Title	
Author(s)	,
Citation	
Issue Date	2004-06
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/955
Rights	
Description	Supervisor: , ,

Extracting Concurrent Threads from Concurrent Objects

Mitsutaka Okazaki

School of Information Science

Japan Advanced Institute of Science and Technology

Abstract

This thesis proposes an approach for extracting concurrent execution sequences from concurrent objects. Recently, object-oriented development methods have played an important role in the domain of practical software engineering and have been adapted into developments of real-time systems. However, while object-oriented technologies have been maturing as a diagrammatical description language of systems, less technical tools and supports have been developed for analysing or verifying real-time properties of systems.

This thesis aims to clarify a logical foundation for analysing significant information for a real-time property: 'How many and what kind of threads are concurrently executed in a system'. A thread here means a execution sequence in a system; and since real systems usually have strict physical constraints on the number of CPUs, real-time performance of a system greatly depends on the number of concurrent threads are being executed in the system. Therefore, for analysing the real-time properties of a system it is important to obtain such information. However, existing object-oriented development methodologies do not give enough support to obtain threads from object-oriented models.

This thesis presents a solution of this problem; a transformation method from an object-oriented model into a thread-based model. In our approach, we clearly define two kinds of model. One is the *concurrent object model* which represents a typical object-oriented behaviour model based on concurrent state machines. The other is the *concurrent thread model* that is modelled as a set of explicit threads. It is easy to obtain information about the number of concurrent threads from the latter model. Then, we provide a method for transforming a concurrent object model into a concurrent thread model. This approach is formalised by using Basic Concurrent Regular Expressions (BCREs), which are an extension of regular expressions. There are two extending operators that represent concurrency and communication of concurrent state machines. As a logical base for the transformation method, we propose and use an axiom system for equivalent transformation of BCREs. It is then confirmed that this system is both sound and complete. We present our transformation procedure based on the equivalent transformation of BCREs. By using our axiom system, we also prove that our method is both sound and terminating.

Key Words: concurrent objects, concurrent threads, concurrent regular expressions, ojbect-oriented design, embedded systems