METRICS-BASED ANALYSIS OF THREAD BEHAVIOR USING ASPECT-ORIENTED PROGRAMMING APPROACH

SUMMARY

Performance concerns in a multi-threaded application naturally lead to issues related to thread behavior. To diagnose the cause of poor performance, one needs to observe thread behavior and gather information as the program executes in order to determine modifications in thread characteristics or resources that would result in better performance.

One of the most important problems in developing multi-threaded programs is that the behavior of threads on a real system cannot be predicted until they are run, and detailed information about the runtime behavior of threads is usually not available. If time and event information can be collected, metrics-based approaches can present useful information about the complex system being analyzed, and useful metrics that represent the behavior of threads can be calculated.

This thesis presents an approach based on accurate time and thread state information to analyze the behavior of multi- threaded applications using metrics and analyzes the efficiency of usage of aspect oriented programming technique for thread profiling. For this purpose, an aspect based thread profiler has been developed, namely Java Aspect Oriented Thread Profiler (JAOTP), which uses aspect oriented programming to profile threads. To deal with the problem of collecting accurate time information, a technique that uses the Java Native Interface (JNI) is applied. Through calls to native methods, information on timing and instantaneous cycle count of a logical processor is retrieved.

At the conclusion of the study, JAOTP is tested on a well-known producer-consumer problem and Apache Tomcat web server and the usefulness of the proposed metrics are discussed. In addition, JAOTP is compared with other well-known profilers and the overhead comparison is presented. According to the results obtained, a thread profiler that was implemented using aspect-oriented programming can be used on Java platform and metrics-based approaches are found to be useful in understanding thread behavior.