible, it is necessary to choose a video format with care.

As we explained earlier, using a format such as WMV or MOV that is tied to a particular multi-media architecture may mean that your video cannot be viewed on all systems. You may decide that this does not matter – if you know for certain that everybody who might watch your video has a recent Windows system, you can safely use WMV and take advantage of the tools for working with that format. For example, this might be the case with a training video that was only intended for distribution over an intranet to employees of a single organization, which had a strict policy concerning the systems that could be attached to the network. More often, though, you will be distributing the video to a heterogeneous collection of systems. This is always the case for video on the Internet. In that case you must use a format that is playable on the largest feasible number of systems.

From our earlier discussion, you should be able to see that MP4 and FLV are the two obvious choices for distribution to heterogeneous systems. MP4 is a standard and playable by DirectShow, QuickTime and the Open Source players available for all platforms. FLV is playable by the Flash Player, which is available for all the major platforms, and is installed on most machines.

Hence, after any resizing and changing of frame rate, video for delivery over networks or for playback on portable devices will usually be exported as MP4, probably using the H.264/AVC

codec, or as FLV, using On2 VP6. Almost any video editing software will be able to export MP4. As you can see from Figure 6.20, the format is chosen during export. Usually, the format is the first thing you must choose, as it determines which codecs can be used. For FLV, you will need to use Flash or the Adobe Media Encoder. Additionally, you will have to create the player movie, as we mentioned previously. (If you are dogmatic about only using Open Source, you will have to prepare an Ogg/Theora movie, but don't expect it to be universally playable.)

For video to be streamed over a network it is common practice to produce a range of different versions matched to the speed of users' network connections. QuickTime and Windows Media allow different versions to be combined into a single movie. The server chooses the appropriate version to stream on the basis of information sent by the player about the connection speed.

KEY POINTS

Video may be delivered over a network as a downloaded file, it may be streamed or it may be delivered by progressive download.

Progressive download means that the movie starts playing when the time taken to download the remaining frames is less than the time it will take to play the whole movie.

When video is streamed, each frame is played as it arrives.

Streaming allows live video and does not require a file to be saved on the user's disk, but it does require sufficient bandwidth to deliver frames fast enough to be played.

A multimedia architecture provides an API for capture, playback and compression; a container format; a streaming server; and software tools, such as a player.

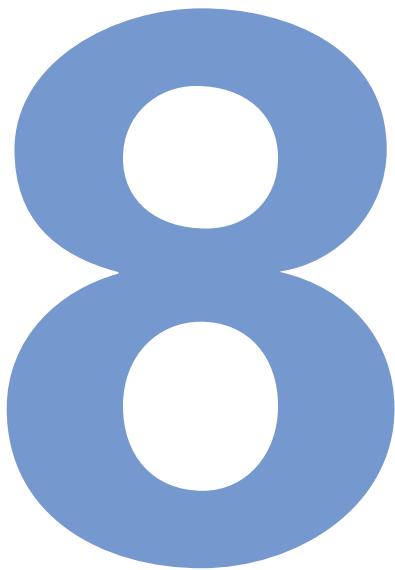
QuickTime and DirectShow are the multimedia architectures included with Mac OS X and Windows, respectively. Their file formats are MOV and WMV.

QuickTime and DirectShow are extensible by way of components, which allow them to use many codecs, including H.264/AVC and WMV 9, and to read and write several additional file formats, such as MP4 and AVI.

Flash Video is widely used for Web video. FLV files must be played in the Flash Player with a SWF that controls the video playback (or in a third-party player).

Ogg is an open format, that can be used in conjunction with the Theora codec to produce movies that are not subject to any restrictions or licence fees.

Web video may need its frame size and frame rate reduced before it is compressed and exported to a suitable format (e.g. MP4 or FLV) which can be played back on most systems.



Sound

■ The Nature of Sound

Waveforms. Perception.

■ Digitizing Sound

Sampling. Quantization. Formats.

■ Processing Sound

Recording and Importing Sound. Sound Editing and Effects.
Combining Sound and Picture.

■ Compression

Speech Compression. Perceptually Based Compression.

■ MIDI

MIDI Messages. General MIDI. MIDI Software.

Sound is different in kind from any of the other digital media types we have considered. All other media are primarily visual, being perceived through our eyes, while sound is perceived through the different sense of hearing. Our ears detect vibrations in the air in a completely different way from that in which our eyes detect light, and our brains respond differently to the resulting nerve impulses. Sound does have something in common with one other topic we have considered, though. Although sound is, for most of us, a familiar everyday phenomenon, like colour it is a complex mixture of physical and psychological factors, which is difficult to model accurately.

Another feature that sound has in common with colour is that you may not always need it. Whereas a multimedia encyclopaedia of musical instruments will be vastly enriched by the addition of recordings of each instrument, few, if any, Web pages need to play a fanfare every time they are visited. Sounds can be peculiarly irritating. Even one's favourite pieces of music can become a jarring and unwelcome intrusion on the ears when inflicted repeatedly by a neighbour's sound system. Almost everyone has at some time been infuriated by the electronic noises of a portable games console, the cuter varieties of ring tone of a mobile phone, or the rhythmic hiss that leaks out of the headphones of a personal stereo. The thoughtless use of such devices has become a fact of modern life; a similar thoughtlessness in the use of sound in multimedia should be avoided. At the very least, it should always be possible for users to turn the sound off.

There are two types of sound that are special: music and speech. These are also the most commonly used types of sound in multimedia. The cultural status of music and the linguistic content of speech mean that these two varieties of sound function in a different way from other sounds and noises, and play special roles in multimedia. Representations specific to music and speech have been developed, to take advantage of their unique characteristics. In particular, compression algorithms tailored to speech are often employed, while music is sometimes represented not as sound, but as instructions for playing virtual instruments.

The Nature of Sound

If a tuning fork is struck sharply on a hard surface, the tines will vibrate at a precise frequency. As they move backwards and forwards, the air is compressed and rarefied in time with the vibrations. Interactions between adjacent air molecules cause this periodic pressure fluctuation to be propagated as a wave. When the sound wave reaches the ear, it causes the eardrum to vibrate at the same frequency. The vibration is then transmitted through the mechanism of the inner ear, and converted into nerve impulses, which we interpret as the sound of the pure tone produced by the tuning fork.

All sounds are produced by the conversion of energy into vibrations in the air or some other elastic medium. The process may involve several steps, in which the energy may be converted into different forms. For example, if one of the strings of an acoustic guitar is picked with a plectrum, the kinetic energy of the musician's hand is converted to a vibration in the string, which is then transmitted via the bridge of the instrument to the resonant cavity of its body, where it is amplified and enriched by the distinctive resonances of the guitar, and then transmitted through the sound hole. If an electric guitar string is picked instead, the vibration of the string as it passes through the magnetic fields of the pickups induces fluctuations in the current which is sent through the guitar lead to an amplifier, where it is amplified and used to drive a loudspeaker. Variations in the signal sent to the speaker coil cause magnetic variations, which are used to drive the speaker cone, which then behaves as a sound source, compressing and rarefying the adjacent air.

While the tines of a good tuning fork vibrate cleanly at a single frequency, most other sound sources vibrate in more complicated ways, giving rise to the rich variety of sounds and noises we are familiar with. As we mentioned in Chapter 2, a single note – such as that produced by a guitar string – is composed of several components, at frequencies that are multiples of the fundamental pitch of the note. Some percussive sounds and most natural sounds do not even have a single identifiable fundamental frequency, but can still be decomposed into a collection – often a very complex one – of frequency components. As in the general case of representing a signal in the frequency domain, which we described in Chapter 2, we refer to a sound's description in terms of the relative amplitudes of its frequency components as its *frequency spectrum*.

The human ear is capable of detecting frequencies between approximately 20 Hz and 20 kHz, although individuals' frequency responses vary greatly. In particular, the upper limit decreases fairly rapidly with increasing age: few adults can hear sounds as high as 20 kHz, although children can. Frequencies at the top end of the range generally only occur as components of the transient attack of sounds. (The general rule that high frequencies are associated with abrupt transitions applies here.) The highest note on an ordinary piano – which more or less defines the limit of most Western music – has a fundamental frequency of only 4186 Hz when in concert pitch. However, it is the transient behaviour of notes that contributes most to the distinctive timbre of instruments. If the attack portion is removed from recordings of an oboe, violin and soprano, playing or singing the same note, for example, the steady portions are indistinguishable.

Waveforms

Interesting sounds change over time. As we just observed, a single musical note has a distinctive attack, and subsequently it will decay, changing its frequency spectrum first as it grows, and then as it dies away. Sounds that extend over longer periods of time, such as speech or music, exhibit a constantly changing frequency spectrum. We can display the *waveform* of any sound by graphically plotting its amplitude against time.

IN DETAIL

The idea of a sound's frequency spectrum changing might be slightly confusing, if you accept that any complex waveform is built out of a collection of frequency components. Strictly, Fourier analysis (as introduced in Chapter 2) can only be applied to periodic signals (i.e. ones that repeat indefinitely). When analysing signals with a finite duration, various expedients must be adopted to fit into the analytic framework.

One approach is to treat the entirety of a signal as one cycle of a periodic waveform; this is roughly what is done when images are broken down into their frequency components.

An alternative is to use a brief section of the signal as if it were a cycle, thus obtaining a snapshot of the frequency make-up at one point. For audio signals, this provides more useful information. A spectrum analysis is typically obtained by sliding a window through the waveform to obtain a sequence of spectra, showing how the signal's frequency components change over time.



Figure 8.1. “Feisty teenager”

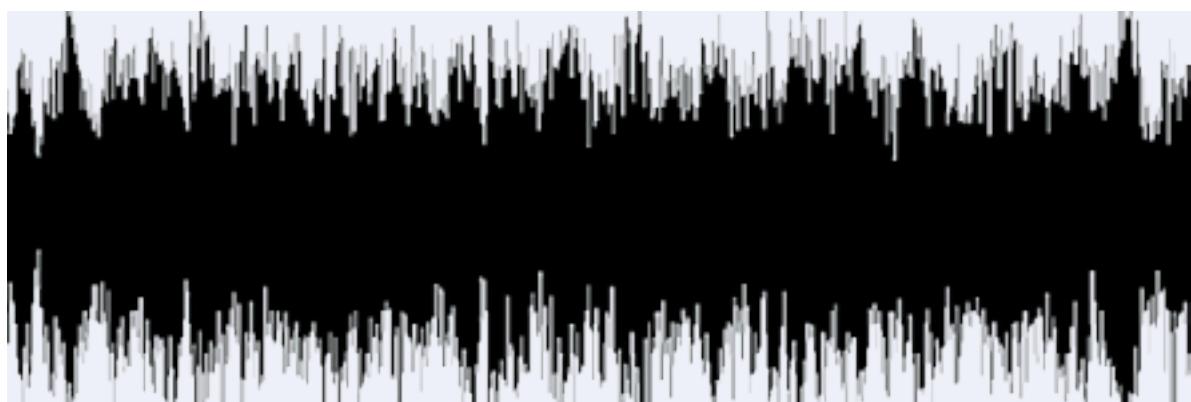


Figure 8.2. *Didgeridoo*

Examination of waveforms can help us characterize certain types of sound. Figures 8.1 to 8.7 show waveforms for a range of types of sound. Figure 8.1 is a short example of speech: the main speaker repeats the phrase “Feisty teenager” twice, then a more distant voice responds. You can clearly identify the separate syllables, and recognize that the same phrase is repeated, the second time faster and with more emphasis. In between the phrases there is almost silence – the sound was recorded in the open air and there is background noise, which is visible as the thin band running along the axis. You can see that it could be possible to extract individual syllables and recombine them to synthesize new words, and that, if it were necessary to compress speech, a lot could be achieved by removing the silences between phrases. The clearly demarcated syllables also provide a good basis for synchronizing sound with video, as we will see later.

