The Impact of Distributed Object Technology on Reengineering
Panel Session Introduction

Scott R. Tilley
Software Engineering Institute
Carnegie Mellon University
stilley@sei.cmu.edu

1 Background

Reengineering is commonly accepted to mean any activity that improves one's understanding of a software-intensive system, or that improves the system itself. Traditionally, the approach taken to reengineering has been to understand the system's structure and to extract its essential functionality so that it can be transformed into a more evolvable system over the long term. But distributed object technology is dramatically changing the nature and economics of legacy system reengineering in two important ways.

The first impact is the new approach to reengineering that distributed object technology offers. Traditional reengineering is based on "deep" program understanding and reverse engineering. The cost/benefit ratio of this approach is staying the same in the face of new technologies such as CORBA, Java, and the Web. However, the benefits of "shallow" interface understanding and component wrapping using these distributed object technologies is rising rapidly relative to the replacement cost. As a result, the economic balance is changing from traditional transformation-based reengineering to wrapping-based reengineering. This may have significant impact on many organizations struggling with modernizing their systems in light of the so-called "Year 2000" problem.

The second impact is on the reengineering of systems that are built using distributed objects. Current research is primarily focused on using static analysis techniques to reengineer monolithic systems. This approach may not be successful when applied to distributed systems built from off-the-shelf components that are essentially black boxes. New analysis techniques may be required to reengineer such systems. For example, binary reverse engineering, interface behavior probing, and protocol analysis may be more useful in understanding the nature of such a system.

2 Panel format

This panel session brings together four prominent individuals who have been actively involved in the reengineering community for some time. The panelists are Hausi Müller, Kurt Wallnau, Harry Sneed, and Mike Olsem. Hausi has extensive experience in the use of reverse engineering tools and program understanding techniques. Kurt's work on component based systems makes use of distributed object technology. Harry has extensive real world experience in both traditional reengineering and object-based wrapping. Mike has investigated reengineering tools for several years, including those for the Year 2000 problem.

Each panelist was asked to prepare a position statement on the impact of distributed object technology on reengineering. A short (approximately 15-minute) presentation will be made by each, followed by an open discussion with the audience. The panelists were asked to consider and address the topical coverage of the session:

1. When is wrapping using distributed objects preferable to understanding using reverse engineering?
2. What is the main problem that you see for practitioners trying to use distributed object technology in reengineering?
3. Is the reengineering community ready for maintaining systems built using distributed objects? If not, what are some of the near-term issues that must be addressed?

The responses of the panelists to these three questions will be discussed during the conference session and are reflected in the written panel position papers that follow this introduction.