▶ Browse by Topic and Author

The Need for Speed

by Jakob Nielsen on March 1, 1997

Topics: Application Design   Human Computer Interaction

Every Web usability study I have conducted since 1994 has shown the same thing: users beg us to speed up page downloads. In the beginning, my reaction was along the lines of "let's just give them better design and they will be *happy* to wait for it". I have since become a reformed sinner since even my skull is not thick enough to withstand consistent user pleas year after year.

Research on a wide variety of hypertext systems has shown that users need response times of less than one second when moving from one page to another if they are to navigate freely through an information space. Traditional human factors research into response times also shows the need for response times faster than a second. For example, studies done at IBM in the 1970s and 1980s found that mainframe users were more productive when the time between hitting a function key and getting the requested screen was less than a second.
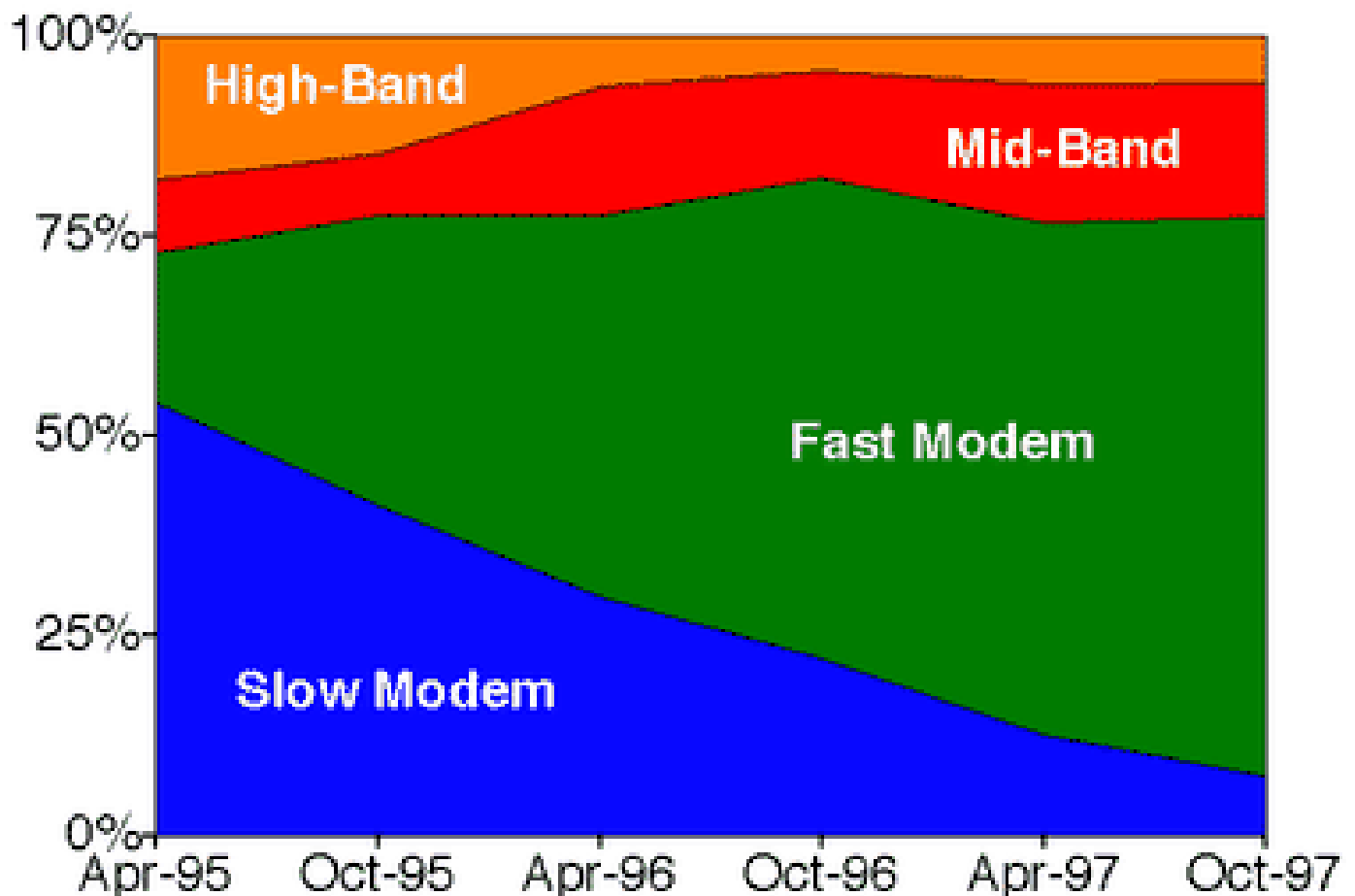
Unfortunately we are not getting subsecond response times on the Web any time soon, so we know that users are going to be hurt by slow downloads. Currently, the minimum goal for response times should therefore be to get pages to users in no more than ten seconds, since that's the limit of people's ability to keep their attention focused while waiting.