The next four figures show the waveforms of some different types of music. The first three are purely instrumental, and do not exhibit the same character as speech. The first, Figure 8.2, is taken from an Australian aboriginal didgeridoo piece. This is characterized by a continuous drone, which requires the musician to employ a “circular breathing” technique to maintain it. The waveform shows this drone as the thick continuous black region, with its rhythmic modulation.



Figure 8.3. *Boogie-woogie*



Figure 8.4. *Violin, cello and piano*

Figure 8.3 shows the waveform of a piece of boogie-woogie, played by a pianist accompanied by a small group. The rhythm is clearly visible, but it is not possible to distinguish the melody played by the right hand (unless, perhaps, you are a very experienced audio technician). Figure 8.4 is a completely different waveform, corresponding to a very different piece of music: a contemporary “classical” work arranged for violin, cello and piano. It shows a great dynamic range (difference



Figure 8.5. “*Men grow cold...*”

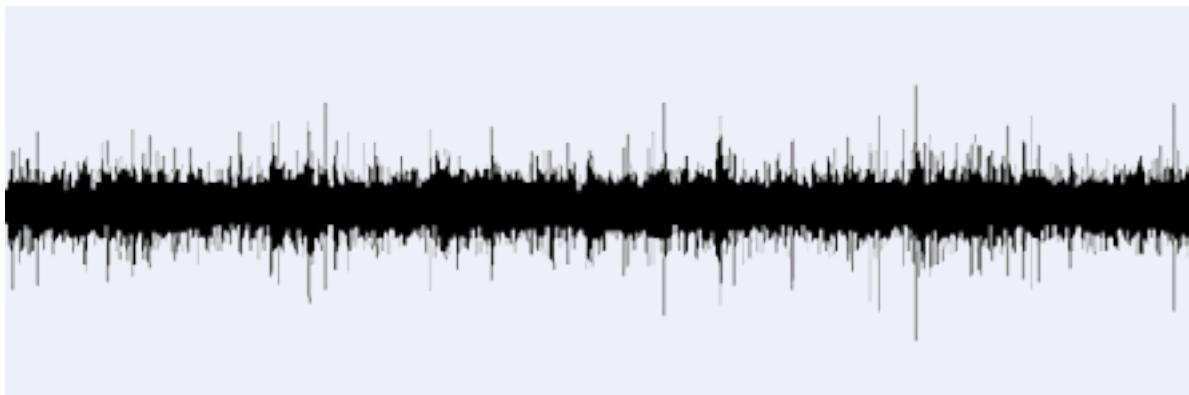


Figure 8.6. *A trickling stream*



Figure 8.7. *The sea*

between the loudest and quietest sounds). Although the steep attack of the louder phrases tells you something about the likely sound of this music, there is no obvious rhythm, and it is not possible to separate out the different instruments (although they can be very clearly identified when listening to the music).

As you would expect, singing combines characteristics of speech and music. Figure 8.5 is typical: the syllables of each word are easily identifiable, as is the rhythm, but the gaps between sung phrases are filled with the musical accompaniment. It is possible to see the singer's phrasing, but quite impossible to deduce the lyrics,[†] and, although voice prints are unique to each individual, it is unlikely that anyone could identify the singer from this waveform, despite her distinctive voice. (It's Marilyn Monroe.)

Figures 8.6 and 8.7 are both natural water sounds. The first is a recording of the trickling sound of water in a small stream; it is almost continuous and has a small dynamic range. The random spikes do not correspond to any audible clicks or other abrupt sound; they are just slight variations in the water's flow, and some background noise. The second waveform was recorded on the seashore. There is a constant background of surf and two distinct events. The first is a wave breaking fairly close to the microphone, while the second is the water splashing into a nearby rock pool and then receding through a gap in the rocks. This waveform can almost be read as a story.

As these illustrations show, the shape of a waveform can show a certain amount of the gross character and dynamics of a sound, but it does not convey the details, and it is not always easy to correlate against the sound as it is heard. The main advantage of these visual displays is their static nature. A piece of sound can be seen in its entirety at one time, with relationships such as the intervals between syllables or musical beats visible. This makes it relatively simple to analyse the sound's temporal structure – which is especially useful for synchronization purposes – compared with trying to perform the same analysis on the dynamically changing sound itself, which is only heard an instant at a time.

Perception

Waveforms and the physics of sound are only part of the story. Sound only truly exists as a sensation in the mind, and the perception of sound is not a simple registering of the physical characteristics of the waves reaching the ears. Proofs of this abound, both in the literature and in everyday experience. For example, if a pure 200 Hz tone is played, first softly, then louder, most listeners will believe that the louder tone has a lower pitch than the quieter one, although the same illusion is not perceived with higher-frequency tones. Similarly, complex tones sometimes seem to have a lower pitch than pure tones of the same frequency. Most people with good hearing can distinguish the sound of their own name spoken on the opposite side of a noisy room, even if the

[†] Men grow cold, as girls grow old, and we all lose our charms in the end.

rest of what is said is inaudible, or carry on a successful conversation with someone speaking at a volume lower than that of the ambient noise.

334

One of the most useful illusions in sound perception is stereophony. The brain identifies the source of a sound on the basis of the differences in intensity and phase between the signals received from the left and right ears. If identical signals are sent to both ears, the brain interprets the sound as coming from a non-existent source that lies straight ahead. By extension, if a sound is recorded using a pair of microphones to produce two monophonic channels, which are then fed to two speakers that are a suitable distance apart, the apparent location of the sound will depend on the relative intensity of the two channels. If they are equal it will appear in the middle, if the left channel is louder (because the original sound source was nearer to the left-hand microphone) it will appear to the left, and so on. In this way, the familiar illusion of a sound stage between the speakers is constructed.

Because of the psychological dimension of sound, it is unwise, when considering its digitization and reproduction, to place too much reliance on mathematics and measurable quantities. Pohlmann's comments[†] about the nature of sound and its reproduction should be borne in mind:

“Given the evident complexity of acoustical signals, it would be naïve to believe that analog or digital technologies are sufficiently advanced to capture fully and convey the complete listening experience. To complicate matters, the precise limits of human perception are not known. One thing is certain: at best, even with the most sophisticated technology, what we hear being reproduced through an audio system is an approximation of the actual sound.”

Digitizing Sound

The digitization of sound is a fairly straightforward example of the processes of quantization and sampling described in Chapter 2. Since these operations are carried out in electronic analogue-to-digital converters, the sound information must be converted to an electrical signal before it can be digitized. This can be done by a microphone or other transducer, such as a guitar pickup, just as it is for analogue recording or broadcasting.

Increasingly, digital audio, especially music, is stored in files that can be manipulated like other data. In particular, digital audio files can be stored on servers and downloaded or distributed as “podcasts” (see Chapter 16). Digital audio players, such as Apple's iPod, store such files on their internal hard disks or flash memory. Almost always, audio in this form is compressed.

[†] Ken C. Pohlmann, *Principles of Digital Audio*, p. 5.

Contemporary formats for digital audio are influenced by the CD format, which dominated audio for over two decades. For instance, the sampling rate and number of quantization levels used for high-quality audio is almost always the same as that used for CD. (Despite the journalistic habit of distinguishing between CD and “digital” music, CD audio is, of course, digital.)

335

Sampling

If high-fidelity audio reproduction is desired, a sampling rate must be chosen that will preserve at least the full range of audible frequencies. If the limit of hearing is taken to be 20 kHz, a minimum rate of 40 kHz is required by the Sampling Theorem. The sampling rate used for audio CDs is 44.1 kHz – the precise figure being chosen by manufacturers to produce a desired playing time given the size of the medium. Because of the ubiquity of the audio CD, the same rate is commonly used by the sound cards fitted to computers, to provide compatibility. Where a lower sound quality is acceptable, or is demanded by limited bandwidth, sub-multiples of 44.1 kHz are used: 22.05 kHz is commonly used for audio destined for delivery over the Internet, while 11.025 kHz is sometimes used for speech.

Some professional and semi-professional recording devices use sample rates that are multiples of 48 kHz. This was the rate used by DAT (digital audio tape) recorders, which were popular for live recording and low-budget studio work in the late 1990s and early twenty-first century. The solid-state memory card and disk-based recorders that have taken over this function often record at double or even four times this sampling rate (96 kHz or 192 kHz).

CD players and solid state recorders have the advantage that they can generate digital output, which can be read in by a suitably equipped computer without the need for extra digitizing hardware. In this respect, they resemble DV cameras. Digital audio inputs on modern computers usually support sampling rates of 44.1 kHz, 48 kHz and 96 kHz, so it should be possible to read digital audio in most formats to disk without the need to resample it.

IN DETAIL

The necessity to resample data sampled at 48 or 96 kHz often occurs if the sound is to be combined with video. Some video applications do not yet support the higher sampling rates used by popular recording devices. For multimedia work it may therefore be preferable to sample sound at 44.1 kHz, if this rate is available, since it is supported by all the major desktop video editing programs.

Sampling relies on highly accurate clock pulses to determine the intervals between samples. If the clock drifts, so will the intervals. Such timing variations are called *jitter*. The effect of jitter is to introduce noise into the reconstructed signal. At the high sampling frequencies required by sound,

there is little tolerance for jitter: it has been estimated that for CD-quality sound, the jitter in the ADC must be less than 200 picoseconds (200×10^{-12} seconds).

336

Even though they may be inaudible, frequencies in excess of 20 kHz are present in the spectra of many sounds. If a sampling rate of around 40 kHz is used, these inaudible components will manifest themselves as aliasing when the signal is reconstructed. In order to avoid this, a filter is used to remove any frequencies higher than half the sampling rate before the signal is sampled.

