Chapter IV

Support for Collaborative Component-Based Software Engineering

Cornelia Boldyreff, University of Lincoln, UK
David Nutter, University of Lincoln, UK
Stephen Rank, University of Lincoln, UK
Phyo Kyaw, University of Durham, UK
Janet Lavery, University of Durham, UK

Abstract

Collaborative system composition during design has been poorly supported by traditional CASE tools (which have usually concentrated on supporting individual projects) and almost exclusively focused on static composition. Little support for maintaining large distributed collections of heterogeneous software components across a number of projects has been developed. The CoDEEDS project addresses the collaborative determination, elaboration, and evolution of design spaces that describe both static and dynamic compositions of software components from sources such as component libraries, software service directories, and reuse repositories. The GENESIS project has focussed, in the development of OSCAR, on the creation and maintenance of large software artefact repositories. The most recent extensions are explicitly
addressing the provision of cross-project global views of large software collections and historical views of individual artefacts within a collection. The long-term benefits of such support can only be realised if OSCAR and CoDEEDS are widely adopted and steps to facilitate this are described.

Introduction

The systemic representation and organisation of software descriptions (e.g., specifications, designs, interfaces, and implementations) of large distributed applications using heterogeneous software components have been addressed by research in the Practitioner and AMES projects (Boldyreff et al., 1990; Boldyreff, 1992; Boldyreff, Burd, Hather, Mortimer, Munro, & Younger, 1995; Boldyreff, Burd, Hather, Munro, & Younger, 1996). The Practitioner project explicitly addressed the reuse of software concepts and developed a standard form to handle representations of software concepts from their specification to their associated implementations as components. The AMES project, while focused on maintenance support, organised the associated software components at various levels of abstract representations using hypertext and the Web. In both projects, it was assumed that the underlying collections of software components would support software reuse and the subsequent evolutions of systems composed from components. However, without appropriate representations and organisations, large collections of existing software are not amenable to the activities of software reuse and software maintenance; these activities are likely to be severely hindered by the difficulties of understanding the software applications and their associated components. In both of these projects, static analysis of source code and other development artefacts, where available, and subsequent application of reverse engineering techniques were successfully used to develop a more comprehensive understanding of the software applications under study (Zhang & Boldyreff, 1990; Fyson & Boldyreff, 1998). Later research addressed the maintenance of a Web-based component library in the context of component-based software product line development and maintenance (Kwon, Boldyreff, & Munro, 1997). The classic horizontal and vertical software decompositions proposed by Goguen (1986) have influenced all of this research. While they are adequate for static composition, they fail to address the dynamic aspects of composing large distributed software applications from components especially where these include software services that may be dynamically bound at run-time.

Recent research within the CoDEEDS project has made some progress toward the determination of design spaces to support both the static and dynamic system composition as well as the determination of the physical deployment and long-term operation of large distributed systems composed from heterogeneous components (Boldyreff, Kyaw, Nutter, & Rank, 2003). The current prototype implementation of collaborative support for the determination, elaboration, and evolution of design spaces, based on the CoDEEDS framework (Boldyreff & Kyaw, 2003), employs as its base another development of our recent research within the GENESIS project, the Open Source Component Artefact Repository, OSCAR (Boldyreff, Nutter, & Rank, 2002a; Boldyreff, Nutter, & Rank, 2002b; Boldyreff, Nutter, & Rank, 2002c; Nutter, Boldyreff, & Rank, 2003).