chapter 1

User Experience and Why It Matters

Technology has been a part of our everyday environment for generations. It empowers us and frustrates us; it simplifies and complicates our lives; it separates us and brings us closer together. But even though we interact with technology every day, we easily forget that technology products are made by people, and that someone, somewhere should get the credit when technology works well for us—or get the blame when it doesn't.

Everyday Miseries

Everyone, every once in a while, has one of those days.

You know the kind of day I'm talking about: You wake up to sunlight streaming in your window and wonder why your alarm clock hasn't gone off yet. You look over to see that your clock thinks it's 3:43 a.m. You stumble out of bed to find another clock, which tells you that you can still make it to work on time—if you leave in 10 minutes.

You turn on the coffeemaker and hustle to get dressed, but when you go to retrieve your dose of life-sustaining caffeine, there's no coffee in the pot. No time to figure out why—you've got to get to work!

You get about a block from your house when you realize that the car needs gas. At the gas station, you try to use the one pump that takes ATM cards, but this time it won't accept your card. So you have to go inside and pay the cashier, but first you have to wait in a line while the cashier very slowly helps everyone in front of you. Finally, you pump your gas and leave the station—to the sound of your gas cap tumbling off the roof of your car and bouncing out into the street.

You have to take a detour because of a traffic accident, so the drive to work takes longer than you expected. It's official: Despite all your efforts, you are now late for work. Finally, you make it to your desk. You're agitated, harried, weary, and irritable—and your day hasn't even really started yet. And you still haven't had any coffee.

Introducing User Experience

It seems like a string of bad luck—just one of those days. But let's look closer and see if, somehow, all that bad luck could have been avoided:

The accident: The accident on the road happened because the driver took his eyes off the road for a moment to turn the radio down. He had to look down because it was impossible to identify which was the volume control by touch alone.

The cap: You lost your gas cap because you set it down on the roof of the car when you started to pump, but then you forgot it because you were feeling rushed. If the cap didn't need to be set down somewhere at all—if it were simply tethered to the car in some way—you couldn't have lost it.

The register: The line at the register in the gas station moved so slowly because the cash register was complex and confusing, and unless the clerk paid extra-close attention while ringing something up, he would make a mistake and have to start all over again. If the register had been simpler and the layout and colors of the buttons different, that line never would have formed.

The pump: You wouldn't have had to stand in that line at all if the pump had accepted your ATM card. It would have done so if you had turned the card around the other way, but nothing on the pump indicated which way the card should be turned, and you were in such a hurry that you didn't think to try every orientation.

The coffeemaker: The coffeemaker didn't make coffee because you didn't push down the On button all the way. The machine doesn't do anything to let you know that it has been turned on: no light, no sound, no little "click" when the button makes contact. You thought you had turned it on, but you were wrong. The problem could have been avoided altogether if you had set the coffeemaker to start brewing automatically first thing in the morning, but you never learned how to use that function—if you knew it existed at all.

The clock: And then, we come to the factor that started the whole chain of events—the alarm clock. The alarm didn't go off because the time was wrong. The time was wrong because your cat stepped on the clock in the middle of the night and reset it for you. (If this sounds implausible to you, don't laugh—it has happened to me. I have had to go to surprising lengths to find a clock that is impervious to cat meddling.) A slightly different configuration of buttons would have prevented the cat from resetting the clock, and consequently you would have been out of bed with plenty of time—no need to rush at all.

In short, every one of the previous incidents of "bad luck" could have been avoided had someone taken more care in designing a product. These examples all demonstrate a lack of attention to the **user experience**: how the product behaves and is used in the real world. When a product is being developed, people pay a great deal of attention to what it does. User experience is the other, often overlooked, side of the equation—how it works—that can often make the difference between a successful product and a failure.

User experience is not about how a product works on the inside (although that sometimes has a lot of influence). User experience is about how it works on the outside, where a person comes into contact with it and has to work with it. That interaction often involves pushing a lot of buttons, as in the case of technology products such as alarm clocks, coffeemakers, or cash registers. Sometimes, it's just a matter of a simple physical mechanism, such as the gas cap on your car. However, every product that is used by someone has a user experience: newspapers, ketchup bottles, reclining armchairs, cardigan sweaters.