Quantization

We mentioned in Chapter 2 that the number of quantization levels for analogue-to-digital conversion in any medium is usually chosen to fit into a convenient number of bits. For sound, the most common choice of sample size is 16 bits – as used for CD audio – giving 65,536 quantization levels. This is generally sufficient to eliminate quantization noise if the signal is dithered, as we will describe shortly. As with images, smaller samples sizes (lower bit-depths, as we would say in the context of images) are sometimes needed to maintain small file sizes and bit rates. The minimum acceptable is 8-bit sound, and even this has audible quantization noise, so it can only be used for applications such as voice communication, where the distortion can be tolerated. In the search for higher-fidelity reproduction, as many as 24 bits are sometimes used to record audio samples, but this imposes considerable demands on the accuracy of ADC circuitry.



Figure 8.8. Undersampling a pure sine wave

Quantization noise will be worst for signals of small amplitude. In the extreme, when the amplitude is comparable to the difference between quantization levels, an analogue signal will be coarsely approximated by samples that jump between just a few quantized values. This is shown in Figure 8.8. The upper waveform is a pure sine wave; below it is a digitized version, where only four levels are available to accommodate the amplitude range of the original signal.[†] Evidently, the sampled waveform is a poor approximation of the original. The approximation could be improved by increasing the number of bits for each sample, but it is usual to employ a more economical technique, which resembles the anti-aliasing applied when rendering vector graphics. Its operation is somewhat counter-intuitive.

Before sampling, a small amount of random noise is added to the analogue signal. The word “dithering” (which we used with a somewhat different meaning in Chapter 5) is used in the audio field to refer to this injection of noise. The effect which this has on sampling is illustrated

[†] If you want to be scrupulous, since these images were prepared using a digital audio application, the top waveform is a 16-bit sampled sine wave (a very good approximation), the lower is the same waveform downsampled to 2 bits.

in Figure 8.9. The upper waveform is the original sine wave with added dither. (We have used rather more noise than is normal, in order to illustrate the effect more clearly.) The lower waveform is a sampled version of this dithered signal.

What has happened is that the presence of the noise has caused the samples to alternate rapidly between quantization levels, instead of jumping cleanly and abruptly from one to the next as in Figure 8.8. The sharp transitions have been softened. Putting it another way, the quantization error has been randomized. The price to be paid for the resulting improvement in sound quality is the additional random noise that has been introduced, but this is less intrusive than the quantization noise it has eliminated.

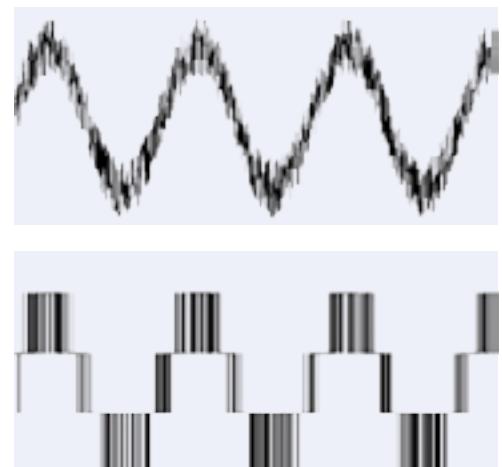


Figure 8.9. Dithering

The effect of sampling and dithering on the signal's frequency spectrum is shown in Figure 8.10; the horizontal x -axis represents frequency, the vertical y -axis amplitude (with the colours being used as an extra visual indication of intensity) and the back-to-front z -axis represents time. The first spectrum is the pure sine wave. As you would expect, it is a spike at the wave's frequency, which is constant over time. To its right is the spectrum of the sampled signal: spurious frequencies and noise have been introduced. These correspond to the frequency components of the sharp edges. Below the pure sine wave is the spectrum of the dithered version. The extra noise is randomly distributed across frequencies and over time. In the bottom right is the sampled version of this signal. The pure frequency has re-emerged clearly, but random noise is present where before there was none. However, although this noise will be audible, the ear will be able to discern the signal through it, because the noise is random. Where the undithered signal was sampled, the noise was concentrated near to the signal frequency, in a way that is much less easily ignored.

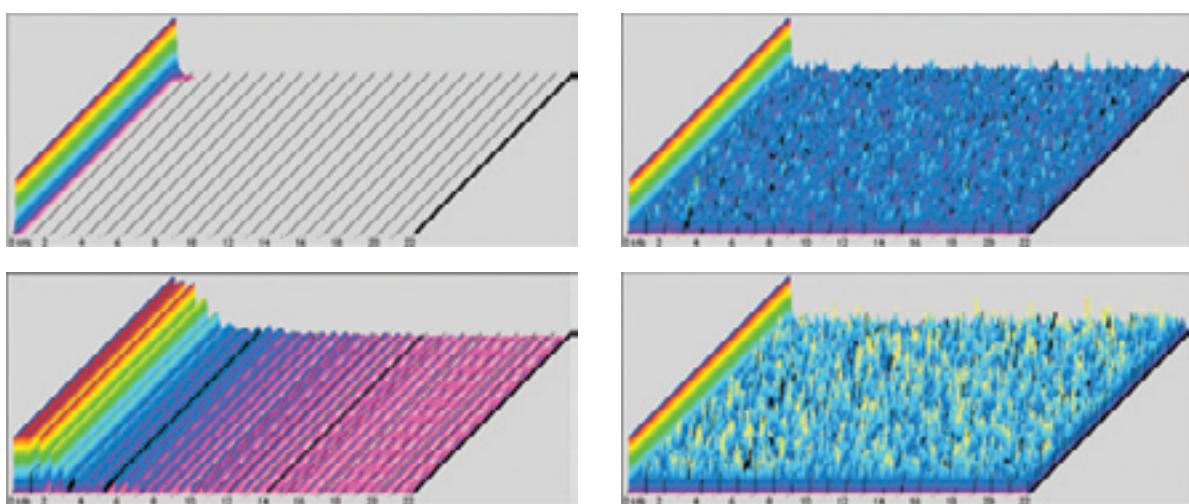


Figure 8.10. Audio frequency spectra showing the effect of undersampling and dithering

Formats

Most of the development of digital audio has taken place in the recording and broadcast industries, where the emphasis is on physical data representations and data streams for transmission and playback. There are standards in these areas that are widely adhered to. The use of digital sound on computers is a much less thoroughly regulated area, where a wide range of incompatible proprietary formats and *ad hoc* standards can be found. Each of the three major platforms has its own sound file format: AIFF for Mac OS, AU for other varieties of Unix, and WAV (or WAVE) for Windows, but support for all three is common on all platforms.

The standardizing influence of the Internet has been less pronounced in audio than it is in graphics. MP3 files have been widely used for downloading and storing music on computers and mobile music players. “Podcasts” typically use MP3 as the format for the audio that they deliver. The popularity of music-swapping services using MP3 led to its emergence as the leading audio format on the Internet, but QuickTime and Windows Media are used as container formats for audio destined for Apple’s iPod music players and various devices that incorporate Windows Media technology. On Web pages, Flash movies are sometimes used for sound, because of the wide deployment of the Flash Player. It is possible to embed sound in PDF documents, but the actual playing of the sound is handled by other software, such as QuickTime, so MP3 is a good choice of format here, too, because it can be played on all the relevant platforms.

MP3 has its own file format, in which the compressed audio stream is split into chunks called “frames”, each of which has a header, giving details of the bit rate, sampling frequency and other parameters. The file may also include metadata tags, oriented towards musical content, giving the title of a track, the artist performing it, the album from which it is taken, and so on. As we will describe later in this chapter, MP3 is primarily an encoding, not a file format, and MP3 data may be stored in other types of file. In particular, QuickTime may include audio tracks encoded with MP3, and Flash movies use MP3 to compress any sound they may include.

In Chapter 6, we explained that streamed video resembles broadcast television. Streamed audio resembles broadcast radio – that is, sound is delivered over a network and played as it arrives, without having to be stored on the user’s machine first. As with video, this allows live transmission and the playing of files that are too big to be held on an average-sized hard disk. Because of the lower bandwidth required by audio, streaming is more successful for sound than it is for video. Streaming QuickTime can also be used for audio, on its own as well as accompanying video. QuickTime includes an AAC codec for high-quality audio. Windows Media audio can also be streamed. Both of these formats, as well as MP3, are used for broadcasting live concerts and for the Internet equivalent of radio stations.

KEY POINTS

Sounds are produced by the conversion of energy into vibrations in the air or some other elastic medium, which are detected by the ear and converted into nerve impulses which we experience as sound.

A sound's frequency spectrum is a description of the relative amplitudes of its frequency components.

The human ear can detect sound frequencies roughly in the range 20 Hz to 20 kHz, though the ability to hear the higher frequencies is lost as people age.

A sound's waveform shows how its amplitude varies over time.

Perception of sound has a psychological dimension.

CD audio is sampled at 44.1 kHz. Sub-multiples of this value may be used for low-quality digital audio. Some audio recorders use sampling rates that are multiples of 48 kHz.

Audio sampling relies on highly accurate clock pulses to prevent jitter.

Frequencies greater than half the sampling rate are filtered out to avoid aliasing.

CD audio uses 16-bit samples to give 65,536 quantization levels.

Quantization noise can be mitigated by dithering, i.e. adding a small amount of random noise which softens the sharp transitions of quantization noise.

Sound may be stored in AIFF, WAV or AU files, but on the Internet the MP3 format is dominant. MP3 data may be stored in QuickTime and Flash movies.

339

Processing Sound

With the addition of suitable audio input, output and processing hardware and software, a desktop computer can perform the functions of a modern multi-track recording studio. Such professional facilities are expensive and demanding on resources, as you would expect. They are also as complex as a recording studio, with user interfaces that are as intimidating to the novice as the huge mixing consoles of conventional studios. Fortunately, for multimedia, more modest facilities are usually adequate.

There is presently no single sound application that has the *de facto* status of a cross-platform desktop standard, in the way that Photoshop and Dreamweaver, for example, have in their respective fields. Several different packages are in use, some of which require special hardware support. Most of the well-known ones are biased towards music, with integrated support for MIDI sequencing (as described later in this chapter) and multi-track recording.

Several more modest programs, including some Open Source applications, provide simple recording and effects processing facilities. A specialized type of audio application has recently achieved some popularity among people who are not audio professionals. Apple's GarageBand and Adobe Soundbooth exemplify this type of program. They provide only primitive facilities for recording, importing and editing sound, and only a few of the effects that are found in professional software. Their novelty lies in facilities for creating songs. In the case of GarageBand, this is done by combining loops, which may either be recorded live instruments, or synthesized. In Soundbooth, templates consisting of several musical segments may be customized – for example by changing the orchestration or dynamics, or by rearranging the segments – to produce unique “compositions”, which might serve as adequate soundtracks for corporate presentations, home videos and similar undemanding productions.

