ABSTRACT
We propose a workshop that will bring together researchers from the areas of component trust and certification, component technology, and software architecture. The goal of this workshop is to ensure that work in the areas of certification of software components and architectural analysis for prediction of system quality attributes will be mutually aware, if not mutually reinforcing. The output of the workshop will be a defined set of community model problems that reflects this intersection of interests.

Keywords
Certification, component, software architecture, analysis

1 INTRODUCTION
The critical role of software components in modern business systems is no longer a matter of analyst speculation but rather an accomplished fact. However, consumers of components are concerned about the trustworthiness of components developed by third-party providers. This concern is reflected in the recent attention to issues of component certification. Indeed, Underwriter Laboratories (UL) has already ventured into the arena of certifying software in programmable components, a fact that demonstrates the economic incentives in improving the level of trust in software components. Efforts to improve the level of trust in software components has also drawn the attention of researchers, as is evidenced by the work of the Trusted Components Initiative as well as the recent development of a variety of technical foundations for independent component certification.

Yet, despite these efforts, fundamental questions remain. What does it mean to trust a component. What technical and/or social processes are needed and appropriate for generating trust? Which properties of a component make it trustworthy? Still more fundamental: what ends are served by certifying (or developing trust) in these properties? Assume, for example, that we can devise a measure of reliability for software component; what would this tell us? It may reveal something essential about the component, but this knowledge would not necessarily translate to a measure of reliability for systems that use the component. Moreover, there are many other properties of interest besides reliability, including responsiveness, throughput, security, availability, modifiability, testability, and so forth. Without the ability to perform compositional reasoning over a variety of quality attributes, trustworthy components may be little more than a chimera.

The study of software architecture provides techniques for compositional analysis that can be applied to assemblies of components. The conceptual languages of software architecture and software components have much in common. Two key concepts of software component technology, component models and their implementation by component infrastructures, allow us to think of component technologies as a direct embodiment of architectural style, while component infrastructures provide a compose-time and run-time environment to enforce these style constraints.

Software architecture analyses are based, at least in part, on

1 Contact author.
the properties of a system’s architectural components as well as their interconnections. The utility of such predictions will be greatly enhanced if the properties of software components are certifiably conformant to the properties of the architectural components they implement. Conversely, the utility of component certification, or some other means of achieving trust in component properties, will be greatly enhanced if these certified properties are linked with architecture-based compositional reasoning.

2 WORKSHOP AIMS
We propose a two-day workshop that brings together researchers from the software architecture, software components, and trusted components communities. The goal of this workshop is to develop a shared understanding of how the work of these communities can lead to the twin objectives of certifiable component properties and predictable assembly of components. We have assembled a strong program committee that is representative of these three communities, and we anticipate a lively and provocative workshop.

The concrete objectives for the workshop are to:

- Define the characteristics of the problem space, for example what types of system attributes are amenable to compositional analysis (from informal heuristics to modular reasoning), which measurement techniques can validate predictions, and so forth.
- Using these characteristics, specify one or more community problems, possibly graduated from simple to “grand challenge,” that can provide a cynosure for community development.
- Relate the research activities of workshop participants to the community problems, and in this way identify gaps where work is needed, as well as fruitful areas for collaboration.

3 WORKSHOP ORGANIZATION
Participation will be by invitation only and will be limited to a maximum of 25 researchers. Invitations will be extended to key leaders in the three relevant research areas. In addition, a call for papers will be posted. All participants will be required to submit a 2-5-page position paper. Acceptance of position papers will be based on relevance to the objective of the workshop. Position papers will be posted at the workshop’s web site in advance of the workshop providing a forum for advanced discussion and group development of the workshop agenda.

Because the size of the workshop will be kept intentionally small, there will be no need for breakout sessions and the concomitant fragmentation of technical discussions. The format of the workshop will be a series of short (10-15 minute) position statements followed by moderated discussion.

4 WORKSHOP ORGANIZERS
Kurt Wallnau is a senior member of the technical staff at the Software Engineering Institute (SEI), Carnegie Mellon University. Mr. Wallnau has successfully organized three ICSE component-based software engineering workshops. He has written numerous articles and technical reports on the subject, and is currently leading the SEI’s Predictable Assembly from Certifiable Components project.

Judith Stafford is a senior member of the technical staff at the Software Engineering Institute, Carnegie Mellon University. Dr. Stafford has worked for several years in the area of compositional reasoning and its application to software architectures. She has written book chapters, papers, and technical reports related to software architecture and its use as a foundation for system analysis.

Heinz Schmidt is Professor for Software Engineering at Monash University where he directs the Centre for Distributed Systems and Software Engineering. He has worked on software specification, environments, and component technology for distributed and parallel systems for over twenty years at the German National Research Centre for Computer Science, the International Computer Science Institute of the UC Berkeley, the Australian CSIRO and Monash University. Dr Schmidt has written books and articles in this area and organized conferences and workshops on objects and software engineering.

Ivica Crnkovic is professor and head of the Computer Science Laboratory at Mälardalens University. He worked for many years at ABB, where he was responsible for development environments as a project leader and a manager of a development group. His main research interests are Software Configuration Management, Component-based Development, and in general Software Engineering. He is active in several conferences and workshops as a co-organizer or a PC member.

Other program committee members include:

Jan Bosch, University of Groningen, The Netherlands
Rob Deline, Microsoft Research, Washington
George Heineman, WPI, Worcester, Massachusetts
Wojtek Kozaczynski, Rational Software (tentative)
Otto Preiss, Asea Brown Boveri (ABB)/CRC, Zurich
Clemens Szyperski, Microsoft Research, Washington
Steve Vestal, Honeywell, Minneapolis, Minnesota
Jeffrey Voas, RST, Reston, Virginia
Wolfgang Weck, Oberon Microsystems, Zurich
Dave Wile, ISI, University of Southern California
Alex Wolf, University of Colorado at Boulder