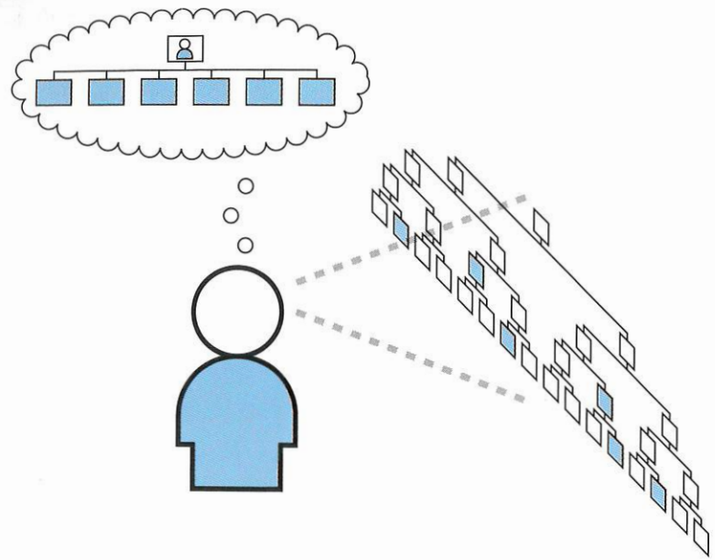
No matter what kind of product is involved, it's the little things that count. Having a button click when you push it down doesn't seem like much, but when that click makes the difference between getting coffee and not getting coffee, it matters a great deal. Even if you never realized that the design of that button was causing you trouble, how would you feel about a coffeemaker that you were able to use successfully only part of the time? How would you feel about the manufacturer? Would you buy another product from that company in the future? Probably not. Thus, for the want of a button that clicks, a customer is lost.

User Experience and the Web

This book is about the user experience of one particular kind of product: Web sites. On the Web, user experience becomes even more important than it is for other kinds of products.

In virtually every case, a Web site is a "self-service" product. There is no instruction manual to read beforehand, no training seminar to attend, no customer service representative to help guide the user through the site. There is only the user, facing the site alone with only her wits and experience to guide her.

It's bad enough that she's been stuck in the position of having to figure out the site on her own. The fact that most sites don't even acknowledge this only makes matters worse. Despite the vital strategic importance of user experience to the success of a Web site, the simple matter of understanding what people want and need has been a low priority for most of the history of the Web.



Faced with a wide array of choices, the user is left to her own devices to determine which features of a site will meet her needs.

How did this come to pass? In the earliest days of the Web, many thought being first to market was the key to success. Sites like Yahoo! built early leads that later competitors struggled to overcome. Established companies raced to set up Web sites, determined not to be perceived as falling behind the times. But in most cases, companies considered merely having deployed the site a great accomplishment; whether the site actually worked for people was, at best, an afterthought.

To gain market share against these "first-mover" sites, competitors began to emphasize features, adding more and more content and functionality to their sites in hopes of drawing in those who were new to the Web (and maybe stealing a few customers from the competition).

Having more features, however, turned out to be only a temporary source of competitive advantage. With the added complexity that came with an ever-expanding feature set, sites became increasingly unwieldy, hard to use, and unappealing to the very first-timers they were supposed to draw in. And still, many organizations paid little attention to what actual users liked, found valuable, or were really able to use.

Businesses have now come to recognize that providing a quality user experience is an essential, sustainable competitive advantage. It is user experience that forms the customer's impression of the company's offerings, it is user experience that differentiates the company from its competitors, and it is user experience that determines whether your customer will ever come back.

Competitive Advantage and ROI

Maybe you don't sell anything on your site. All you provide is information about your company. It might seem that you have a monopoly on that information—if people want it, they have to get it from you. You don't have competition in the same way that an online bookstore does. Nevertheless, you can't afford to neglect the user experience of your site.

If your site consists mainly of what we Web types call "content"—that is, information—then one of the main goals of your site is to communicate that information as effectively as possible. It's not enough just to put it out there. It has to be presented in a way that

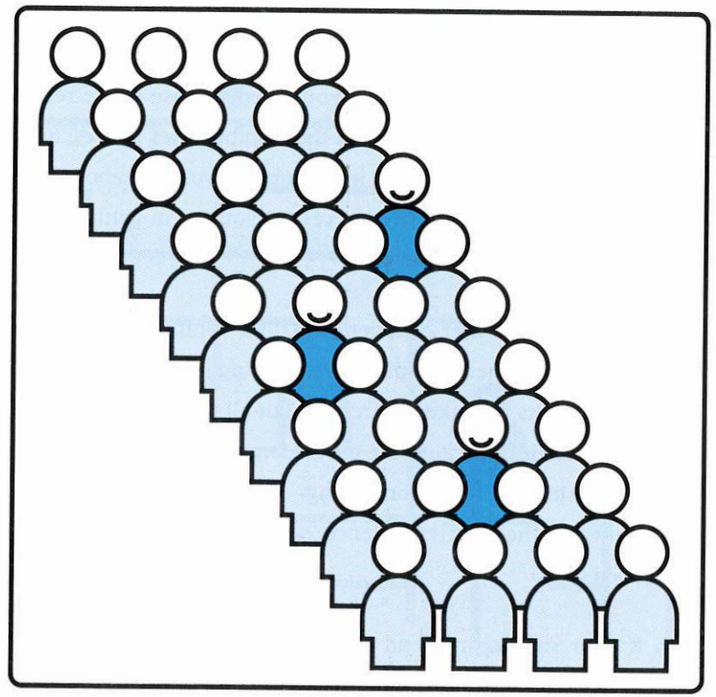
helps people absorb it and understand it. Otherwise, the user might not ever find out that you offer the service or product they're looking for. And even if they do manage to find that information, they're likely to draw the conclusion that if your site is difficult to work with, you probably are as well.

