

## Where Can I Get the Answers to My Questions?

Visitors may come to your site with a definite purpose, but get distracted because they do not understand how to navigate through your menus or links, how to fill out your forms, how to place an order, or how to track down a human being in support. They wonder, they suffer doubts, they make strange hypotheses, and they become superstitious. And think about **where** this confusion attacks your visitors.

Often, the visitors get stuck at site-critical moments, such as when the visitors are:

- Filling out a registration form or customer profile
- Wondering whether to provide private information because it might be passed along to spammers and boiler-room phone operations
- Trying to grasp and use the organization of the site through the menu system
- Struggling to find a topic using the search mechanism
- Landing on a page that is close, but no cigar, with no suggestions for related topics
- Trying to get an answer to a question about a product, process, or service, and finding that the content ignores that question, or treats it with the back of the hand
- Wondering whether or not they can return a product easily
- Struggling to pick the right shipping options
- Trying to complete an order and make a purchase
- Looking for a way to get in touch with a human being in support

If visitors fail in these tasks, your site fails. The visitors don't see the information you've written, don't act in the ways your organization hoped for, and most folks leave, never to come back. Better customer assistance might make their visits successful, encouraging another visit. But, to overcome a mediocre interface

*If you have an unhappy customer on the Internet, he doesn't tell his six friends, he tells his 6,000 friends.*

—Jeff Bezos, Amazon.com

and confusing design, you have to write this material, organize it, and advertise it so it speaks to visitors in terms of their goals.

## Assistance gets buried

Web sites often bury their customer assistance, providing almost no explanations on the spot and hiding the detailed information behind confusing or ambiguously named links. For instance, in a survey of the 100 most popular Web sites in mid-2000, we found sites using a confusing array of similar terms for different aspects of customer assistance or for the same content. 75% use the term *Help*, 35% use the term *Customer Service*, 30% use the term *FAQ*, 20% talk of *Customer Support*, and fully 25% use additional terms. (Yes, that adds up to more than 100% because one site may use multiple terms).

Guests see your site as part of the Web, so, following Web conventions, they tend to expect that a *Help* button will actually help, and a *FAQ* link will actually bring up answers. But the links to *Customer Service* or *Customer Support* may just bring up e-mail forms. Who knows?

Whatever the term, only 35% of the sites we surveyed put a helpful link at the top of every page, and most of those also put the same link at the bottom of every page. Another 30% of the sites included a link in their side menus. The other sites put links on only some pages, slipped links into running text, or added them mainly to interactive forms. The inconsistent, hit-or-miss approach indicates that almost half the sites regarded customer assistance as something people only need occasionally.

Chaos and indifference is what many consumers experience when using the average site's customer assistance. And many sites treat the writing of customer assistance materials as a kind of janitorial task, something best left to new hires, part-timers, and junior programmers. Little wonder the results drive users away.

## The genres of customer assistance

Customer assistance generally takes several forms (shown here with the percentage of the top-100 sites that offer each type):

- Answers to frequently asked questions (FAQs) (90%)

- A bland privacy statement (90%)
- Responses to customer e-mail (80%)
- A separate section devoted to help, organized around tasks, concepts, or departments (65%)
- Phone numbers for customer support (45%)
- Labels embedded within the page on which the visitor is working (40%)
- Instructions during an activity (20%)
- Some kind of customer discussion board (20%)
- Chat with a live customer support person (10%)

Customer assistance shows up in many different forms and locations. Unlike a manual or a help system, Web customer assistance is decentralized, often diffuse, sometimes conflicting with itself or providing multiple overlapping chunks—more like gossip than a publication. This casual approach, in which assistance grows here and there, watered by various departments, means visitors see plenty of innovation, but little coherence.

And guess what? Almost all customer assistance is verbal. Fewer than a quarter of the sites we looked at use any diagrams, screenshots, or animations to show people what they need to know. Most sites seem to think that assistance must equal text. Crazy, but think of the responsibility that puts on the words you write.

See: Price (2000).

## Embedded Assistance—Labels, Tips, and Clues

*Design the interface as if the product will have no documentation.*

—Andrea Ames,  
*Architecture and Design of Online  
Information for Information-Rich  
User Interfaces and User Assistance*

If you're in the middle of ordering, and wonder how to fill out a slot in the form, or ask yourself what the choices mean, you may have to leave the order, go to the top of a FAQ menu, make a choice, read the material, realize it is not what you want, go back to the menu, choose another item, and, if it is relevant, memorize it, and then return, back, back, back to the form, to apply what you learned, if you can still remember it.

If you have a follow-up question, well, you just have to go through the same routine again.

In testing, we see people go through these loops two, three, four, even five times, just trying to understand one form. Little wonder that sites report half to three quarters of their shopping carts are abandoned before checkout is complete. GO TO is bad practice in programming, and GO TO is terrible for a visitor who just wants a simple answer to a question.

### Put the assistance where people need it

Embedding assistance in the interface works best. Just as software should explain itself, your site's interface should offer advice, rather than requiring people to leave the page, go wandering around, find something relevant, and write it on a yellow sticky (or memorize it) and then return and try again. You may need to provide a big pile of information on other pages, in the form of FAQs and Help, but start out by giving people a word to the wise right when they need it.

