Wrapping Coarse-Grained Objects Using Standard Infrastructure Technology
Panel Position Statement

Hausi A. Müller, Kenny Wong, Margaret-Anne Storey
Department of Computer Science
University of Victoria
{hausi,kenw,mstorey}@csr.csc.uvic.ca

1 Introduction

With the expansion of high-bandwidth networks, the emergence of the World Wide Web, and the development of electronic commerce, businesses are under tremendous pressure to adapt and integrate their monolithic legacy systems into network-centric environments. There are high risks in trying to reengineer these legacy systems. One way to control and minimize these risks is to migrate to the new platform incrementally. The system is decomposed into a manageable set of subsystems, which are then migrated one at a time to the new platform.

There are two basic approaches to migrating a particular subsystem: reengineering and wrapping. In reengineering, the original subsystem is reverse engineered to produce an abstract specification, which is then forward engineered to create a new subsystem for the target platform. In wrapping, the original subsystem is essentially entombed and integrated into the new setting using infrastructure technologies.

Legacy systems that have evolved over many years are notoriously difficult to decompose. Wrapping is most effective if the legacy system can be decomposed into a few large chunks where the number of communication paths among subsystems is minimized. Transforming a monolithic legacy system into a set of fine-grained distributed objects is hard and risky. However, if the process of identifying objects and gluing them together can be automated, then the risks could be reduced. This approach might be feasible in some narrow, well-understood domains and thus worth investigating. Indeed, automatic transformation approaches have advanced considerably in recent years and have become key to correcting the Year 2000 problem.

2 Infrastructure Technologies

Software engineers can exploit software infrastructure technologies such as CORBA (Common Object Request Broker Architecture) and Java to build systems faster with better software reuse and interoperability. Wrapping existing applications or subsystems with standard or accepted technologies such as CORBA, ActiveX, or Java Beans has many advantages, including platform independence and opportunities for reuse. Once subsystems are entombed using standard wrappers, maintainers can safely ignore the details inside them, and can focus on the communication patterns among the subsystems. By using standard CORBA design patterns for these communications, future maintainers will find the system more familiar.

Nevertheless, it is not clear whether the maintainers of legacy systems are ready to embrace these rapidly emerging technologies. Developers and maintainers are under enormous time pressures and have limited time to keep up with the new technologies. Moreover, legacy systems evolve over decades, whereas network-centric computing technologies are evolving at a much more rapid pace. For example, the web browser and its associated infrastructure and user experience are evolving to become the next computer desktop.

3 Summary

In summary, wrapping is one of the most effective strategies for software migration. Using standard infrastructure technologies for coarse-grained objects is particularly promising. Finally, maintainers of legacy systems should exercise caution before jumping onto the latest technology bandwagon since the infrastructure technologies are still evolving rapidly while legacy systems evolve over decades.