Even if your site mostly comprises interactive tools that people can use to accomplish certain tasks (like buying airplane tickets or managing bank accounts), effective communication is a key factor in the success of your product. The world's most powerful functionality will falter and fail if users can't figure out how to make it work.

Simply put, if your users have a bad experience, they won't come back. If they have an okay experience with your site but a better experience with your competitor's site, they'll go back to that competitor, not you. Features and functions always matter, but user experience has a far greater effect on customer loyalty. All your sophisticated technology and corporate messaging won't bring those customers back a second time. A good user experience will—and you don't get much of a second chance to get it right.

Customer loyalty isn't the only way that focusing on the user experience of your site can pay off. Businesses with an eye on the bottom line want to know about the **return on investment**, or ROI. ROI is usually measured in terms of money: For every dollar you spend, how many dollars of value are you getting back? That's the ROI. But return on investment does not have to be expressed in strictly monetary terms. All you need is a measurement that shows that your money going out translates into value for your company.

One common measure of return on investment is **conversion rate**. Any time you want to encourage your customers to take the next step in building a relationship with you—whether that involves something as complex as customizing the site to their preferences or as simple as signing up to receive an e-mail newsletter—there's a conversion rate you can measure. By keeping track of what percentage of users you "convert" to the next level, you can measure how effectively your site is meeting your business goals.



Conversion rate is a common way of measuring the effectiveness of a user experience.

3 subscription sign-ups

÷

36 visitors

=

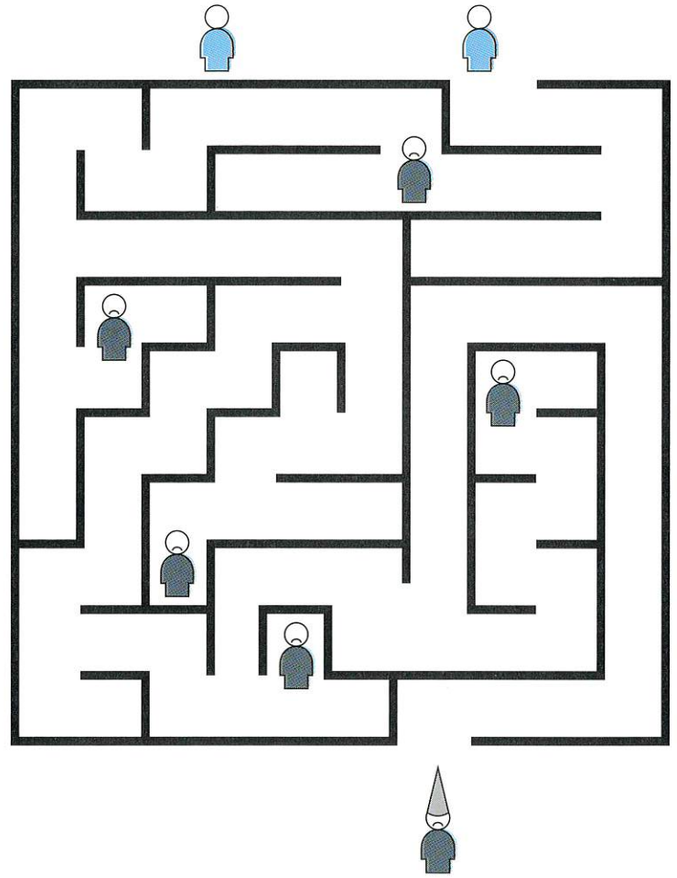
Conversion rate becomes even more important in the case of commerce sites. Far more people browse a commerce site than buy from it. A quality user experience is a key factor in "converting" these casual browsers into active buyers. Even a tiny increase in your conversion rate can translate into a dramatic leap in revenue. It's not uncommon for a change in conversion rate of only one-tenth of one percent to result in a revenue increase of ten percent or more.

On any site where users have the opportunity to give you some money, you have a conversion rate you can measure, whether you're selling books, cat food, or subscriptions to the content of the site itself. Conversion rate can give you a better sense of the return on your user experience investment than simple sales figures. Sales can suffer if you're not successful in getting the word out about your site. Conversion rate tracks how successful you are in getting those who visit to spend some money.

A related measure of user experience ROI for commerce sites is the incidence of abandoned shopping carts. Putting items in a cart indicates a willingness to make a purchase, but all too frequently those carts simply languish because the purchase process itself was too difficult, confusing, or time-consuming. As with conversion rate, refinements to the user experience can reduce the incidence of abandoned carts.