Video editing packages usually include some integrated sound editing and processing facilities, and some offer basic sound recording. These facilities may be adequate for multimedia production in the absence of special sound software, and are especially convenient when the audio is intended as a soundtrack to accompany picture.

Given the absence of an industry standard sound application for desktop use, we will describe the facilities offered by sound programs in general terms only, without using any specific example.

Recording and Importing Sound

Many desktop computers are fitted with built-in microphones, and it is tempting to think that these are adequate for recording sounds. It is almost impossible to obtain satisfactory results with these, however – not only because the microphones themselves are usually small and cheap, but because they are inevitably close to the machine's fan and disk drives, which means that they will pick up noises from these components. It is much better to plug an external microphone into a sound card, but if possible, you should do the actual recording using a dedicated device, such as a solid-state audio recorder, and a professional microphone, and capture it in a separate operation. Compression should be avoided at this stage. Where sound quality is important, or for recording music to a high standard, it will be necessary to use a properly equipped studio. Although a computer can form the basis of a studio, it must be augmented with microphones and other equipment in a suitable acoustic environment, so it is not really practical for a multimedia producer to set up a studio for one-off recordings. It may be necessary to hire a professional studio, which offers the advantage that professional personnel will generally be available.

Before recording, it is necessary to select a sampling rate and sample size. Where the sound originates in analogue form, the choice will be determined by considerations of file size and bandwidth, which will depend on the final use to which the sound is to be put, and the facilities available for sound processing. As a general rule, the highest possible sampling rate and sample size

should be used, to minimize deterioration of the signal when it is processed. If a compromise must be made, the effect on quality of reducing the sample size is more drastic than that of reducing the sampling rate. The same reduction in size can be produced by halving the sampling rate or halving the sample size, but the former is the better option. If the signal is originally a digital one – the digital output from a solid-state recorder, for example – the sample size should be matched to the incoming rate, if possible.

A simple calculation suffices to show the size of digitized audio. The sampling rate is the number of samples generated each second, so if the rate is r Hz and the sample size is s bits, each second of digitized sound will occupy $rs/8$ bytes. Hence, for CD quality, $r = 44.1 \times 10^3$ and $s = 16$, so each second occupies just over 86 kbytes (86×1024 bytes), each minute roughly 5 Mbytes. These calculations are based on a single channel, but audio is almost always recorded in stereo, so the estimates should be doubled. Conversely, where stereo effects are not required, the space occupied can be halved by recording in mono.

The most vexatious aspect of recording is getting the levels right. If the level of the incoming signal is too low, the resulting recording will be quiet, and more susceptible to noise. If the level is too high, **clipping** will occur – that is, at some points, the amplitude of the incoming signal will exceed the maximum value that can be recorded. The value of the corresponding sample will be set to the maximum, so the recorded waveform will apparently be clipped off straight at this threshold. (Figure 8.11 shows the effect on a pure sine wave.) The result is heard as a particularly unpleasant sort of distortion.

Ideally, a signal should be recorded at the highest possible level that avoids clipping. Sound applications usually provide level meters, so that the level can be monitored, with clipping alerts. Where the sound card supports it, a gain control can be used to alter the level. If this is not available, the only option is to adjust the output level of the equipment from which the signal originates.

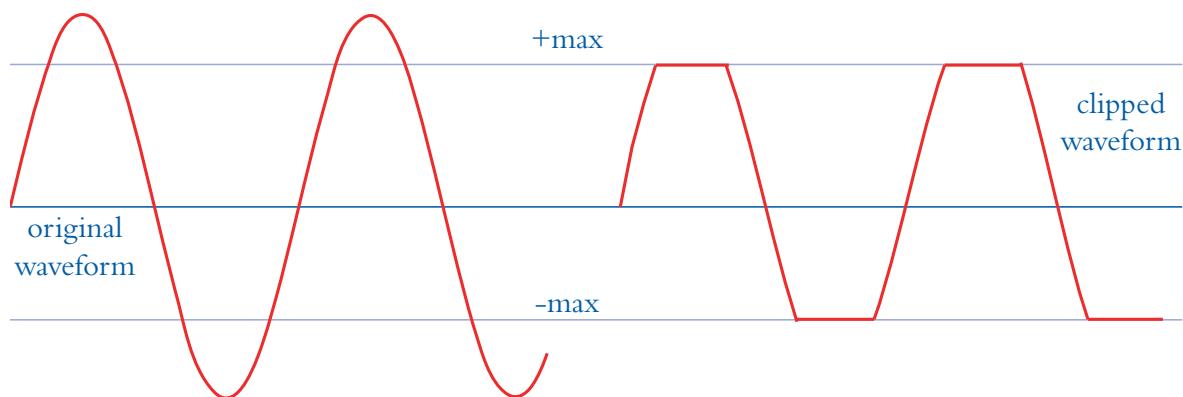


Figure 8.11. Clipping

Setting the level correctly is easier said than done, especially where live recordings are being made. To preserve the dynamic range of the recording the same gain must be used throughout, but the optimum can only be determined at the loudest point. When the sound is live, this cannot be known in advance, and only experience can be used to choose gain settings. Where the material already exists (on CD, for example) it is possible – and usually necessary – to make several passes in order to find the best values.

Some software includes automatic gain controls, which vary the gain dynamically according to the amplitude of the signal, in order to prevent clipping. To do this they must reduce the volume of louder passages, so as a side-effect they reduce the dynamic range of the recording. This is generally undesirable, but it may be necessary if suitable levels cannot be maintained throughout the recording.

IN DETAIL

It may be obvious, but it seems worth emphasizing that once a signal has been clipped, nothing can be done to restore it. Reducing the amplitude subsequently just produces a smaller clipped signal. There is no way to recover the lost waveform.

Similarly, although sound programs often provide a facility for “normalizing” a sound after recording, by amplifying it as much as possible without causing clipping, this stretches the dynamic range of the original without adding any more detail. In practice it may be necessary to use this facility, or to select and amplify particularly quiet passages within a sound editing application after the recording has been made. In principle, though, the gain should always be set correctly, both when recording to a dedicated device, and when recording or capturing to disk.

A technically simpler alternative to recording sound is to import it from an audio CD. Although audio CDs use a different format from CD-ROM, they are nevertheless a structured collection of digital data, so they can be read by suitable software. QuickTime, for example, includes an audio CD import component that allows any sound application based on QuickTime to open tracks on a CD just like any other file. This is the simplest way of importing sounds, but most recorded music is copyrighted, so it is necessary to obtain permissions first. Copyright-free collections of original music and sound effects can be obtained, much like royalty-free image libraries, although the music tends to be anodyne. Composers and musicians with access to professional recording facilities may supply their work on CD, avoiding the need for the multimedia producer to deal with the sound recording process. However, even when importing sounds from CDs there can be difficulty in getting the levels right.

The Internet is a rich source of ready-made sounds and music. The early music download and file-sharing services of dubious legality have been largely superseded by legitimate downloads through commercial online music stores. While it is legal to download these files, or to listen to them and transfer them to a music player, it remains generally illegal to use them in any published form without obtaining clearance from the copyright holders. Some downloaded music is therefore subject to **Digital Rights Management (DRM)**, which aims to prevent it from being used for any purpose not approved by the copyright owners. However, the efficacy of DRM is limited and it causes considerable resentment among consumers. At the time of writing, its use in connection with music is declining.

In any case, music that is distributed over the Internet is usually compressed using MP3 or AAC (see below). Like JPEG image compression, these compression algorithms discard information, so compressed sound is not an ideal source for subsequent processing.

Sound Editing and Effects

We can identify several classes of operation that we might want to apply to recorded sounds. Most of them have counterparts in video editing, and are performed for similar reasons.

First there is editing, in the sense of trimming, combining and rearranging clips. The essentially time-based nature of sound naturally lends itself to an editing interface based on a timeline. A typical sound editing window is divided into tracks – in imitation of the separate tape tracks used on traditional recording equipment – providing a clear graphic representation of the sound through time. The sound in each track may usually be displayed as a waveform; the time and amplitude axes can be scaled, allowing the sound to be examined in varying degrees of detail. Editing is done by cutting and pasting – or dragging and dropping – selected parts of the track. Each stereo recording will occupy two tracks, one for each channel. During the editing process many tracks may be used to combine sounds from separate recordings. Subsequently, these will be “mixed down” onto one or two tracks, for the final mono or stereo output. When mixing, the relative levels of each of the tracks can be adjusted to produce the desired balance – between different instruments, for example.

A special type of edit has become common in audio: the creation of loops. Very short loops are needed to create voices for the electronic musical instruments known as samplers (whose functions are increasingly performed by software). Here, the idea is to create a section of sound that represents the sustained tone of an instrument, such as a guitar, so that arbitrarily long notes can be produced by interpolating copies of the section between a sample of the instrument’s attack and one of its decay. It is vital that the sustained sample loops cleanly. There must not be abrupt discontinuities between its end and beginning, otherwise audible clicks will occur where the copies fit together. Although some software makes such loops automatically, using built-in heuristics such

as choosing zero crossings for each end of the loop, the best results require a detailed examination of the waveform by a person. Longer loops are used in certain styles of dance music based on the combination of repeating sections. Again, there is a requirement for clean looping, but this time at the coarser level of rhythmic continuity. Software, such as GarageBand, can be used to put together even longer loops from a pre-recorded library, pitch- and time-shifting them so they are in the same key and tempo. This allows non-composers to produce music of a sort, and musicians to create backing tracks and orchestrations.

As well as editing, audio has its equivalent of post-production – altering sounds to correct defects, enhance quality, or otherwise modify their character. Just as image correction is described in terms of filters, which are a digital equivalent of traditional optical devices, so sound alteration is described in terms of gates and filters, by analogy with the established technology. Whereas analogue gates and filters are based on circuitry whose response produces a desired effect, digital processing is performed by algorithmic manipulation of the samples making up the signal. The range of effects – and the degree of control over them – that can be achieved in this way is much greater than is possible with analogue circuits. Several standard plug-in formats are in use that allow effects to be shared among programs. Although it is not an audio application, Premiere's effects plug-in format is becoming widely used. At a more professional level, the formats associated with Cubase VST and with DigiDesign ProTools are popular.

