Before you develop a solution, make sure you really understand the problem

devblogs.microsoft.com/oldnewthing/20060323-00

March 23, 2006

Raymond Chen

A common obstacle when trying to help people solve their problems is that what people ask for and what they actually want are not always the same thing. For technical problems, you often get a question that makes you shake your head in disbelief, but upon closer questioning, you find that the person really doesn't want what they're asking for. What they really want is something else, but they've already "solved" half of the problem and only need your help with the other half—the half that doesn't make any sense. For example, the literal answer to "How do I write a regular expression that matches everything except XYZ" is often a horrible mess, but if you dig deeper, you'll find that they really don't need a regular expression that matches everything except XYZ; they just simplified their problem to "I know, I'll use regular expressions" and ended up creating a bigger problem. (The best solution is often a mix of regular expressions and simple program logic.)

This problem also exists in user interface design. Rick Schaut describes one case where a user asked for a feature, when what they really wanted was an entirely different feature. Understanding the customer's problem is the first step towards solving it.

Raymond Chen

Follow