Even if your site doesn't lend itself readily to ROI measures like these, it doesn't mean the effect of user experience on your business is any less significant. Whether they are used by your customers, your partners, or your employees, Web sites can have all kinds of indirect effects on the bottom line.

Web sites are complicated pieces of technology, and something funny happens when people have trouble using complicated pieces of technology: they blame themselves. They feel like they must have done something wrong. They feel like they weren't paying enough attention. They feel stupid. Sure, it's irrational. After all, it's not their fault the site doesn't work the way they expect it to. But they feel stupid anyway. And if you intend to drive people away from your site, it's hard to imagine a more effective approach than making them feel stupid when they use it.



Technology products that don't work the way people expect make them feel stupid—even if they actually accomplished what they set out to do.

No one outside your company might ever see your site (as in the case of an intranet), but the user experience still makes a huge difference. Often, it can mean the difference between a project that creates value for the organization and a project that becomes a resource-consuming nightmare.

Any user experience effort aims to improve efficiency. This basically comes in two key forms: helping people work faster and helping them make fewer mistakes. Improving the efficiency of the tools you use improves the productivity of the business as a whole. The less time it takes to complete any given task, the more you can get done in a day. In keeping with the old notion that time is money, saving your employees time translates directly into saving your business money.

Efficiency doesn't only affect the bottom line, though. People like their jobs more when their tools are natural and easy to use, not frustrating and needlessly complex. If that person is you, these kinds of tools make the difference between coming home at the end of the day satisfied and coming home exhausted. (Or at least, if you are coming home exhausted, it's for the right reasons, not because you've been struggling with your tools.)

If that person is your employee, providing these kinds of tools increases not only their productivity, but also their job satisfaction, making the employee less likely to seek a new job. This, in turn, means you save on recruiting and training costs, plus you benefit from the higher level of quality that a more dedicated, experienced employee brings to the work she does.

Minding Your Users

The practice of creating engaging, efficient user experiences is called **user-centered design**. The concept of user-centered design is very simple: Every step of the way, take the user into account as you develop your product. The implications of this simple concept, however, are surprisingly complex.

Everything the user experiences should be the result of a conscious decision on your part. Realistically, you might have to make a compromise here and there because of the time or expense involved in creating a better solution. But a user-centered design process ensures that those compromises don't happen by accident. By thinking about the user experience, breaking it down into its component elements, and looking at it from several perspectives, you can ensure that you know all the ramifications of your decisions.

The biggest reason user experience should matter to you is that it matters to your users. If you don't provide them with a positive experience, they won't use your site. And without users, all you've got is a dusty Web server somewhere, idly waiting to fulfill a request that will never come. For the users who do come, you must set out to provide them with an experience that is coherent, intuitive, and maybe even pleasurable—an experience in which everything works the way it should. No matter how the rest of their day has gone.

chapter 2

Meet the Elements

The user experience development process is all about ensuring that no aspect of the user's experience with your site happens without your conscious, explicit intent. This means taking into account every possibility of every action the user is likely to take and understanding the user's expectations at every step of the way through that process. It sounds like a big job, and in some ways it is. But by breaking the job of crafting user experience down into its component elements, we can better understand the problem as a whole.

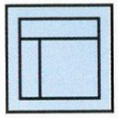
The Five Planes

Most people, at one time or another, have purchased a book over the Web. The experience is pretty much the same every time—you go to the site, you find the book you want (maybe by using a search engine or maybe by browsing a catalog), you give the site your credit card number and your address, and the site confirms that the book will be shipped to you.

That neat, tidy experience actually results from a whole set of decisions—some small, some large—about how the site looks, how it behaves, and what it allows you to do. These decisions build upon each other, informing and influencing all aspects of the user experience. If we peel away the layers of that experience, we can begin to understand how those decisions are made.

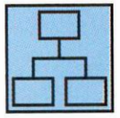
The Surface Plane

On the **surface** you see a series of Web pages, made up of images and text. Some of these images are things you can click on, performing some sort of function such as taking you to a shopping cart. Some of these images are just illustrations, such as a photograph of a book cover or the logo of the site itself.



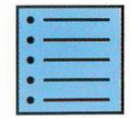
The Skeleton Plane

Beneath that surface is the **skeleton** of the site: the placement of buttons, tabs, photos, and blocks of text. The skeleton is designed to optimize the arrangement of these elements for maximum effect and efficiency—so that you remember the logo and can find that shopping cart button when you need it.



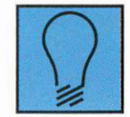
The Structure Plane

The skeleton is a concrete expression of the more abstract **structure** of the site. The skeleton might define the placement of the interface elements on our checkout page; the structure would define how users got to that page and where they could go when they were finished there. The skeleton might define the arrangement of navigational items allowing the users to browse categories of books; the structure would define what those categories actually were.



