

ACM SAC2002 Software Engineering: Theory and Applications (SETA) Track Description

Track Chairs:

David C. Rine
Professor of Software Engineering
Department of Computer Science
George Mason University

James Power
Department of Computer Science,
N.U.I. Maynooth,

Paul Gibson
Department of Computer Science,
N.U.I. Maynooth,

The **THEME** of the SETA Track is Improving Software Engineering Methods and Tools through:

- Software Reuse and Integration.
- Software Requirements and Process Improvements.
- Software Tools and Applications.
- Best Software Engineering Practices and Standards.
- Improvements in Existing Software Validation and Verification Methods.
- Tying Improved Practices to Improved Selections of Tools with Methods.
- Improvements in Software and Systems Requirements.
- Quality of Software Processes and Products.
- Innovations in Software Quality Assurance.

Software Reuse and Integration Papers.

- 'Reusable subsystems: domain-based approach' by Hisham Haddad
- 'Using an object oriented model for resolving representational differences between heterogeneous systems' by Paul Young
- 'A method for assessing the reusability of object oriented code using a validated set of automated measurements' by Fatma Dandashi
- 'An approach to integrating semi-formal and formal notations in software specification' by Sergiu Dascalu and Peter Hitchcock

Software Requirements and Process Improvements Papers,

- 'Behavioral analysis patterns: towards a new representation of systems requirements based on actions and events' by Assem El-Ansary
- 'Proxy viewpoints model-based requirements discovery' by Seok-Won Lee
- 'Project Data Analysis for Software Process Improvement' by Derek Lawson and Gerry Coleman
- 'Strategies for personal process improvement' by Rory O'Connor and Gerry Coleman

Software Tools and Applications Papers.

- 'Evaluating collaborative software in supporting the mental model concept of organizational learning with Bayesian networks' by Mahmoud Elish
- 'Using XML to implement abstraction for model checking' by Maria del Mar Gallardo and Jesus Martinez and Pedro Merino and Estefania Rosales
- 'Using UML for secure systems development' by Jan Jurjens
- 'A UML statechart semantics with message-passing' by Jan Jurjens

Value of the Research Reported in these Papers.

The value of the research reported to the software engineering community is in helping managers and organizations make better decisions concerning which new methods and tools will result in the greatest improvement in quality and productivity. The research also helps demonstrate the need to adequately support those decisions.

Software Development Capabilities Essential to Survival.

Increasing the quality and productivity of large software development efforts is essential to many organization's competitive position, reducing their development time to market, and remaining viable in today's rapidly changing, information-based economy. Many organizations are facing the dual problems of needing to produce higher quality software within limited budgets and schedules. Most of these organizations recognize that their current methods and tools for software development are not quick enough and often produce systems whose quality is inadequate. This research provides a framework for making better decisions concerning which changes in software development technologies (methods and tools) are appropriate.

Focus on Achieving Software Development Goals.

Software development managers are constantly tasked to do things better, cheaper, and faster. Their traditional response to demands for these three measures of success is to answer, "Pick two out of the three". Over time, organizations develop cultural norms in which quality, schedule, or cost considerations predominate. The purpose in proposing the use of the interdependencies model is to help project managers better understand the factors affecting their decisions on methods or tools in order to have a high percentage of their projects in the "D" (for delightful) range.

How Much Improvement Will New Technologies Produce?

As a result of the need to produce better systems more quickly, many organizations are adopting, or considering adopting, new development methods and tools. These organizations can benefit from the proposed model since it can predict the level of improvement that can be achieved through changes in methods and tools. Organizations using the decision model will be able to invest their limited resources in the software development technologies that will provide the greatest return on their investment.

Methods and Tools Decisions - the Responsibility of the Project Manager.

While all of the factors depicted have some affect on the availability and use of methods and tools, the dissertation research will focus on the methods and tools direct impacts on productivity and quality. These factors (e.g. project requirements, corporate culture, staffing, etc.) will be treated as exogenous factors outside of the decision authority of managers who are determining which methods and tools to use on a project. The managers making the methods and tools decisions must factor these staffing, culture, and reuse influences in; but, will not typically be able to change them in the timeframe of the project.

Deciding how a software development project is to be performed is essential to the success of most projects. Unfortunately, many projects spend too little time considering how to do the work and instead depend on the heroic efforts of the staff to make the project a success despite the lack of defined methods and the lack of adequate tools. By using the decision model proposed, organizations can make more coherent decisions and then be able to track the accuracy of their assumptions.

Managers Need to Make Informed Decisions on New Methods and Tools.

Managers can use the decision model to make well-informed decisions about the appropriate level of resources to invest in new methods or tools for their projects. Despite the fact that in most projects, the project manager has little control or authority over many of the factors affecting the success of the project, the project manager typically can only directly choose:

- Which software development methods and tools to use
- The level of reuse to expect and support
- The training provided to the project staff
- The communication and coordination of the project staff.

Managers who understand and properly consider all the interdependencies within the environment in which they must operate (even those over which they have no control) are better able to achieve the maximum potential benefit of the new software development technologies.