The most frequently required correction is the removal of unwanted noise. For example, in Figure 8.1, it might be considered desirable to remove the background noises that were unavoidably picked up by the microphone, since the recording was made in the open air. A **noise gate** is a blunt instrument that is used for this purpose. It eliminates all samples whose value falls below a specified threshold, with samples above the threshold left alone. As well as specifying the threshold, it is usual to specify a minimum time that must elapse before a sequence of low-amplitude samples counts as a silence, and a similar limit before a sequence whose values exceed the threshold counts as sound. This prevents the gate being turned on or off by transient glitches. By setting the threshold just above the maximum value of the background noise, the gaps between words in our example will become entirely silent. Since the noise gate has no effect on the speaker's words, the accompanying background noise will cut in and out as he speaks, which may well turn out to be more distracting than the original noise. (You may have heard this phenomenon on the soundtracks of old films that have been restored and reissued on DVD.) This illustrates a general problem with noise removal: the noise is intimately combined with the signal, and although people can discriminate between the two, computer programs generally cannot.

Noise gates can be effective at removing hiss from music, since, in this case, the noise is hidden except in silent passages, where it will be removed by the noise gate. There are more sophisticated ways of reducing noise than the all-or-nothing filtering of the noise gate, though. Filters that

remove certain bands of frequencies can be applied to noise that falls within a specific frequency range. **Low pass** filters, which allow low frequencies to pass through them, removing high frequencies, can be used to take out hiss. **High pass** filters, which pass the high frequencies and block the low ones, are used to remove “rumble”, that is, low-frequency noise caused by mechanical vibrations. Figures 8.12 and 8.13 show the effect of low and high pass filters on the spectrum and waveform of the sea sound from Figure 8.7. (The upper spectrum and waveform in each figure are the original sound; the lower ones are the sound after filtering.)

A **notch filter** removes a single narrow frequency band. The commonest use of notch filters is to remove hum picked up from the mains, which will have a frequency of exactly 50 Hz or 60 Hz, depending on the geographical location in which the sound was recorded. Some sophisticated programs offer the user the ultimate facility of being able to redraw the waveform, rubbing out the individual spikes that correspond to clicks, and so on. To do this effectively, however, requires considerable experience and the ability to interpret the visual display of a waveform in acoustic terms, which, as the examples shown earlier demonstrate, is not always easy.

IN DETAIL

Although the noise reduction facilities available in desktop sound applications are fairly crude and ineffectual, more elaborate – and more computationally expensive – approaches have been developed. One approach is based on attempting to analyse the acoustic properties of the original recording apparatus on the basis of the make-up of the noise in quiet passages, and then compensating for it in the music. Sophisticated noise reduction techniques are used to restore old records from the early part of the twentieth century, and also to reconstruct other damaged recordings, such as the tapes from voice recorders of crashed aircraft.

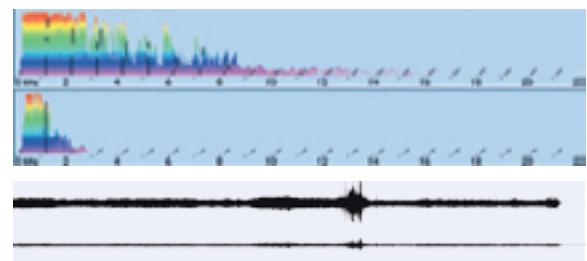


Figure 8.12. Low pass filtering

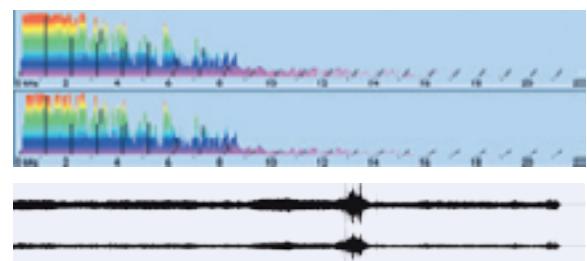


Figure 8.13. High pass filtering

Specialized filters are available for dealing with certain common recording defects. A **de-esser** is a filter that is intended to remove the sibilance that results from speaking or singing into a microphone placed too close to the performer. **Click repairers** are intended to remove clicks from recordings taken from damaged or dirty vinyl records. (There are also effects plug-ins that attempt to add authentic-sounding vinyl noise to digital recordings.) Although these filters are

more discriminating than a noise gate, they are not infallible. The only sure way to get perfect sound is to start with a perfect recording – microphones should be positioned to avoid sibilance, and kept well away from fans and disk drives, cables should be screened to avoid picking up hum, and so on.

346

When we consider effects that alter the quality of a sound, there is a continuum from those that perform minor embellishments to compensate for poor performance and recording, to those that radically alter the sound, or create new sounds out of the original. A single effect may be used in different ways, at different points in this continuum, depending on the values of parameters that affect its operation. For example, a **reverb** effect is produced digitally by adding copies of a signal, delayed in time and attenuated, to the original. These copies model reflections from surrounding surfaces, with the delay corresponding to the size of the enclosing space, and the degree of attenuation modelling surfaces with different acoustic reflectivity. By using small delays and low reflectivity, a recording can be made to sound as if it had been made inside a small room. This degree of reverb is often a necessary enhancement when the output from electric instruments has been recorded directly without going through a speaker and microphone. Although cleaner recordings

are produced this way, they are often too dry acoustically to sound convincing. Longer reverb times can produce the illusion of a concert hall or a stadium. Still longer times, with the delayed signals being amplified instead of attenuated, can be used creatively to generate sustained rhythm patterns from a single chord or note. Figure 8.14 shows the effect on the spectrum and waveform of adding an echo to our sea sound.

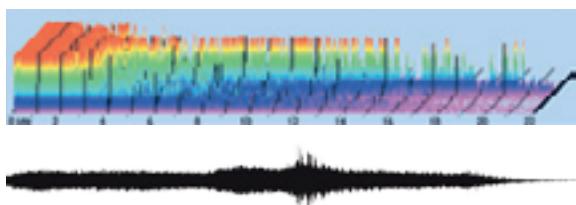


Figure 8.14. Echo reverb

Other effects can be put to a variety of uses in a similar way. These include **graphic equalization**, which transforms the spectrum of a sound using a bank of filters, each controlled by its own slider and each affecting a fairly narrow band of frequencies. These can be used to compensate for recording equipment with idiosyncratic frequency response, or to artificially enhance the bass, for example, to produce a desired frequency balance.

Envelope shaping operations change the outline of a waveform. The most general envelope shapers allow the user to draw a new envelope around the waveform, altering its attack and decay and introducing arbitrary fluctuations of amplitude. Specialized versions of envelope shaping include faders, which allow a sound's volume to be gradually increased or decreased, and tremolo, which causes the amplitude to oscillate periodically from zero to its maximum value.[†]

[†] To classical musicians, “tremolo” means the rapid repetition of a single note – this does produce a periodic oscillation of amplitude. The “tremolo arm” fitted to Fender Stratocasters and other electric guitars actually produces a periodic change of pitch, more accurately referred to as “vibrato”.

Time stretching and **pitch alteration** are two closely related effects that are especially well suited to digital sound. With analogue recordings, altering the duration of a sound could only be achieved by altering the speed at which it was played back, and this altered the pitch. With digital sound, the duration can be changed without altering the pitch, by inserting or removing samples. Conversely, the pitch can be altered without affecting the duration.

347

Time stretching may be required when sound is being synchronized to video or another sound. If, for example, a voice-over is slightly too long to fit over the video scene it describes, the soundtrack can be shrunk in time, without raising the pitch of the speaker's voice, which would happen if the voice track was simply played at a faster speed. Time stretching can also be applied to music, to alter its tempo. This makes it possible to combine loops that were sampled from pieces originally played at different tempos. (Time stretching software is sometimes used to slow down recorded music to make it easier to transcribe.)

Pitch alteration can be used in several ways. It can be applied uniformly to alter the pitch of an instrument, compensating for an out-of-tune guitar, for example. It can be applied periodically to add a vibrato (periodic fluctuation of pitch) to a voice or instrument, or it can be applied gradually, to produce a “bent note”, in the same way a blues guitarist changes the tone of a note by bending the string while it sounds. The all-important shape of the bend can be specified by drawing a curve showing how the pitch changes over time. Pitch alteration can also be used to transpose music into a different key – again, this allows samples from disparate sources to be combined harmoniously.

Beyond these effects lie what are euphemistically called “creative” sound effects. Effects such as flanging, phasing, chorus, ring modulation, reversal, Doppler shift and wah-wah, which were originally pioneered in the 1960s on albums such as the Beatles' *Sergeant Pepper's Lonely Hearts Club Band* and Jimi Hendrix's *Electric Ladyland*, have been reproduced digitally, and joined by new extreme effects such as roboticization. These effects, if used judiciously, can enhance a recording, but they are easily over-used, and are generally best enjoyed in private.

Combining Sound and Picture

When sound is used as part of a video or animation production, synchronization between sound and picture becomes a matter of considerable importance. This is seen most clearly where the picture shows a person talking and the soundtrack contains their speech. If synchronization is slightly out, the result will be disconcerting. If it is substantially out, the result will at best be unintentionally funny, but more likely incoherent. Although speech makes the most exacting demands on synchronization, wherever sound and picture are related it is necessary that the temporal relationship between them is maintained. Voice-overs should match the picture they describe, music will often be related to edits, and natural sounds will be associated with events on screen.

In order to establish synchronization, it is necessary to be able to identify specific points in time. Film is divided into physical frames, which provides a natural means of identifying times. Video does not have physical frames, but – as we mentioned in Chapter 6 – it does have timecode, allowing the frames to be identified precisely.

348

Sound is effectively continuous, though, even in the digital domain. The high sampling rates used for digital sound mean that a single sample defines too short a time interval to be useful. For sound, therefore, the division into frames imposed by timecode is just a useful fiction. This fictional division continues to be used when synching digital audio and video. It enables sound and picture tracks in a video editing application such as Final Cut or Premiere to be arranged on the same timeline.

Unlike the soundtrack on a piece of film or video tape, a sound track in a digital video editing program is physically independent of the video it accompanies, so it is easy to move the sound in time relative to the picture, simply by sliding the sound track along the timeline. This is not something you would normally want to do if the sound and picture had originally been recorded together. In that case, you will usually want to maintain their synchronization during editing. For this purpose, tracks can be locked together, so that, for example, cutting out part of the video track will remove the accompanying part of the sound.

Audio tracks may be displayed as waveforms. When a sound track has been made independently of the picture – a voice-over or musical accompaniment, for example – it will be necessary to fit the sound to the picture. By looking at the waveform to identify the start of syllables in speech, or stressed beats in music, an editor can identify meaningful points in the sound track, which can then be lined up with appropriate picture frames. Performing this matching by eye is difficult, so a method that is often used is to scrub through the sound to identify the precise cue point by ear, and place a marker that can then be lined up on the timeline with the video frame (which can also be marked for identification). Sometimes, it may be necessary to apply a time-stretching filter to adjust the duration of the sound to fit the picture, as described earlier.