The Scope Plane

The structure defines the way in which the various features and functions of the site fit together. Just what those features and functions are constitutes the **scope** of the site. Some sites that sell books offer a feature that enables users to save previously used addresses so they can be used again. The question of whether that feature—or any feature—is included on a site is a question of scope.



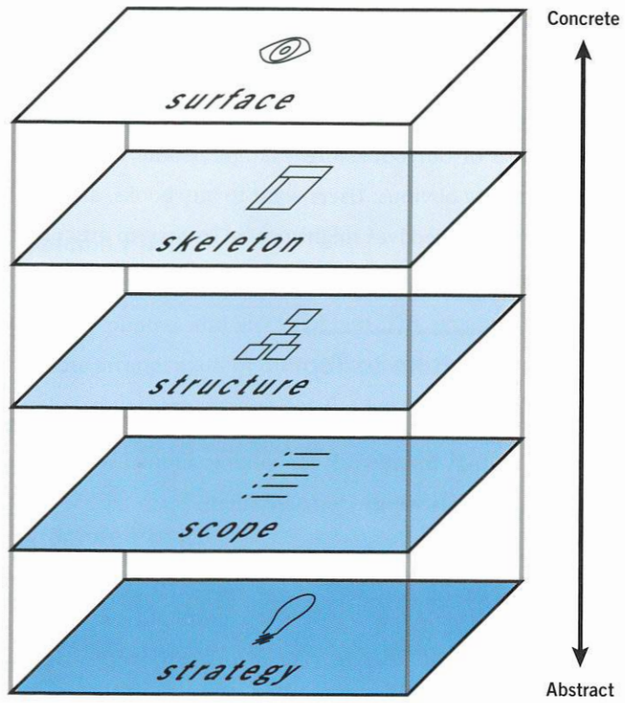
The Strategy Plane

The scope is fundamentally determined by the **strategy** of the site. This strategy incorporates not only what the people running the site want to get out of it but what the users want to get out of the site as well. In the case of our bookstore example, some of the strategic objectives are pretty obvious: Users want to buy books, and we want to sell them. Other objectives might not be so easy to articulate.

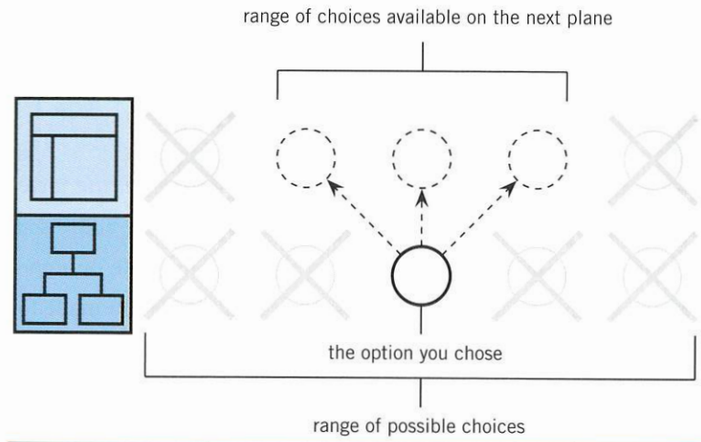
Building from Bottom to Top

These five planes—strategy, scope, structure, skeleton, and surface—provide a conceptual framework for talking about user experience problems and the tools we use to solve them.

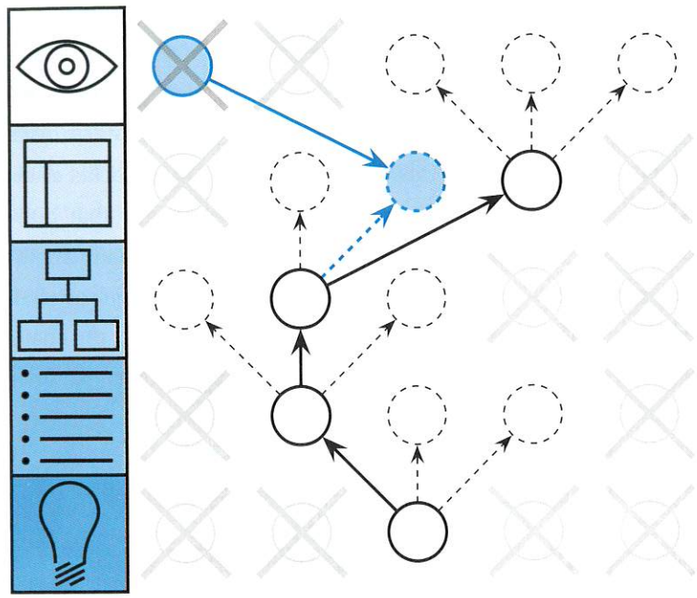
On each plane, the issues we must deal with become a little less abstract and a little more concrete. On the lowest plane, we are not concerned with the final shape of the site at all—we only care about how the site will fit into our strategy (while meeting the needs of our users). On the highest plane, we are only concerned with the most concrete details of the appearance of the site. Plane by plane, the decisions we have to make become a little more specific and involve finer levels of detail.



