

Object mutability analysis for Java objects

Note: another student is working on this topic at the moment, using the Maxine Java VM as basis. This topic may still be taken using Hotspot as base platform.

While working with many Java applications we noticed that after startup. Long-living objects do not seem to mutate much, most objects seem to be immutable in a lot of applications. The task for this work is to experimentally verify this hypothesis on some applications.

Exact extent of this work will be determined with the supervisor depending on the type of course you want to get credit for. E.g. changes may be limited to the interpreter in Hotspot.

Possible Goals

- think about how to track mutability in a garbage collector efficiently. Possibilities may contain reference update tracking of (only long-living?) objects using e.g. card table and write barrier.
- statistically evaluate the observed behavior for different applications; slightly formalize the hypothesis and verify it.
- is there a correlation between object age and mutability?
- compare different approaches analyzing this hypothesis

Possible Tasks

- track modifications in the Java application, possibly using write barriers and card table, generating log output as necessary.
- evaluate and visualize the data; not only summary data is interesting, also over-time visualization.

Other information

- programming language C++ (using HotSpot); some additional code needed for evaluation and visualization in a programming language of your choice in coordination with the advisor.

Further information

- Thomas Schatzl, HF305, thomas.schatzl@jku.at