When asked how useful they considered conventional help systems attached to software, almost three quarters of novice users call it "not helpful." (Two thirds of experts say the same thing). The whole idea of a separate help facility supporting users is antiquated, an echo of paper manuals. Yes, you may need to provide full documentation in a separate place, but you should let guests avoid that big pile, by offering them on-the-spot tips.

## Make information part of the interface

Anywhere you want people to act, or where they might need to act, put advice, explanations, or instructions. Build assistance into the interface.

Your aim should be to help keep people on task, to make them successful, and to reduce the need to go elsewhere for information.

Of course, politically, this approach means you must join the design team, and make sure that the page layout has room for these little verbal flourishes. You are no longer creating a whole page about a subject, or even a few paragraphs. You are adding a phrase here, a sentence there, never more than a dozen words, total. You are not “writing about” the site—you are annotating the interface.

## Label those fields

How should I enter the date, so your system accepts it? What do you mean, *province*? Do I have to fill in this second address line?

Silly questions. Stupid users, right?

No—stupid interface.

You can smarten up the interface by having it explain exactly how to enter the date. And I’m not thinking of the old-fashioned MM/DD/YYYY code from mainframe applications. You have room to say you want the month, day, and year, in that order, and you can give an example, so people don’t accidentally confuse the software by entering March when the programmers expected “3” or, worse, “03.”

Please enter month, day, year, like this: 3/25/2002

Notice that we are also trying to make the interface polite. As your mother told you, well-brought-up writers say “Please” and “Thank you,” even in these labels.

If a field is required, say so. I know the marketing team worries that saying “Required” over and over again sounds authoritarian, and it is. But you need to make sure that users don’t inadvertently miss a field and then press Submit, only to be bounced back to the form with an error message like “Illegal input. Retry.” (And, in some cases, all the data has been wiped out of the form, so the user must type it all in again).

*Impatient people always arrive too late.*

—Jean Dutourd,  
*Le fond et la forme*

*I think that in order to write really well and convincingly, one must be somewhat poisoned by emotion.*

*Dislike, displeasure, resentment, fault-finding, imagination, passionate remonstrance, a sense of injustice—they all make fine fuel.*

—Edna Ferber

## Explain why you want the information

If you suspect people will wonder why you are asking, give a parenthetical explanation. If you are asking for a visitor’s mother’s maiden name, say why: “So we can make sure it’s you calling, if you want to check your account or change your password over the phone.”

Yes, a lot of words. But here more words equal more reassurance.

Generally, any request you have to explain belongs on a secure server, and you should be constantly—over and over—stressing that the information will be protected. Make a big deal of your secure server. At least once on every form page, down by the Submit button, state your privacy policy in a sentence, to allay fears and encourage that final click.

## Put embarrassing information where they need it

Have you ever put a product in your shopping cart, then gone to check out, entered your credit card number, and gone to a confirmation page only to discover that the shipping charges are outrageous? That happens a lot, because sites feel guilty about the charges, and fear that if you know the shipping costs in advance, you will refuse to buy.

Best practice: put the shipping options—and their costs—on every product page. The costs vary by weight, delivery time, and delivery service. That’s understandable. But part of the buying decision involves figuring out the total cost, so visitors need to see these options and the costs before they can confidently go ahead with the purchase.

Test: go to any e-commerce site and see how well they hide their policy on returning products. (Most sites seem to figure that they will avoid returns if they refuse to talk about them or limit their mention to a few cryptic sentences, implying that only Martians need to consider the issue).

Hiding key information is a form of lying. Don’t do it. Remember: people have seen elsewhere on the Web, sites that expose all this information at first glance. People know you can do it. So redesign your pages to give people the facts that your team feels may be embarrassing.

*Apple Guide and similar applications permit users to perform actual tasks as they interact with the help system. The guide provides step-by-step procedures that the user can perform immediately, perform with assistance, or allow the system to perform.*

—JoAnn Hackos and Dawn Stephens, “Instructional Information,” *Standards for Online Communication*

We’re not just talking shipping rates here. Your team knows what facts are most embarrassing, because the reviewers have complained about those aspects of the product, customers keep calling in with questions about them, and your competitors take pleasure in pointing to those weaknesses. Expose yourself then. Overcome the shame and include those facts along with the more positive ones.

How come? On the Web people will find this stuff out anyway, sooner or later. If you tip your hand, right off, they feel they can trust you. If you hide the facts, or reveal them only when you have to (like at the last moment during the order process), you make people mad.

## **Give examples, even on search forms**

If your advanced search offers Boolean choices, such as AND, OR, or NOT, make those into dropdown choices, so someone can build a query without guessing about the punctuation and sequence. But, in addition to the dropdown choices, give examples, right on the search page. Complex filters are great on large sites, but you need to tell people a little story with each choice.

Start by saying what someone wanted. “If you wanted to find a book by an author whose last name is Price, and you know you don’t want books by Willard Price...” You are posing the scenario. When people know what the purpose is, they understand the next part of the example much better.