Each plane is dependent on the planes below it. So, the surface depends on the skeleton, which depends on the structure, which depends on the scope, which depends on the strategy. When the choices we make don't align with those above and below, projects often derail, deadlines are missed, and costs begin to skyrocket as the development team tries to piece together components that don't naturally fit. Even worse, when the site finally does launch, the users will hate it. This dependence means that decisions on the strategy plane will have a sort of "ripple effect" all the way up the chain. Conversely, the choices available to us on each plane are constrained by the decisions we make about issues on the planes below it.



The choices you make on each plane affect the choices available to you on the next plane above it.

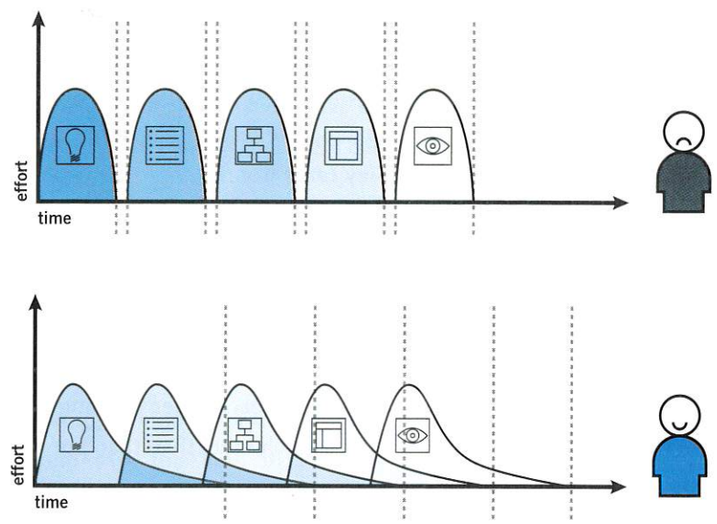


This ripple effect means that choosing an "out of bounds" option on an upper plane will require rethinking decisions on lower planes.

That does not mean, however, that every decision about the lower plane must be made before the upper plane can be addressed. Dependencies run in both directions, with decisions made on upper planes sometimes forcing a reevaluation (or an evaluation made for the first time!) of decisions on lower planes. At each level, we make decisions according to what the competition is doing, industry best practices, and plain old common sense. These decisions can have a ripple effect in both directions.

If you consider your decisions on lower planes to be set in stone before you take on your decisions on higher planes, you will almost certainly be throwing your project schedule at the very least—and possibly the success of your final product—into jeopardy.

Instead, you should plan your project so that work on any plane cannot *finish* before work on lower planes has finished. The important consideration here is not to build the roof of the house before we know the shape of its foundation.



Requiring work on each plane to **finish** before work on the next can **start** leads to unsatisfactory results for you and your users.

A better approach is to have work on each plane **finish** before work on the next can **finish**.

A Basic Duality

Of course, there are more than just five elements of user experience, and as with any specialized field, this one has evolved a vocabulary all its own. To someone encountering the field for the first time, user experience can appear to be a complicated business. All these seemingly identical terms are thrown around: interaction design, information design, information architecture. What do they mean? Anything? Or are they just more meaningless industry buzzwords?

To further complicate matters, people will use the same terms in different ways. One person might use "information design" to refer to what another knows as "information architecture." And what's the difference between "interface design" and "interaction design?" Is there one?

Fortunately, the field of user experience seems to be moving out of this Babel-like state. Consistency is gradually creeping into our discussions of these issues. To understand the terms themselves, however, we should look at where they came from.

