Methods and Tools for Information Systems Development
Introduction to Minitrack

Alan R. Hevner
Department of Information Systems
University of South Florida
Tampa, FL 33620

Richard C. Linger
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Information systems development is an exciting and challenging field that is undergoing many significant changes. Customer requirements for high-quality systems delivered on-time, within budget, and with full functionality provide a constant impetus for new and better methods and tools for information systems development. At the same time, research and development activities struggle to keep up with new industrial practices such as internet/intranet strategies, formal development methods, object-oriented innovations, distributed client-server architectures, and other provocative techniques. We are also faced with serious challenges, such as the Year 2000 crisis and the increasing demand for complex, safety-critical systems.

Over the past four years, this minitrack has been a forum for scholarly papers on research and development in the field. State-of-the-art research ideas in many diverse areas and disciplines have been presented. For this year's minitrack, we solicited and received research papers in many areas:

- Software requirements definition and validation
- Domain-specific modeling and architectures
- Software reuse strategies and techniques
- CASE tool integration and life cycle support
- Cleanroom software engineering methods and tools
- Software process modeling and assessment
- Software metrics and measurement
- Empirical studies of tool and method effectiveness
- Formal methods for systems development

After a thorough process of review and evaluation, six papers will be presented in this year's Methods and Tools for Information Systems Development minitrack. These papers will be presented in two sessions.

The first session begins with a paper by Stephanie Watts and P.J. Guinan. They present a longitudinal study of 49 software development teams to analyze the interactions between team (conflict resolution) and technology (reuse) factors and the degree of complexity and ambiguity in the projects. The empirical data clearly support the hypotheses that effective application of conflict resolution and reuse are associated with higher customer satisfaction of the implemented system. In addition, the authors demonstrate that conflict resolution has a positive impact on highly complex systems and reuse has a positive impact on highly ambiguous systems. The second paper by Robert Winter details a methodology for the specification and implementation of concurrency control in information systems. An extended conceptual model is used to define the database abstractions, as well as the active database triggers. Once implemented in the database, the triggers are reused by every module accessing the database instead of having to be replicated in every application. The methodology has been integrated into a commercial CASE environment and used to develop a production information system for a large mechanical engineering company. A controversial paper by Robert Galliers and Jacky Swan completes the first session. They argue that the information systems development field has long been dominated by structured approaches that emphasize technology-based data processing. They call for a reassessment of our assumptions in information system development. They propose a contrasting perspective on information requirements analysis as being socially mediated and thereby requiring greater awareness of the socio-political aspects of IS development process, methods, and tools.

The second session brings together papers on formal development processes, software metrics, and hypermedia authoring. Rick Linger and Carmen Trammell report on work at the Software Engineering Institute to map Cleanroom software engineering onto the Capability Maturity Model (CMM). The paper demonstrates the clear advantages of performing process improvements via the formal development methods of Cleanroom. The second paper by Don Berndt and Alan Hevner presents a formal model for collecting, organizing, and analyzing change metrics during information systems development. Based on the concept of punctuated equilibrium, states of equilibrium in the IS life cycle are identified and system metrics are collected and compared with previous equilibriums. A COR model is developed to study system change as defined by system core, obsolescence, and recency. Two interesting case studies demonstrate the use of the COR model. The final paper by Koen Hendriks, H. Olivie, and Erik Duval discusses a new approach for the development of object-oriented hypertext information systems. New ideas for hypermedia modeling, design, and construction are presented for internet authoring. The authors describe an implementation of the authoring system with innovative object-oriented features.