Show the actual syntax. Continue your story with Part Two: the actual action taken. “You would type Price NOT Willard.”

Finally, Part Three of your example describes the results, so visitors can see what effect that action had. “You would get a list of all books by authors whose last name is Price other than Willard Price.”

Three parts, then: 1) scenario 2) action 3) results.

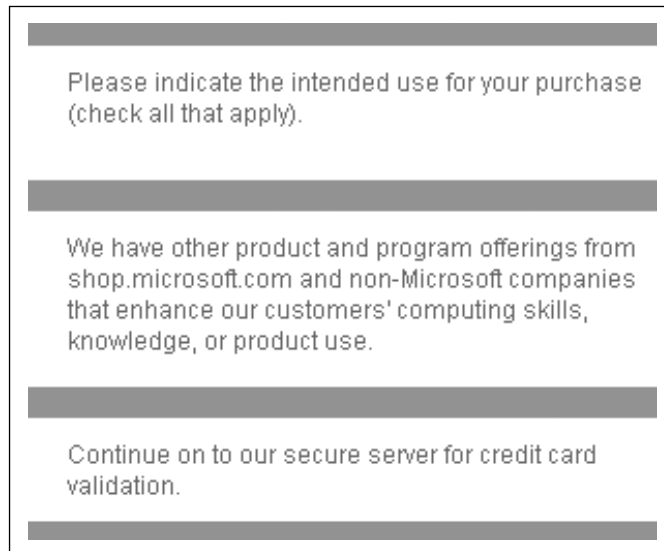
This tiny narrative helps people see how their own purpose might gibe with the imaginary character, and, if it does, they then discover what action they should take, and, just in case they are still in doubt, they can confirm that the results are what they would expect.

Use examples wherever you know that customers find the

process metaphysical—so abstract that only mathematicians and logicians think it makes sense. Times when you should consider adding examples, right on the page: when users are configuring a system, filling out a user profile, thinking about whether to subscribe, and choosing the most relevant marketing pitch.

**See:** Ames (2000), Boggan, Farkas and Welinske (1996), Duffy, Palmer, and Mehlenbacher (1992), Horton (1990), Price (2000), Price and Korman (1993).

## Case Study: Embedded Assistance at Shop.Microsoft.com



Please indicate the intended use for your purchase (check all that apply).

We have other product and program offerings from shop.microsoft.com and non-Microsoft companies that enhance our customers' computing skills, knowledge, or product use.

Continue on to our secure server for credit card validation.

The Microsoft shopping site holds your hand pretty tightly, with clear labels for sections (“Billing Info,” “Shipping Info”), clues about your progress (“Checkout (Step 1 of 2)”), and advice next to groups of fields, such as “Select a shipping option and enter a shipping address if different from your billing address.”

You can see that the writers have chosen a pretty clipped style here, opting for the short form of information, ampersands instead of *ands*, and one sentence to explain as many as a dozen fields. The tone is impersonal, but efficient. No joking around, but no hectoring, either.

Unable to explain shipping options in a phrase, the writers offer a link into the middle of a list of Common Questions. Unfortunately, I was deposited at the wrong spot, a few questions below the target. When I scrolled up, I went past the right answer,

and then slowly inched down to the info I wanted. Once I got there, the text went beyond the usual, explaining that shipping times are from the moment the box leaves the warehouse, but it might take Microsoft a day or two to get the box ready. “Expected processing time is two business days.” A lot of customers think that “2nd day” means that their package will arrive within two days of the order, and when nothing shows up, they start calling customer support in outrage. Here the writers—twice—emphasize that shipping times start after the package is ready to go. Good job! Of course, I’d like a personal and active style more like, “We usually get your order ready for shipping within two business days. (We don’t work on weekends.)” Now that kind of confession would be good for business, I think.

When I clicked FINALIZE my Order, without having filled in my address, I caught hell. First a giant “Attention” came up in red with the self-pitying complaint from the verification routine: “The following errors were encountered.” I had 16 errors, all in red, and all in the self-centered prose of system-centric messages, such as “Billing E-mail Address is a required field and was blank or only contained invalid characters.” Bad user! Showing me a list like this makes me think I have to memorize each of my mistakes, go back to the earlier page, and atone, line by line. If I forget, I think I will get another lecture. Actually, when I scroll down to the last message, I discover the top of the form again, where, helpfully, the labels of all fields that need data showed up in red. I could have used a little labeling next to or within the list of errors, telling me to scroll down and try again. (But maybe the team never imagined anyone would be such an idiot as to make 16 mistakes on one form, driving the form out of sight).

The tone of these messages is a little like the barking of an accusatory grammar teacher telling me that I have made a series of “errors,” I have not done something that was “required,” and I may have used “invalid characters.” The case against me is pretty damning.

So, despite working hard to offer assistance to customers, the site team sometimes focuses on its own concerns, its requirements—and my sins. They rarely tell me how to fix the problems.

Of 16 error messages, only one tells what to do to fix the mistake (the others tell me what they needed or expected and what data was missing, leaving it up to me to figure out the right way to behave). Net result: despite a good effort at assistance, the team has not completely overcome a tendency to be self-centered and arrogant.