

Thiagi on

Rapid Prototyping

Excerpted from "Rapid Prototyping" by Ron Zemke, *TRAINING*, September 2002, pages 92-100.

Sivasailam "Thiagi" Thiagarajan agrees that rapid prototyping and other appropriated technology ideas have real processes and techniques behind them. He sees concepts like rapid prototyping, extreme programming and concurrent development as important tools for making instructional systems design and development a faster, more flexible and responsive process.

Thiagi believes that the interest around ideas like rapid prototyping is both a break with, and a refreshment of, the ISD paradigm. The "old" paradigm represented by the analyze, design, develop, implement and evaluate, or ADDIE model, he contends essentially mimics a 1950s automobile—or airplane or television or computer—manufacturing model. It is being replaced by something Thiagi sees as more organic. Something he calls continuous instructional design. While crediting these new approaches

with improving instructional design, Thiagi cautions against turning one's back on the "old ways" of ISD. "When I was learning this trade from people like Susan Markel, one of the field's pioneers, there was always a premium placed on lean programming and systematic try-outs. Start with the final test or performance and work back-

"I teach that design and development is a continuous process. We should constantly find ways to make a program better. In truth, a program is never finished."

ward, only adding instruction and information as the result of successive try-outs of the material dictated. That spirit of experimenting and trial-and-error must be incorporated in these new ideas and methods as well. It is really what prototyping should be all about," he says.

Thiagi is less conventional and very collegial in his approach to prototyping. "I teach that design and development is a continuous process," he says. "We should constantly find ways to make a program better. In

truth, a program is never finished."

In keeping with that philosophy he starts with the test approach and moves backward to design, essentially doing continuous prototyping.

"I do many things to make it work," Thiagi says. "I like to ask a group of learners who want to learn conflict management, 'How would you prove to someone that you've mastered conflict management?' From there I move back one step and ask, 'Where does this knowledge reside?'

Sometimes they say, 'In books, sometimes in people.' Take conflict resolution again. I bring together a library of books on the subject and send people to different corners with a book and say, 'Come back with one good idea.' We exchange books and do that again and again. We compile and classify those good ideas. And then we ask, 'How would you learn these good ideas? Write down those approaches and pick the best ones to try. What we have done is build the prototype as a group."