The figure shows the distribution of the speeds with which users have connected to the Internet according to the last six Georgia Tech Web user surveys. Two things are clear from the figure: users are replacing slow modems (14.4) with fast modems (28.8 or 33.6), but the proportion of users who connect at modem-speeds stays about the same. Even worse, the proportion of users connecting at **high bandwidths (T-1 or better) is going down**, even though the Web requires at least T-1 speed to work well. Mid-band (ISDN, leased lines, etc.) is increasing somewhat, but the truth is that mid-band speeds are

insufficient for decent Web response times.

The problem is that the user's experienced response time is a determined by the weakest link in the chain from server to browser:

- the **throughput of the server**: ought not to be a problem since the cost of hardware is the smallest part of developing a website, but popular sites often get surprised by rapidly increasing traffic and do not upgrade their machines fast enough.
- the **server's connection** to the Internet: many sites try to save on their connection and put off upgrading from, say, a T-1 to a T-3 even when their current connection is saturated
- the **Internet itself**: even though the net keeps getting upgraded, it still has bottlenecks, especially for cross-continent connections and for use at peak hours
- the **user's connection** to the Internet: as discussed above, connection speeds are very low for the majority of users and will stay that way for several years
- the rendering **speed of the user's browser** and computer: rarely a big problem, though complex tables can take significant time to lay out on low-end machines

Each of these many steps introduces its own delay in getting a webpage from the server to the user. Unfortunately, the delays are **cumulative**, meaning that we are not going to get snappy response times simply by improving a single link in the chain: when you upgrade from a modem to an ISDN line, you typically get twice as good Web performance and not the five times improvement implied by the bitrates.

Considering these fundamental facts in both human factors and computer networking, there is only one conclusion: webpages have to be designed with speed in mind. In fact, **speed must be the overriding design criterion**. To keep page sizes small, graphics should be kept to a minimum and multimedia effects should only be used when they truly add to the user's understanding of the information.

Conservative use of graphics does not imply boring pages. Much can be done by colored table cells and creative (but restrained) use of different fonts. In particular, style sheets can be used to improve page design without incurring a download penalty. If you are using style sheets (and everybody should be doing so by the end of 1997), I do recommend making them linked rather than embedded: a linked style sheet only needs to be downloaded once (assuming that you have a consistent style for your site), whereas embedded styles add to the size of every single page.

The most important issue in response time is when the user gets to see a screenful of useful information. It matters less if it takes longer to load the full page and all its illustrations if the user can start acting on *some* information fast. Guidelines for fast initial loading are:

- the top of the page should be meaningful even when no images have been downloaded (i.e., more text, less images)
- use ALT attributes for images so that the user can understand what they are about before they are rendered
- the browser must draw the top of the page fast: it can only do so if it has all the information it will need for layout, so include WIDTH and HEIGHT attributes on all images and table columns
- complex tables often take a long time to render, so cut down on the complexity of your tables by splitting the information into several tables. In particular, the top table should be simple and fast to render.

