Performance is one of the important aspects of software applications. With growing complexity of different parts of computer systems, the search for problematic code areas is getting more and more difficult. To help in this task, tools called profilers are available. Ideally, a profiler would provide data from all layers of the system - from hardware through virtual machines of interpreted or partially compiled languages to the application itself. Unfortunately, profilers for languages that are not compiled to native code, such as Java, do not provide facilities to read data from lower levels. The goal of this work is to devise a profiler capable of profiling both Java and native code that would support modern features of native profilers such as triggering on configurable hardware events.