

Is Backtracking so Bad? The Role of Learning in Software Development

Ben Kovitz
Vertel Corp.
USA
ben-kovitz@vertel.com

Abstract

One of the hopes of good requirements engineering is that it prevents backtracking later in the development process. We hope that by researching customer needs well, we can prevent costly re-work after coding.

This talk proposes a different view of backtracking. The key is to see that analogies to manufacturing, plausible and attractive as they are, mislead when applied to software development. Software is a mental product, created by learning. The economics of learning are completely different from the economics of manufacturing.

About the Speaker

Ben Kovitz is a humble computer programmer at the Vertel Corporation in San Diego, California. He has worked in software development since 1984, as tester, programmer, analyst, technical writer, and user-interface designer, on everything from children's educational software to financial software to a CORBA ORB for embedded systems. His résumé is a mess. Ben is the author of "Practical Software Requirements: A Manual of Content & Style".