A practical tip: links to a directory should include the final slash in the URL when embedded in webpages. The slash should be left out when writing the URL for human consumption. For example, the link to the list of Alertbox columns should be written as `http://www.useit.com/alertbox` in a (printed) magazine article but coded as `http://www.useit.com/alertbox/` for the hypertext anchor in the online version of the article. The reason to include the slash in online links is to avoid a redirect when the server would have told the browser that the link refers to a directory and not a file. Adding trailing slashes to HREFs when appropriate reduces latency by a small but noticable amount, so you may as well do it.

A final recommendation is to use a server that supports HTTP keep-alive: saving the overhead of establishing a new TCP/IP connection for every "hit" cuts latency dramatically. The experienced response time to load a page often drops to about half by using keep-alive.

More on Responsiveness

[2010 update to this article](#) (including eyetracking example of the devastating effect of slow response times).

Share : [Twitter](#) | [LinkedIn](#) | [Facebook](#) | [Email](#)

Learn More

[Subscribe to the weekly newsletter](#) to get notified about future articles.

Articles

[Wizards: Definition and Design Recommendations](#)

[What the Erroneous Hawaiian Missile Alert Can Teach Us About Error Prevention](#)

[Progress Indicators Make a Slow System Less Insufferable](#)

[Input Controls for Parameters: Balancing Exploration and Precision with Sliders, Knobs, and Matrices](#)

[Response Times: The 3 Important Limits](#)

All articles about: [Application Design](#) [Human Computer Interaction](#)

UX Conference Training Courses

[Application Design for Web and Desktop](#)

[Interaction Design: 3-Day Course](#)

[Emerging Patterns for Web Design](#)

[User Interface Principles Every Designer Must Know](#)

[UX Basic Training](#)

Research Reports

[Mobile Intranets and Enterprise Apps](#)

[Website Tools and Applications with Flash](#)

[Customization Features Done Correctly for the Right Reasons](#)

[Application Design Showcase: 2012](#)

[Application Design Showcase: 2008](#)

Online Seminars

[Initiating a Design System](#)
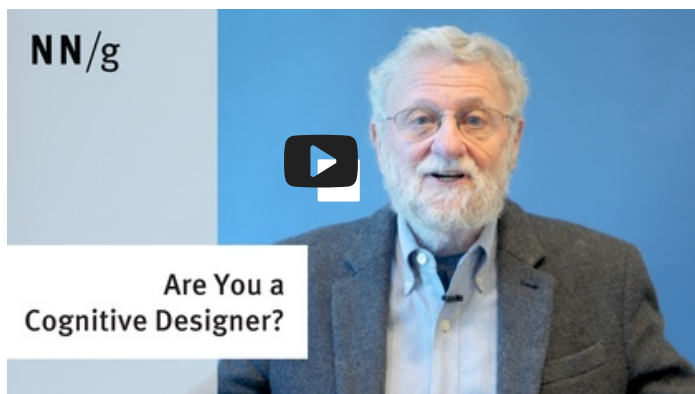
[UX for Big Data](#)

[Beyond Screens](#)

Videos

Will UX Ever Become Obsolete? (Jakob Nielsen)
2 minute video

The 3 B's Test For When to Follow Design Trends
3 minute video

Are You a Cognitive Designer? (Don Norman)
2 minute video

Popular Topics

Application Design

E-Commerce Design

Eyetracking

Human Computer Interaction

Information Architecture

Intranet Design

Interaction Design

Mobile and Tablet Design

Persuasive Design

Research Methods

User Testing

Web Usability

Web Writing & Content Strategy

Follow Us

Subscribe to our weekly UX newsletter, Jakob Nielsen's Alertbox

Visit Don Norman's website, www.jnd.org

Visit Bruce Tognazzini's website, www.asktog.com

Follow @nngroup

About

Contact information

About Us

Why NN/g?

Return Policy

Subscribe to our RSS feed

Cookie Declaration

Privacy Policy