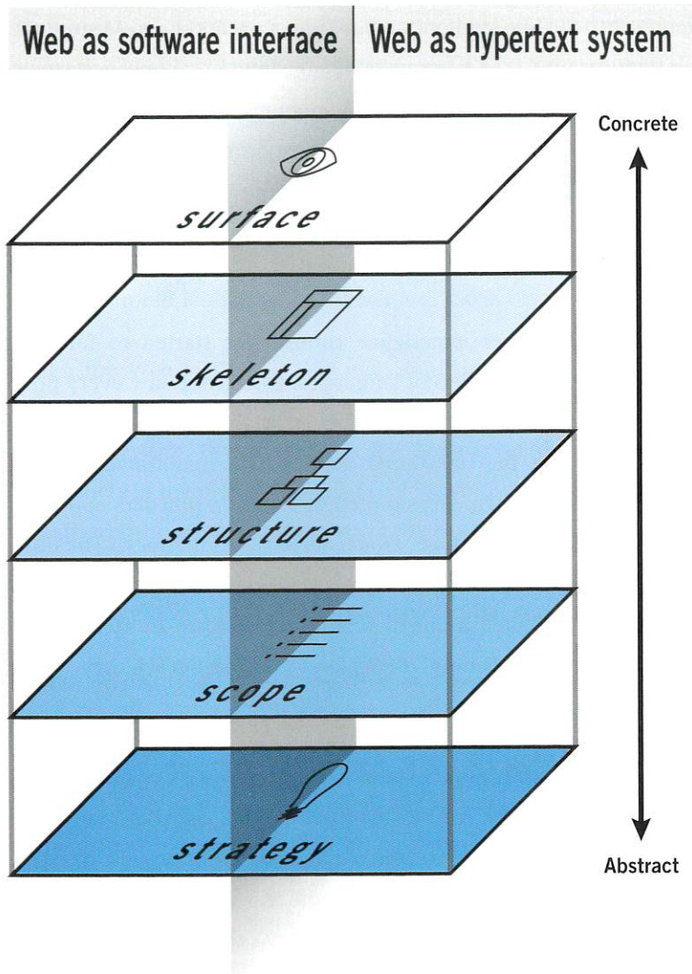
When the Web started, it was just about hypertext. People could create documents, and they could link them to other documents. Tim Berners-Lee, the inventor of the Web, created it as a way for researchers in the high-energy physics community, who were spread out all over the world, to share and refer to each other's findings. He knew the Web had the potential to be much more than that, but few others really understood how great its potential was.

People originally seized on the Web as a new publishing medium, but as technology advanced and new features were added to Web browsers and Web servers alike, the Web took on new capabilities. After the Web began to catch on in the larger Internet community, it developed a more complex and robust feature set that would enable Web sites not only to distribute information but to collect and manipulate it as well. With this, the Web became more interactive, responding to the input of users in ways that were very much like traditional desktop applications.

With the advent of commercial interests on the Web, this application functionality found a wide range of uses, such as electronic commerce, community forums, and online banking, among others. Meanwhile, the Web continued to flourish as a publishing medium, with countless newspaper and magazine sites augmenting the wave of Web-only "e-zines" being published. Technology continued to advance on both fronts as all kinds of sites made the transition from static collections of information that changed infrequently to dynamic, database-driven sites that were constantly evolving.

When the Web user experience community started to form, its members spoke two different languages. One group saw every problem as an application design problem, and applied problem-solving approaches from the traditional desktop and mainframe software worlds. (These, in turn, were rooted in common practices applied to creating all kinds of products, from cars to running shoes.) The other group saw the Web in terms of information distribution and retrieval, and applied problem-solving approaches from the traditional worlds of publishing, media, and information science.

This became quite a stumbling block. Very little progress could be made when the community could not even agree on basic terminology. The waters were further muddied by the fact that many Web sites could not be neatly categorized as either applications or hypertext information spaces—a huge number seemed to be a sort of hybrid, incorporating qualities from each world.



To address this basic duality in the nature of the Web, let's split our five planes down the middle. On the left, we'll put those elements specific to using the Web as a **software interface**. On the right, we'll put the elements specific to **hypertext information spaces**.

On the software side, we are mainly concerned with **tasks**—the steps involved in a process and how people think about completing them. Here, we consider the site as a tool or set of tools that the user employs to accomplish one or more tasks.

On the hypertext side, our concern is **information**—what information the site offers and what it means to our users. Hypertext is about creating an information space that users can move through.

The Elements of User Experience

Now we can map that whole confusing array of terms into the model. By breaking each plane down into its component elements, we'll be able to take a closer look at how all the pieces fit together to create the whole user experience.

The Strategy Plane

The same strategic concerns come into play for both software products and information spaces. **User needs** are the goals for the site that come from outside our organization—specifically from the people who will use our site. We must understand what our audience wants from us and how that fits in with other goals it has.

