

The Importance of Speed

Posted At : August 10, 2007 2:27 PM | Posted By : Steve

Related Categories: Usability

I went with my wife to Baltimore this week for a conference.

Among the vendor booths at the conference is a company that makes software targeted at the businesses attending. Knowing that I am a programmer, my wife's boss asked me if I would talk with one of them. I cautioned him that this was desktop software and I am a web programmer, so I would be of limited use. He understood, but was curious what I would think.

This sounded like fun, so I went there after we finished our lunch (at the Hard Rock of Baltimore which seemed to be populated mostly by those of us from "Flyover Country").

Upon talking to the people at the booth, it quickly became clear that my questions were too technical for them, but they did their best. This software is sufficiently old to be what I would call DOS-based (green text on a black screen). They didn't care for this term - preferring text-based. They are also coming out with a new version that their pamphlet calls "windows-based", but they preferred to call it the GUI version.

I was mostly asked to find out about the GUI version, so I directed most of my questions about that. The girl I talked to indicated that the GUI version wouldn't be full ready for a few months, but that the text-based version was very, very stable.

Now (and here comes the interesting part), she also told me that the text-based version is much faster. I said something about it being faster for the computer, but not the user. She said "Oh. If you know the keyboard commands, you can actually use the text-based version very quickly." I nodded and told her that I was sure she was right.

I confess, however, that I wasn't really convinced. I am sure that what she said is correct, but I am not sure that it matters. My wife's office has occasional turnover. This means that they have to train new employees from time to time. Most of the employees they hire are familiar with traditional Windows programs, but are not familiar with a DOS interface. It takes them quite some time to become familiar with using the software.

I mentioned the comment about speed to my wife. She said "I don't have any problem with how fast it runs. I don't like that it often calculates incorrectly, forcing me to recalculate several entries by hand. Then I have to correct each entry individually." Effectively, the software often wastes hours of her time and the time of other employees at the office.

Here then, is the crucial piece of importance about speed. Often much time is spent on improving performance in milliseconds that should be spent improving performance in hours (or days). Which would make my wife happier - If the clock-speed of the program were cut in half or if the program were made to calculate correctly 100% of the time?

Imagine if the software ran 5 times slower but always got the right answer. She would still be ahead by several hours (or days).

After calculation problems, the companies biggest performance problem with the software is training new users and having them complete tasks in the software. If the software were more intuitive to the modern user, imagine how much time (in days!) they could save. If this meant that the clock-speed of the program had to be cut by nearly an order-of-magnitude, they would still be ahead.

As a ColdFusion programmer, I am often bothered by the notion of "speed". I am told that PHP and ASP.NET are faster. My typical response is "ColdFusion is fast enough." It is fast enough for the sites that I work on and the sites that most people work on. It is fast enough to run Adobe.com. Is it fast enough to run MySpace.com? Maybe it is and maybe it isn't. I don't care. I'm not trying to write the next MySpace.com.

When confronted with the question of speed, I ask people about their experience with project failure and success. I ask them how many projects they know that failed because the program wasn't fast

enough. I ask them how many they know that failed because the project was late or over budget. So far, no-one I have met could think of a project that failed because the program was too slow. Almost everyone has known of at least one project that failed because the program was over-budget or over-schedule.

Productivity Speed matters. Speed to market matters. These are things that are measure in days/weeks/months. Computer speed can matter as well, but these are things usually measure in seconds. If you saved enough time writing the software, you will have enough time to optimize the speed (assuming that you took the time to write the software well - and you should have).

So next time you are thinking about the speed of our program, remember to concentrate on things that will impact your users by days and hours before you worry about things that will impact them by seconds and milliseconds.