MODEL DRIVEN EXCEPTION MANAGEMENT FRAMEWORK

Susan Carole Entwisle

Dissertation submitted in fulfilment of the Degree of
Doctor of Philosophy
Faculty of Information Technology
Monash University, Melbourne, Australia
October 2007

Supervisor: Dr. Sita Ramakrishnan
Associate Supervisor: Dr Ian Peake

Abstract

To support quality, a software system must be able to operate effectively when a failure occurs. Programming languages provide exception handling mechanisms to structure fault tolerant activities into software systems to manage failures. However, the use of exceptions at this low level of abstraction can be error-prone and complex thereby leading to new programming errors. This thesis proposes an approach to exception management based on the model-driven development paradigm. It proposes a domain engineering method and a generic, extensible, exception management framework that supports the model driven generation of exception handling features. We assess the use of the domain engineering method, and the features and architecture of the model driven exception management framework using a question-based evaluation framework and two case studies.

Thesis Contributions: To support the thesis that a generic, extensible exception management framework based on a model-driven architecture will support an automated approach to exception management and software quality, we have:
• identified the key requirements that a domain engineering method and a model driven framework need to support the exception domain.
• developed a tailorable domain engineering method to support the analysis, design and implementation of reusable modules for a wide range of exception domains.
• developed a reference implementation of the model driven exception management framework, which is generic enough to support different exception domains and provides support to automate the generation of exception handling capabilities.
• designed a generic evaluation framework for assessing a domain engineering method and domain specific modelling environment.
• evaluated the domain engineering method and model driven framework using our evaluation framework and through two case studies.