

## Software Reuse: Silver Bullet?

Melody M. Moore

Software reuse has been touted as a panacea, the solution to the software crisis, and a crucial technique in improving software engineering productivity. In practice, however, software reuse techniques and technologies have been plagued with difficulties in categorizing, storing, and retrieving reusable components. In our continuing series spotlighting the TCSE committees, this month's TCSE column features an interview with the chair of the TCSE Software Reuse Committee, Bill Frakes of Virginia Polytechnic Institute and State University.

Q: Is there a new definition of reuse? What is the practice's current state of the practice?

A: Reuse has evolved beyond simply reusing code or objects. Currently, we are focusing reusing knowledge itself, not just life-cycle artifacts such as specifications, code, and test data.

Q: How do we reuse knowledge?

A: The key is in domain engineering and analysis. This involves determining common elements across a domain and capturing this information so that it can inform further domain development. Domain knowledge sources include code, specifications, documents, as well as information that only exists in the heads of domain experts. Knowledge reuse is challenging because domain information is often not written down.

Q: Once we have collected knowledge about a domain, what do we do with it?

A: Domain models can assist in

forming generic templates for software architectures. Standard notations have recently been extended to describe multiple systems in a nonspecific fashion. Ontologies, or domain vocabulary models, can be extracted from documents using faceted classification. This technique involves extracting words and phrases from documents and analyzing them to determine which facets or groups are important. These facets can form the basis for a domain's high-level requirements definition. *Feature tables* are created during a step called *feature analysis*, which can be performed at either a user or builder level. Users and builders choose features from each of the facet groups (similar to choosing from a restaurant menu) to specify a system.

Q: Is reuse the panacea that the software engineering community has hoped for?

A: Reuse can work very well, but not in all circumstances. Initially the community held a naïve view of how long it would take to analyze each domain. For example, a very successful case of reuse is the `lex` and `yacc` code for creating compilers, which produce a 40-to-1 payoff from specification to code. However, this is a very narrow and specific domain. The domain must be well defined and well understood; attempting to form a cohesive domain model for the entire telecom field, for example, is too big and complicated to be effective.

Q: Are there key factors in the success of reuse efforts?

A: Perhaps surprisingly, the key fac-

### For More Information

The reuse information Web site is at <http://frakes.cs.vt.edu/renews.html>. The International Conference on Software Reuse (ICSR) is scheduled for 15-19 April 2002 in Austin, Texas.

tor in making reuse work is support and commitment from upper management. Reuse practices pay off, but they require resources up front. Organizations that only receive support from technical personnel are typically not successful; the overhead costs of reuse need to be amortized across an entire project or organization.

Q: What's on the horizon for reuse technology?

A: The success of a systematic domain analysis approach to reuse has highlighted the need for domain engineering research. The development community has embraced object patterns, but this level is not high enough. We need to study the economics and metrics of reuse—when is it worth doing? Can we employ strategic economic models to determine whether reuse is cost-effective? Answering these questions will help make reuse more effective. ☞

**Melody M. Moore** is an assistant professor of computer information systems at Georgia State University. Her research focuses on reengineering and assistive technology for people with severe disabilities, specifically brain-computer interfaces. Contact her at the Computer Information Systems Dept., College of Business Administration, Georgia State Univ., Atlanta, GA 30303-4013; melody@gsu.edu; www.cis.gsu.edu/~mmoore.