Rapid Application Development

Arnie Pittler
Director, Software Engineering Research Laboratory
Corporate Software Research and Development

RAD means different things
...depends on which part of the life-cycle you want to optimize

- Before the Competition Does It
- Before the Customer Asks for it:
  - Examples: Post-It Notes, Minivans, ...
What fraction of your work uses some RAD technique?
- Nearly all use some RAD technique
- None, use RAD techniques in all phases of development.

What types of applications most / least need RAD?
- All need some form of RAD. At current trends what used to take 100 months will need to be done in 1 month.

What RAD techniques do you use? Which provide the most payoff?
- Virtual Prototypes
- Behavioral Modeling and Simulation
- Domain Engineering
- Application Frameworks
- Trusted Assets, i.e. Software Components
- Domain Specific Languages
- Automatic Test Generation
Have you noticed a trade-off between schedule compression and costs?

- Cost of Slippage
  - cost of lost business on current product,
  - cost of failing to staff future products,
  - depreciation related costs,
  - ...
- Cost of Slippage vs. Cost to Compress Schedule
- Longer schedules usually mean higher not lower costs!

Is there a point of diminishing returns? Can the software releases come out too fast?

- Releases CAN come out too fast.
- Demand for additional features appears to be insatiable.
- RAD lowers market risk.
  - Wait as late as possible to start a project.
  - Enable development of more products.
What are the roadblocks to adopting RAD?

- Organization
- "Brand You" Problem
  Technology for fast cycle-time vs. Technology for career value.
- Re-allocation of staff dedicated to non-value activities.
- Lowering the cost of Reuse.
- Cost of software tools
  - Western Europe/North America vs. Emerging Markets.