ANALYSIS of ASPECT ORIENTED PROGRAMMING BASED PROFILING of REAL TIME JAVA APPLICATIONS

SUMMARY

Performance is usually a major quality characteristic in software development. Although it can vary depending on the type of software, common performance indicators are memory allocation amount, method execution time and processor usage ratio. For better performing software, it is required to measure those indicators during development. This measuring process is called profiling and can easily reveal the hotspots and bottlenecks of a program. However, almost always profiling comes with the cost of an overhead to the program which can eventually reduce the accuracy of the profiling. Thus, an efficient profiler's overhead should be small enough to rely on.

Profiling the performance of software is a cross cutting concern, because a profiler application needs to cut across the whole program in order to produce performance data. This paper analyzes the efficiency of using aspect oriented programming (AOP) technique for profiling real time applications. AOP is a programming paradigm which complements object oriented programming (OOP). The main principle of AOP is separating the cross cutting concerns from software's business logic to increase readability, maintainability and modularity of code. For analyzing purposes, an AOP based java profiler (JAAOP) is introduced which uses aspectj for implementing AOP behavior. JAAOP can profile; method execution times, method execution counts, object initialization times, object counts and exceptions.

Profiler overhead results of JAAOP and a number of other known java profilers are presented for profiling Jake2, a java based 3D game engine. Jake2 is simply the java porting of well-known Quake2 game [1] and it is a real time Java application. The conclusion whether profiling a real time Java application, using AOP and aspectj is efficient, reusable and maintainable is presented in end of the study according to profiling results.