Industrial Experience in Automating Software Re-engineering

Lawrence Z. Markosian, Panel Chair
Reasoning Systems, Palo Alto, CA

Panelists:
Thomas Driscoll and Ron Hunter-Duvar, Formal Logic, Inc, Fredericton, New Brunswick, Canada
John Hartman, University of Texas at Austin
Wojtek Kozaczynski, Andersen Consulting, Chicago
Jim Ning, Andersen Consulting, Chicago
Mark Seipler, Motorola, Inc. Schaumberg, IL

Eighty percent or more of programming resources are targeted toward maintaining and re-engineering existing software. On the other hand, most CASE tools have been aimed at assisting the development of new code.

This panel will discuss the special problems of software re-engineering and will examine technologies and approaches that specifically address these re-engineering problems. The panel will be seeking confirmation of such technologies and approaches by industrial experience.

Re-engineering is distinguished from forward engineering by the need to analyze and understand large amounts of existing code. This code may be written in old languages or dialects, may run on obsolete hardware, and may use flat files instead of a more appropriate and efficient relational database.

The panel will specifically address approaches for—

- analyzing and abstracting the functionality of old code
- converting software to work with new languages or dialects
- translating large systems
- defining appropriate roles for the user of highly automated tools
- identifying the automatable steps in a large-scale project to re-engineer assembly language code.

The panel will indicate on the basis of actual experience what aspects of the re-engineering process seem to be most amenable to automation and what productivity gains can be expected.