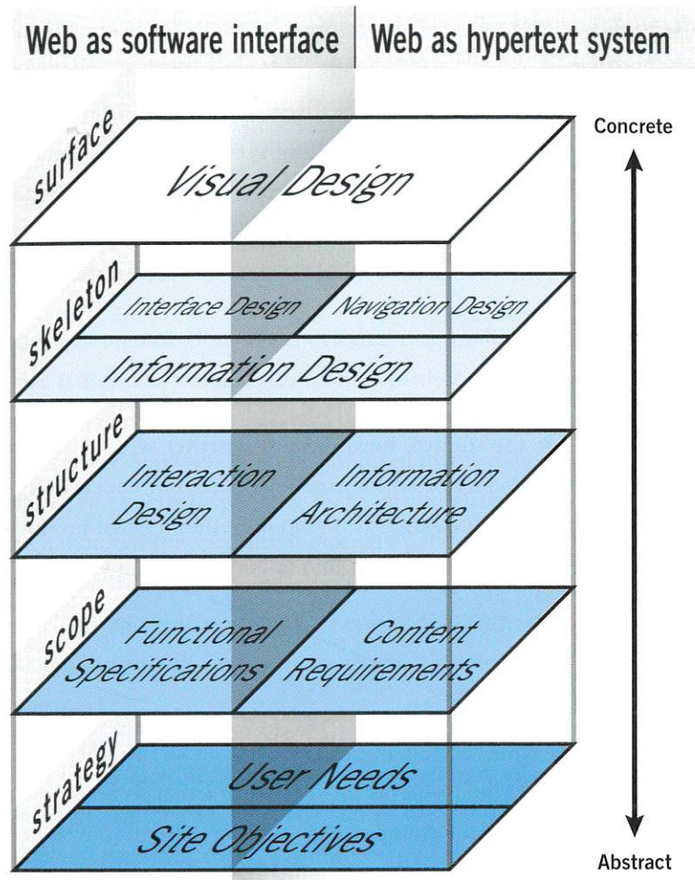
Balanced against user needs are our own objectives for the site. These **site objectives** can be business goals ("Make $1 million in sales over the Web this year") or other kinds of goals ("Inform voters about the candidates in the next election"). In Chapter 3 we'll go into more detail about these elements.

The Scope Plane

On the software side, the strategy is translated into scope through the creation of **functional specifications**: a detailed description of the "feature set" of the product. On the information space side, scope takes the form of **content requirements**: a description of the various content elements that will be required. Chapter 4 will cover the scope elements.

The Structure Plane

The scope is given structure on the software side through **interaction design**, in which we define how the system behaves in response to the user. For information spaces, the structure is the **information architecture**: the arrangement of content elements within the information space. You'll find more details on these in Chapter 5.



The Skeleton Plane

The skeleton plane breaks down into three components. On both sides, we must address **information design**: the presentation of information in a way that facilitates understanding. For software products, the skeleton also includes **interface design**, or arranging interface elements to enable users to interact with the functionality of the system. The interface for an information space is its **navigation design**: the set of screen elements that allow the user to move through the information architecture. There's more about the skeleton plane in Chapter 6.

The Surface Plane

Finally, we have the surface. Regardless of whether we are dealing with a software product or an information space, our concern here is the same: the **visual design**, or the look of the finished product. It's trickier than it sounds; you can find out all about it in Chapter 7.

Using the Elements

Few sites fall exclusively on one side of this model or the other. Within each plane, the elements must work together to accomplish that plane's goals. For example, information design, navigation design, and interface design jointly define the skeleton of a site. The effects of decisions you make about one element from all other elements on the plane is very difficult. All the elements on every plane have a common function—in this example, defining the site's skeleton—even if they perform that function in different ways.

This model, divided up into neat boxes and planes, is a convenient way to think about user experience problems. In reality, however, the lines between these areas are not so clearly drawn. Frequently, it can be difficult to identify whether a particular user experience problem is best solved through attention to one element instead of another. Can a change to the visual design do the trick, or will the underlying navigation design have to be reworked? Some problems require attention in several areas at once, and some seem to straddle the borders identified in this model.

The way organizations often delegate responsibility for user experience issues only complicates matters further. In some organizations, you will encounter people with job titles like information architect or interface designer. Don't be confused by this. These people generally have expertise spanning many of the elements of user experience, not just the specialty indicated by their title. It's not necessary to have a member of your team who is a specialist in each of these areas; instead, you only have to ensure that someone is responsible for thinking about each of these issues.

A couple of additional factors go into shaping the final user experience that you won't find covered in detail here. The first of these is **content**. The old saying (well, old in Web years) is that "content is king" on the Web. This is absolutely true—the single most important thing most Web sites can offer to their users is content that those users will find valuable.

Users don't visit Web sites to experience the joy of navigation. The content that is available to you (or that you have resources to obtain and manage) will play a huge role in shaping your site. In the case of our bookstore site example, we might decide that we want the users to be able to see cover images of all the books we sell. If we can get them, will we have a way to catalog them, keep track of them, and keep them up to date? And what if we can't get photos of the book covers at all? These content questions are essential to the ultimate user experience of the site.

Second, **technology** can be just as important as content in creating a successful user experience. In many cases, the nature of the experience you can provide your users is largely determined by technology. In the early days of the Web, the tools to connect Web sites to databases were fairly primitive and limited. As the technology has advanced, however, databases have become more widely used to drive Web sites. This in turn has enabled more and more sophisticated user experience approaches, such as dynamic navigation systems that change in response to the way users move through the site. Technology is always changing, and the field of user experience always has to adapt to it. Nevertheless, the fundamental elements of user experience remain the same.

The rest of this book looks at the elements, plane by plane, in greater detail. We'll take a closer look at some of the tools and techniques commonly used to address each element. We'll see what the elements on each plane have in common, what makes each one different, and how they affect each other to create the total user experience.