Synchronization can thus be established in a video editing program, but it must then be maintained when the video and its soundtrack are played back, possibly over a network. If the sound and video are physically independent – travelling over separate network connections, for example – synchronization will sometimes be lost. This is a fact of life and cannot be avoided. Audio and video data streams must therefore carry the equivalent of timecode, so that their synchronization can be checked and they can be resynched if necessary. Usually, this will require some video frames to be dropped, so that picture can catch up with sound – the greater data rate of the video means that it is video that is more likely to fall behind.

KEY POINTS

For a sampling rate of r Hz and sample size of s bits, each second of digitized sound will occupy $rs/8$ bytes. For CD quality, $r=44.1 \times 10^3$ and $s=16$, so each second occupies just over 86 kbytes (for a mono signal).

If the recording level is too high, clipping will occur, causing distortion.

Sound editing programs use a timeline interface, with multiple tracks (usually displayed as waveforms), which are mixed down to produce a stereo or mono output.

Short loops may be used to create voices for samplers; longer loops may be combined (e.g. in GarageBand) to build songs from repeating sections.

Filters and gates are used to correct defects (e.g. remove noise) or to enhance or modify sounds (e.g. reverb).

Time stretching (slowing down and speeding up) and pitch alteration are more easily applied to digital audio than they were to analogue audio. They are used for synchronization and for matching (e.g. when combining separately recorded loops).

Sound can be combined with pictures in a video editing program: sound tracks are displayed on the same timeline as video tracks, where they can be synchronized.

Timecode is just a fiction when working with sound, owing to the high sampling rate, but it is valuable for synchronization.

If sound and video are physically independent in a movie, synchronization may be lost, especially when it is sent over a network.

349

Compression

While the data rate for CD-quality audio is nothing like as demanding as that for video, lengthy sound recordings rapidly consume disk space. A single three-minute song, recorded in stereo, will occupy over 25 Mbytes. Hence, where audio is used in multimedia, and especially when it is delivered over the Internet, there is a need for compression. The complex and unpredictable nature of sound waveforms makes them difficult to compress using lossless methods. Huffman coding can be effective in cases where the amplitude of the sound mainly falls below the maximum level that can be represented in the sample size being used. In that case, the signal could have been represented in a smaller sample size, and the Huffman algorithm, by assigning short codes to the values it does encounter, will effectively do this automatically. This is a special case, though. In general, some form of lossy compression will be required.

An obvious compression technique that can be applied to speech is the removal of silence. That is, instead of using 44,100 samples with the value of zero for each second of silence (assuming a 44.1 kHz sampling rate) we record the length of the silence. This technique appears to be a special case of run-length encoding, which is lossless (see Chapter 4). However, as Figure 8.1 shows, “silence” is rarely absolute. We would obtain little compression if we simply run-length encoded samples whose value was exactly zero. Instead, we must treat samples falling below a threshold as if they were zero. The effect of doing this is equivalent to applying a noise gate, and is not strictly lossless, since the decompressed signal will not be identical to the original.

The principles behind lossy audio compression are different from those used in lossy image compression, because of the differences in the way we perceive the two media. In particular, whereas the high frequencies associated with rapid changes of colour in an image can safely be discarded, the high frequencies associated with rapid changes of sound are highly significant, so some other principle must be used to decide what data can be discarded.

Speech Compression

Telephone companies have been using digital audio since the early 1960s, and have been forced by the limited bandwidth of telephone lines to develop compression techniques that can be effectively applied to speech. An important contribution of this early work is the technique known as **companding**. The idea is to use non-linear quantization levels, with the higher levels spaced further apart than the low ones, so that quiet sounds are represented in greater detail than louder ones. This matches the way in which we perceive differences in volume.

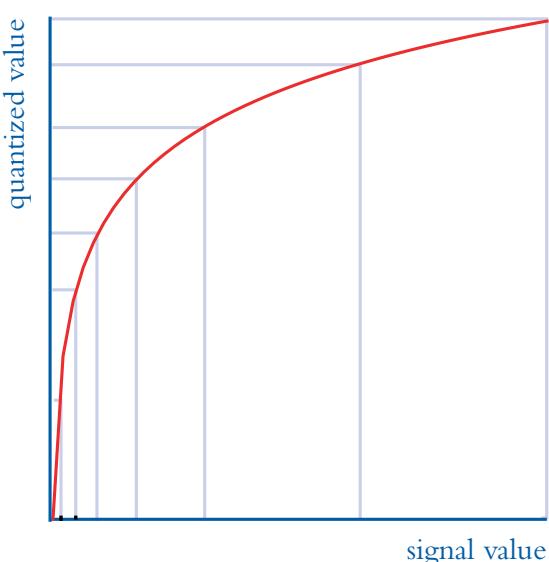


Figure 8.15. Non-linear quantization

Figure 8.15 shows an example of non-linear quantization. The signal value required to produce an increase of one in the quantized value goes up logarithmically. This produces compression, because fewer bits are needed to represent the full range of possible input values than a linear quantization scheme would require. When the signal is reconstructed an inverse process of expansion is required, hence the name “companding” (itself a compressed version of “compressing/expanding”).

Different non-linear companding functions can be used. The principal important ones are defined by ITU Recommendations for use in telecommunications. Recommendation G.711 defines a function called the **μ -law**, which is used in North America and Japan. This

companding method is used in AU files. A different ITU Recommendation is used in the rest of the world, based on a function known as the ***A-law***.

Telephone signals are usually sampled at only 8 kHz. At this rate, μ -law compression is able to squeeze a dynamic range of 12 bits into just 8 bits, giving a one-third reduction in data rate.

351

IN DETAIL

The μ -law is defined by the equation:

$$y = \log(1 + \mu x) / \log(1 + \mu) \text{ for } x \geq 0$$

where μ is a parameter that determines the amount of companding; $\mu=255$ is used for telephony.

The *A*-law is:

$$y = \begin{cases} Ax / (1 + \log A) & \text{for } 0 \leq |x| < 1/A \\ (1 + \log Ax) / (1 + \log A) & \text{for } 1/A \leq |x| < 1 \end{cases}$$

Another important technique that was originally developed for – and is widely used in – the telecommunications industry is ***Adaptive Differential Pulse Code Modulation (ADPCM)***.[†] This is related to inter-frame compression of video, in that it is based on storing the difference between consecutive samples, instead of the absolute value of each sample. Because of the different nature of audio and video, and its origins in hardware encoding of transmitted signals, ADPCM works somewhat less straightforwardly than a simple scheme based on the difference between samples.

Storing differences will only produce compression if the differences can be stored in fewer bits than the sample. Audio waveforms can change rapidly, so, unlike consecutive video frames, there is no reason to assume that the difference will necessarily be much less than the value. Basic ***Differential Pulse Code Modulation (DPCM)*** therefore computes a predicted value for a sample, based on preceding samples, and stores the difference between the prediction and the actual value. If the prediction is good, the difference will be small.

Adaptive DPCM obtains further compression by dynamically varying the step size used to represent the quantized differences. Large differences are quantized using large steps, small differences using small steps, so the amount of detail that is preserved scales with the size of the difference. The details of how this is done are complicated, but as with companding, the effect is to make efficient use of bits to store information, taking account of its rate of change.

[†] “Pulse Code Modulation” is the term used in audio and communications circles for encoding digital data as a sequence of pulses representing ones and zeros. Whereas this is more or less the only sensible representation for computer use, alternatives, such as “Pulse Width Modulation”, exist where the data is to be represented as a stream for transmission, rather than as stored values.

ITU Recommendation G.721 specifies a form of ADPCM representation for use in telephony, with data rates of 16 kbps and 32 kbps. Lower rates can be obtained by a much more radical approach to compression. **Linear Predictive Coding** uses a mathematical model of the state of the vocal tract as its representation of speech. Instead of transmitting the speech as audio samples, it sends parameters describing the corresponding state of the model. At the receiving end, these parameters can be used to construct the speech, by applying them to the model. The details of the model and how the parameters are derived from the speech lie beyond the scope of this book. Speech compressed in this way can be transmitted at speeds as low as 2.4 kbps. Because the sound is reconstructed algorithmically, it has a machine-like quality, so it is only suitable for applications where the content of the speech is more important than a faithful rendition of someone's voice.

Perceptually Based Compression

The secret of effective lossy compression is to identify data that doesn't matter – in the sense of not affecting perception of the signal – and to throw it away. If an audio signal is digitized in a straightforward way, data corresponding to sounds that are inaudible may be included in the digitized version. This is because the signal records all the physical variations in air pressure that cause sound, but the perception of sound is a sensation produced in the brain, via the ear, and the ear and brain do not respond to the sound waves in a simple way.

Two phenomena in particular cause some sounds not to be heard, despite being physically present. Both are familiar experiences: a sound may be too quiet to be heard, or it may be obscured by some other sound. Neither phenomenon is quite as straightforward as it might appear.

The **threshold of hearing** is the minimum level at which a sound can be heard. It varies non-linearly with frequency, as shown in Figure 8.16. A very low- or very high-frequency sound must

be much louder than a mid-range tone in order to be heard. It is surely no coincidence that we are most sensitive to sounds in the frequency range that corresponds to human speech. When compressing sound, there is no point in retaining sounds that fall below the threshold of hearing, so a compression algorithm can discard the corresponding data. To do this, the algorithm must use a **psycho-acoustical model** – that is, a mathematical description of aspects

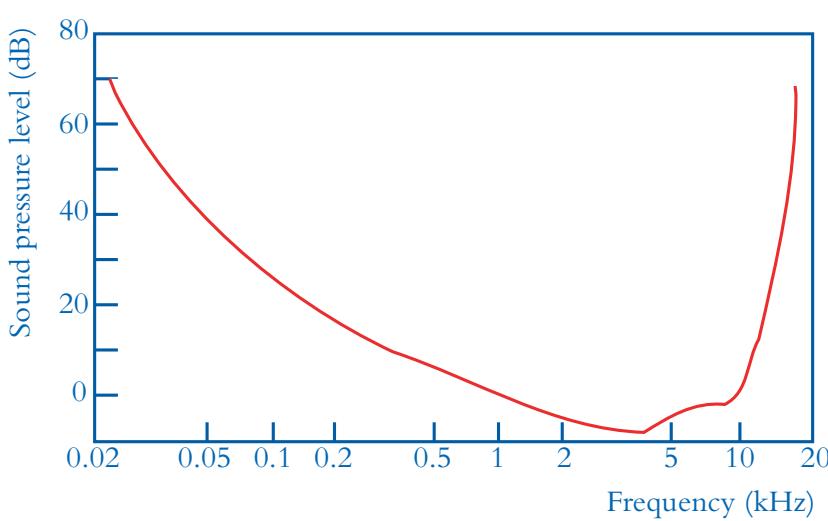


Figure 8.16. The threshold of hearing

of the way the ear and brain perceive sounds. In this case, what is needed is a description of the way the threshold of hearing varies with frequency.

