A Framework for Using Component Redundancy for self-Optimising and self-Healing Component Based Systems

Ada Diaconescu[1], John Murphy[*]
Performance Engineering Laboratory, Dublin City University
{diacones,murphyj}@eeng.dcu.ie

Abstract

The ever-increasing complexity of software systems makes it progressively more difficult to provide dependability guarantees for such systems, especially when they are deployed in unpredictably changing environments. The Component Based Software Development initiative addresses many of the complexity related difficulties, but consequently introduces new challenges. These are related to the lack of component intrinsic information that system integrators face at system integration time, as well as the lack of information on the component running-context that component providers face at component development time.

We propose an addition to existing component models, for enabling new capabilities such as adaptability, performance optimisation and tolerance to context-driven faults. The concept of 'component redundancy' is at the core of our approach, implying alternate utilisation of functionally equivalent component implementations, for meeting application-specific dependability goals.

A framework for implementing component redundancy in component-based applications is described and an example scenario showing the utility of our work is given.