Joyce G. Fitzpatrick  
(for Mike Sweeney)  
EDS  

USC Executive Workshop  
March 11, 1998  
joyce.fitzpatrick@eds.com  

What fraction of work using some RAD technique?  

- 33%  
  - EDS has 40,000 Information Analysts/Engineers in applications development.  
  - 27,000 are working with legacy applications and 13,000 develop new systems.  
  - RAD techniques are used on almost all new development and occasionally on legacy work.
All new development that aims for meeting a reasonable schedule and cost structure should consider RAD techniques.

- Applications with heavy user interaction are best satisfied through RAD techniques.
- Enhancement and problem fix work can consider applying some RAD techniques but this application is atypical.

**What RAD techniques do you use?**

- Prototyping
- Facilitated sessions with User Involvement
- Case tools
- Data and process modeling
- Reuse
- Object-oriented methods
- Visual programming environments
- Storyboarding
- Rapidly timed system builds
- Customization of Off-the-Shelf Packages
Yes

Prototypes can be created with VPEs or storyboards. The visual impact creates an ideal tool for communication between users and technicians.

1. Solutions vary based on differing needs of each project.
2. Prototyping and facilitated sessions with heavy user involvement play a primary part of many successful RAD approaches.
Typically when schedule is compressed significantly, costs per unit would increase due to increase in training, coordination, and communication among additional personnel.

Long-term costs of support need to be considered as well. Schedule compression that forces abandonment of standards or structured code would be detrimental in the longer term maintenance and enhancement of code.

Typically, schedules can be compressed a maximum of 70% if starting with a nominal schedule.

Will a 10% longer schedule result in as much as a 20% lower development cost?

- It can……depending on the relative size of the project, existing project schedule, and quality of the estimate.
- There are many variables that impact schedule and cost metrics. Each project would need to evaluate whether they have a significant size, reasonable schedule, and accurate estimate.
Yes, each project has its own cost/benefit analysis. There are certainly times when completing a project sooner will achieve certain operational or financial benefits which are greater than the increased cost. Risk continues to increase as the project schedule is compressed.

Significant roadblocks to adopting RAD?

- Limited user availability and commitment
- Limited user decision-making authority
- Lack of management sponsorship
- Constrained subject matter expertise on business/application processes
- Constrained project management expertise especially in negotiating and prioritizing user requirements, scope management, and change management
- Business processing requirements becoming too specialized
- Just-in-time training on new tool sets
What are the best ways to start getting the roadblocks removed?

- Top management commitment by user group and technical group
- Empowerment of individuals and self-managing teams
- Commitment upfront to following RAD processes by all affected parties

Tools needed to make RAD commonplace?

- Continued progression of visual programming environments and their ease of use
- Libraries of reusable code
- Continued progression of case tools
- Continued progression of user tools
- Interactive collaboration tools
- Continued progression of automated testing tools
- Continued progression of effective estimating tools
Yes, proof-of-concept prototyping is common to resolve the integration issues of integrating and testing many differing tool sets in the client/server environment.

Can software releases come out too fast?

- Yes
  - It can present difficulties in training and acclimating users.
  - It can require a great deal of resources in testing and distribution.
  - It can cause a need to maintain multiple versions of software.
Allowing business requirements to become too specialized can slow development time significantly.

Heavy end-user involvement can lead to a convergence of sometimes disparate requirements. Discussions involving end-user management can often lead to a more concise "must-have" versus "nice-to-have" requirements categorization.

The amount of reduction would be project dependent.

---

Do engineers like working in a RAD shop?

- User involvement causes developers to receive answers to their questions faster so they can remain focused on writing code.
- Provides more immediate feedback to engineers because they see prototype solutions earlier in the development cycle.
- Allows more opportunities to learn more tool sets.
- Non-value added tasks are removed which engineers appreciate.
For more information about EDS, visit our website:

www.eds.com

The information contained in this report reflects the findings of a preliminary research effort. Conclusions and recommendations contained in this report do not necessarily represent EDS' strategy or position relative to the report's subject matter.