Loud tones can obscure softer tones that occur at the same time. In fact, they can also obscure softer tones that occur a little later or – strange as it may seem – slightly earlier. This is not simply a case of the loud tone “drowning out” the softer one; the effect is more complex, and depends on the relative frequencies of the two tones. **Masking**, as this phenomenon is known, can be conveniently described as a modification of the threshold of hearing curve in the region of a loud tone. As Figure 8.17 shows, the threshold is raised in the neighbourhood of the masking tone. The raised portion, or **masking curve**, is non-linear and asymmetrical, rising faster than it falls. Any sound that lies within the masking curve will be inaudible, even though it rises above the unmodified threshold of hearing. Thus, there is an additional opportunity to discard data. Masking can be used more cleverly, though. Because masking hides noise as well as some components of the signal, quantization noise can be masked. Where a masking sound is present, the signal can be quantized relatively coarsely, using fewer bits than would otherwise be needed, because the resulting quantization noise can be hidden under the masking curve.

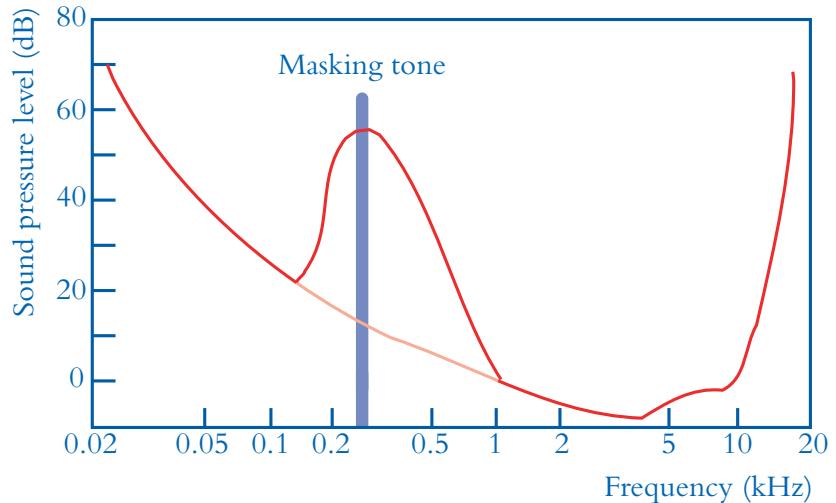


Figure 8.17. *Modification of the threshold of hearing by a masking tone*

It is evident that the phenomena just described offer the potential for additional compression. It is not obvious how a compression algorithm can be implemented to take advantage of this potential. The approach usually adopted is to use a bank of filters to split the signal into bands of frequencies; 32 bands are commonly used. The average signal level in each band is calculated, and using these values and a psycho-acoustical model, a masking level for each band is computed. That is, it is assumed that the masking curve within each band can be approximated by a single value. If the signal in a band falls entirely below its masking level, that band is discarded. Otherwise, the signal is quantized using the least number of bits that causes the quantization noise to be masked.

Turning the preceding sketch into a working algorithm involves many technical details that lie beyond the scope of this book. The best-known algorithms that have been developed are those specified for audio compression in the MPEG standards. MPEG-1 and MPEG-2 are primarily

video standards, but, since most video has sound associated with it, they also include audio compression. MPEG audio has been so successful that it is often used on its own purely for compressing sound, especially music.

354

MPEG-1 specifies three “layers” of audio compression. All three layers are based on the principles just outlined. The encoding process increases in complexity from Layer 1 to Layer 3, while as a result, the data rate of the compressed audio decreases. The quality obtained at 192 kbps for each channel at Layer 1 only needs 128 kbps at Layer 2, and 64 kbps at Layer 3. (These data rates will be doubled for stereo.) MPEG-1 Layer 3 audio, or **MP3** as it is usually called,[†] achieves compression ratios of around 10:1 while maintaining high quality. A typical track from a CD can be compressed to under 3 Mbytes. The sound quality at this rate is sometimes claimed to be “CD quality”, but this is something of an exaggeration. Higher bit rates can be used at Layer 3, however, giving correspondingly better quality. Variable bit rate (VBR) coding is also possible, with the bit rate being changed, so that passages which do not compress easily can be encoded at a higher rate than those which do. MP3 can also encode audio at lower bit rates, for example for streaming. At 64 kbps, stereo quality is claimed to be as good as FM radio.

The audio part of the MPEG-2 standard includes an encoding that is essentially identical with MPEG-1 audio, except for some extensions to cope with surround sound. The MPEG-2 standard also defined a new audio codec, **Advanced Audio Coding (AAC)**. AAC is also incorporated in MPEG-4, and is most often considered part of that standard. Unlike MP3, AAC is not backwards compatible with earlier MPEG standards, or lower layers. By abandoning backwards compatibility, AAC was able to achieve higher compression ratios at lower bit rates than MP3. Like MP3, AAC is based on perceptual coding, but it uses additional techniques and a more complicated implementation. Subjective listening tests consistently rate AAC quality as superior to MP3 at the same bit rates, and the same subjective quality is attained by AAC at lower rates than MP3. For instance, AAC audio at 96 kbps is considered to be superior to MP3 at 128 kbps. AAC is the codec used for audio distributed over the Internet from the popular iTunes service.

Lossy compression always sounds like a dubious practice – how can you discard information without affecting the quality? In the case of MPEG audio, the argument is that the information that has been discarded is inaudible. This contention is based on extensive listening tests, and is supported by the rapid acceptance of MP3 and AAC as formats for downloading music. (However, it should be remembered that some people care much more about audio quality than others.) As with any lossy form of compression, though, MPEG audio will deteriorate progressively if it is decompressed and recompressed a number of times. It is therefore only suitable as a delivery format, and should not be used during production, when uncompressed audio should be used whenever possible.

[†] Despite what you may sometimes read, MP3 does not stand for MPEG-3. There is no MPEG-3.

KEY POINTS

Sound is difficult to compress using lossless methods, except for special cases.

Some compression of audio can be obtained by run-length encoding samples that fall below a threshold that can be considered to represent silence.

Companding uses non-linear quantization to compress speech. μ -law and A-law companding are used for telephony.

Adaptive Differential Pulse Code Modulation (ADPCM), which works by storing information about the difference between a sample and a value predicted from the preceding sample, is also used in telephony.

Perceptually based compression discards inaudible sounds.

A psycho-acoustical model describes how the threshold of hearing varies non-linearly with frequency.

Masking is a modification of the threshold of hearing curve in the region of a loud tone. The threshold is raised in the neighbourhood of the masking tone.

Filters are used to split a signal into 32 bands, and a masking level for each band is computed. Signals that fall below the level can be discarded.

Practical implementations of perceptually based compression are the basis of MP3 and AAC compression.

355

MIDI

If we had written a piece of music, there are two ways we could send it to you. We could play it, record the performance, and send you the recording, or we could write it down using some form of notation, indicating the arrangement, and send you the sheet music, so that you could play the piece for yourself. In the first case, we send you the actual sound. In the second, we send you what amounts to a set of instructions telling you how to produce the sound. In either case we are making some assumptions about what you can do. For the recording, we assume you have a machine capable of playing back whichever medium we have recorded our performance on. For the sheet music, we are making the more demanding assumptions that you can read our chosen music notation, have access to the instrument or instruments indicated in the arrangement, and can either play yourself or get musicians to play those instruments. If the music is arranged for a symphony orchestra, this might present some difficulties for you, whereas if we were to send a recording, all the difficulties would lie at our end.

In the digital realm, there is a similar choice of options for delivering music. So far, we have considered ways of delivering digitized sound, that is, the equivalent of recordings. There also

exists an equivalent to delivering the sheet music, i.e. a way of delivering instructions about how to produce the music which can be interpreted by suitable software or hardware. Similar assumptions must be made. For sound files, you must have software that can read them – but as we have seen, this is not a demanding requirement. For instructions, you must have software that can interpret the instructions, and some means of producing sounds that correspond to the appropriate instruments.

MIDI (Musical Instruments Digital Interface) provides a basis for satisfying these requirements. Originally, MIDI was devised as a standard protocol for communicating between electronic instruments, such as synthesizers, samplers and drum machines.

By defining a standard hardware interface, and a set of instructions indicating such things as the start and end of a note, it provided a means of controlling a collection of such instruments from a single keyboard. This removed the requirement for huge banks of keyboards, and opened the way for playing traditional keyboard instruments, particularly synthesizers, with other controllers, such as drum pads or wind instruments.

More significantly, perhaps, MIDI allowed instruments to be controlled automatically by devices that could be programmed to send out sequences of MIDI instructions. Originally, **sequencers**, as these devices are known, were dedicated hardware devices, programmed using their own built-in, relatively clumsy interfaces. It was not long before it was realized that computer programs could offer a more convenient and flexible means of sequencing, provided that a computer could be fitted with a MIDI interface so that it could send the necessary signals to other MIDI devices. Such an interface is a relatively simple and inexpensive device, so computer-based sequencers rapidly became available. A software sequencer provides editing and compositional functions, so it needs to store MIDI sequences in files. This requirement led to the development of a standard file format for MIDI files – that is, a way of storing MIDI on disk. Clearly, such files can be exchanged between computers equipped with MIDI software. They can also be incorporated into multimedia.

Playing back MIDI files requires an instrument that understands MIDI, but a computer, equipped with suitable hardware or software, can be such an instrument itself. Sounds can be either synthesized on a sound card, or held on disk in the form of samples, to be played back in response to MIDI instructions. MIDI files are therefore a means of communicating music. Because they do not contain any audio data, they can be much more compact than actual digitized sound files. For the same reason, though, they cannot guarantee the same fidelity. The samples available when the file is produced may be of higher quality than those used to play it back – just as the musician who plays a piece of music from a score may not be sufficiently accomplished to realize the composer's intentions. In both cases, the result is unpredictable.

MIDI Messages

A **MIDI message** is an instruction that controls some aspect of the performance of an instrument. Messages are encoded in much the same way as machine instructions: a status byte indicates the type of the message, and is followed by one or two data bytes giving the values of parameters. Although wind instruments, drum pads and guitars are used as MIDI controllers (as devices that transmit MIDI signals are called), MIDI is markedly biased towards keyboard instruments. Thus, for example, the most commonly used message is “Note On”, which takes two parameters. The first is a number between 0 and 127 indicating the note to be sounded, where consecutive numbers represent notes that are a semi-tone apart, like keys on a piano. The second parameter is a key velocity, indicating how fast the key was pressed, and hence the attack of the note. When an actual keyboard is being used to generate MIDI messages, these values will be sensed by the keyboard’s hardware as the musician plays the key. When the message is being generated by software, the values are specified by the user.

Other significant MIDI messages include “Note Off”, which ends a note, “Key Pressure”, which indicates the degree of “aftertouch” to be applied, and “Pitch Bend”, which changes note values dynamically, as a guitarist does by bending the string (on MIDI keyboards, a wheel is used for this function).

The status bytes and data bytes in a stream of MIDI instructions are distinguishable by the value of their most significant bit. This makes an optimization possible – where a sequence of messages all have the same status byte, it may be omitted from the second and subsequent messages, for which it will be inferred from the most recent value. This arrangement is called “running status”; it can save an appreciable number of bytes where a sequence of notes is being played with no modifications. Using the convention that the end of a note can be indicated by a “Note On” message with a velocity of zero, the whole sequence can consist of a single “Note On” status byte, followed by a series of data bytes giving the notes to be played and the velocities to be applied to them.

When a MIDI message is interpreted, we say that an event occurs. In a live performance, the timing of events is determined by the player in real time. In a MIDI file, it is necessary to record the time of each event. Each message is preceded by a “delta time”, that is, a measure of the time since the preceding event. Near the beginning of the file is a specification of the units to be used for times.

General MIDI

The preceding account indicates how notes are produced, but leaves unanswered the question of how they are to be associated with particular sounds. Typically, the sorts of instruments controlled by MIDI – synthesizers and samplers – provide a variety of “voices”. In the case of synthesizers, these are different synthesized sounds, often called “patches” by synthesists. In the case of samplers,

they are different instrument samples. A MIDI “Program Change” message selects a new voice, using a value between 0 and 127. The mapping from these values to voices is not specified in the MIDI standard, and may depend on the particular instrument being controlled. There is thus a possibility that a MIDI file intended to specify a piece for piano and violin might end up being played on trombone and kettle drum, for example. To help overcome this unsatisfactory situation, an addendum to the MIDI standard, known as **General MIDI**, was produced, which specifies 128 standard voices to correspond to the values used by “Program Change” messages. The assignments are shown in Figure 8.18. For drum machines and percussion samplers, Program Change values are interpreted differently, as elements of drum kits – cymbals of various sorts, snares, tom-toms, and so on – as shown in Figure 8.19.

General MIDI only associates program numbers with voice names. There is no guarantee that identical sounds will be generated for each name by different instruments. A cheap sound card may attempt to synthesize all of them, while a good sampler may use high-quality samples of the corresponding real instruments. However, adherence to General MIDI offers some guarantee of consistency, which is otherwise entirely missing.

QuickTime incorporates MIDI-like functionality. QuickTime Musical Instruments provides a set of instrument samples, and the QuickTime Music Architecture incorporates a superset of the features of MIDI. QuickTime can read standard MIDI files, so any computer with QuickTime installed can play MIDI music using software alone. QuickTime can also control external MIDI devices. MIDI tracks can be combined with audio, video or any of the other media types supported by QuickTime.

MIDI Software

MIDI sequencing programs, such as Cakewalk Metro and Cubase, perform capture and editing functions equivalent to those of video editing software. They support multiple tracks, which can be allocated to different voices, thus allowing polytimbral music to be constructed. In addition, such packages support composition.

Music can be captured as it is played from MIDI controllers attached to a computer via a MIDI interface. The sequencer can generate metronome ticks to help the player maintain an accurate tempo. Although it is common to use the sequencer simply as if it were a tape recorder, to capture a performance in real time, sometimes MIDI data is entered one note at a time, which allows musicians to “play” music that would otherwise be beyond their competence. Facilities normally found in conventional audio recording software are also available, in particular the ability to “punch in” – the start and end point of a defective passage are marked, the sequencer starts playing before the beginning, and then switches to record mode, allowing a new version of the passage to be recorded to replace the original.

1	Acoustic Grand Piano	44	Contrabass	87	Synth Lead 7
2	Bright Acoustic Piano	45	Tremolo Strings	88	Synth Lead 8
3	Electric Grand Piano	46	Pizzicato Strings	89	Synth Pad 1
4	Honky-tonk Piano	47	Orchestral Harp	90	Synth Pad 2
5	Rhodes Piano	48	Timpani	91	Synth Pad 3
6	Chorused Piano	49	Acoustic String Ensemble 1	92	Synth Pad 4
7	Harpsichord	50	Acoustic String Ensemble 2	93	Synth Pad 5
8	Clavinet	51	Synth Strings 1	94	Synth Pad 6
9	Celesta	52	Synth Strings 2	95	Synth Pad 7
10	Glockenspiel	53	Aah Choir	96	Synth Pad 8
11	Music Box	54	Ooh Choir	97	Ice Rain
12	Vibraphone	55	Synvox	98	Soundtracks
13	Marimba	56	Orchestra Hit	99	Crystal
14	Xylophone	57	Trumpet	100	Atmosphere
15	Tubular bells	58	Trombone	101	Bright
16	Dulcimer	59	Tuba	102	Goblin
17	Draw Organ	60	Muted Trumpet	103	Echoes
18	Percussive Organ	61	French Horn	104	Space
19	Rock Organ	62	Brass Section	105	Sitar
20	Church Organ	63	Synth Brass 1	106	Banjo
21	Reed Organ	64	Synth Brass 2	107	Shamisen
22	Accordion	65	Soprano Sax	108	Koto
23	Harmonica	66	Alto Sax	109	Kalimba
24	Tango Accordion	67	Tenor Sax	110	Bagpipe
25	Acoustic Nylon Guitar	68	Baritone Sax	111	Fiddle
26	Acoustic Steel Guitar	69	Oboe	112	Shanai
27	Electric Jazz Guitar	70	English Horn	113	Tinkle bell
28	Electric clean Guitar	71	Bassoon	114	Agogo
29	Electric Guitar muted	72	Clarinet	115	Steel Drums
30	Overdriven Guitar	73	Piccolo	116	Woodblock
31	Distortion Guitar	74	Flute	117	Taiko Drum
32	Guitar Harmonics	75	Recorder	118	Melodic Tom
33	Wood Bass	76	Pan Flute	119	Synth Tom
34	Electric Bass Fingered	77	Bottle blow	120	Reverse Cymbal
35	Electric Bass Picked	78	Shakuhachi	121	Guitar Fret Noise
36	Fretless Bass	79	Whistle	122	Breath Noise
37	Slap Bass 1	80	Ocarina	123	Seashore
38	Slap Bass 2	81	Square Lead	124	Bird Tweet
39	Synth Bass 1	82	Saw Lead	125	Telephone Ring
40	Synth Bass 2	83	Calliope	126	Helicopter
41	Violin	84	Chiffer	127	Applause
42	Viola	85	Synth Lead 5	128	Gunshot
43	Cello	86	Synth Lead 6		

359

Figure 8.18. General MIDI voice numbers

35	Acoustic Bass Drum	47	Low Mid Tom Tom	59	Ride Cymbal 2
36	Bass Drum 1	48	Hi Mid Tom Tom	60	Hi Bongo
37	Side Stick	49	Crash Cymbal 1	61	Low Bongo
38	Acoustic Snare	50	Hi Tom Tom	62	Mute Hi Conga
39	Hand Clap	51	Ride Cymbal 1	63	Open Hi Conga
40	Electric Snare	52	Chinese Cymbal	64	Low Conga
41	Lo Floor Tom	53	Ride Bell	65	Hi Timbale
42	Closed Hi Hat	54	Tambourine	66	Lo Timbale
43	Hi Floor Tom	55	Splash Cymbal		
44	Pedal Hi Hat	56	Cowbell		
45	Lo Tom Tom	57	Crash Cymbal 2		
46	Open Hi Hat	58	Vibraslap		

Figure 8.19. General MIDI drum kit numbers

Sequencers will optionally *quantize* tempo during recording, fitting the length of notes to exact sixteenth notes, or eighth note triplets, or whatever duration is specified. This allows rhythmically loose playing to be brought into strict tempo, which may be felt desirable for certain styles of music, but the result often has an unnatural machine-like quality, since live musicians very rarely play so precisely to the beat.

Most programs allow music to be entered using classical music notation, often by dragging and dropping notes and other symbols onto a stave. Some programs allow printed sheet music to be scanned, and will perform optical character recognition to transform the music into MIDI. The opposite transformation, from MIDI to a printed score, is also often provided, enabling transcriptions of performed music to be made automatically. (In this case, quantization is usually necessary, since otherwise the program will transcribe exactly what was played, even if that involves dotted sixty-fourth notes and rests.) Those who do not read music usually prefer to use the “piano-roll” interface, which allows the duration of notes to be specified graphically, essentially using a timeline. For music which is constructed out of repeating sections, loops can be defined and reused many times.

Once a piece of music has been recorded or entered, it can be edited. Individual notes’ pitch and duration can be altered, sections can be cut and pasted, or global changes can be made, such as transposing the entire piece into a different key, or changing the time signature. The parameters of individual MIDI events can be changed – the velocity of a note can be altered, for example. Voices can be changed to assign different instruments to the parts of the arrangement.

Because digital audio is very demanding of computer resources but MIDI is much less so, the two forms of music representation were originally separated, with different software being used for each. Now that computers have become powerful enough to take audio in their stride, the two

are commonly integrated in a single application, which allows MIDI tracks to be combined and synchronized with full audio. This arrangement overcomes one of the major limitations of MIDI, namely the impossibility of representing vocals (except for “Oohs” and “Aahs”). MIDI can be transformed into audio, much as vector graphics can be rasterized and transformed into pixels. The reverse transformation is sometimes supported, too, although it is more difficult to implement. MIDI captures the musical structure of sound, since MIDI events correspond to notes. Being able to transform audio into MIDI allows music to be recorded from ordinary instruments instead of MIDI controllers – it can even be recorded from somebody’s whistling – and then edited or transcribed in terms of musical notes.

361

KEY POINTS

MIDI provides a way of representing music as instructions describing how to produce notes, instead of as a record of the actual sounds.

MIDI provides a standard protocol and hardware interface for communicating between electronic instruments, such as synthesizers, samplers and drum machines, allowing instruments to be controlled by hardware or software sequencers.

A computer can control instruments through a MIDI interface, synthesize notes on a sound card or play back samples from disk in response to MIDI instructions.

MIDI messages are instructions that control some aspect of the performance of an instrument.

Each instruction has a status byte, indicating the type of message, and two data bytes, providing the values of its parameters. (e.g. Note On + note number + key velocity.)

Running status allows the status byte to be omitted if it is the same as in the preceding message.

General MIDI is a standard association between 128 Program Change values and voice names. (There is no guarantee that identical sounds will be produced for the same voice names on different instruments.)

QuickTime incorporates MIDI-like functionality. MIDI tracks can be combined with audio, video or any of the other media types supported by QuickTime.

MIDI software allows recording from a MIDI device, input as musical notation or on a “piano roll”, and editing, often integrated with sound editing.