

VM Research at Ghent University

Lieven Eeckhout
Ghent University, Belgium



Research topics

- High-Level Language (HLL) VM
 - Java
- Memory management
- Performance analysis
- JIT compilation

Memory management in 64-bit JVMs

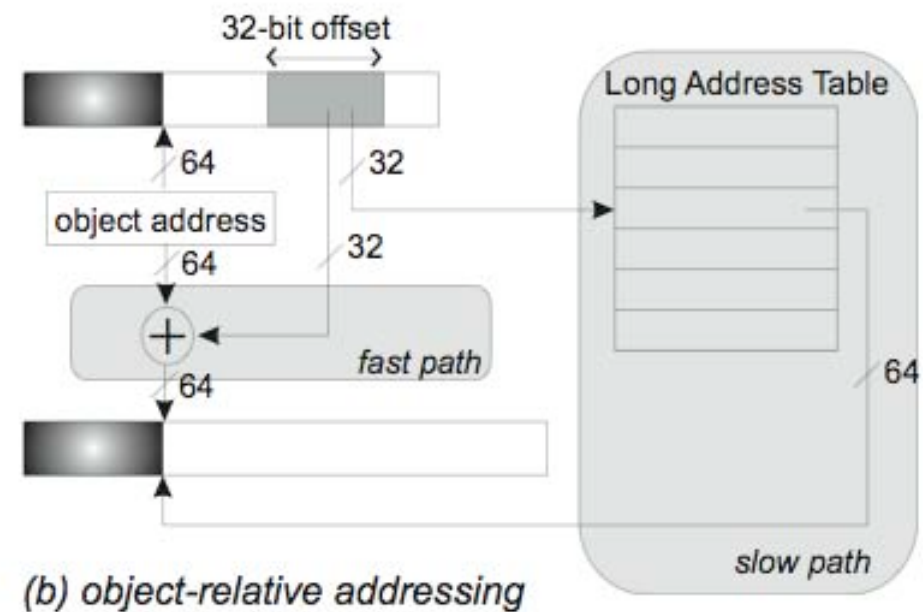
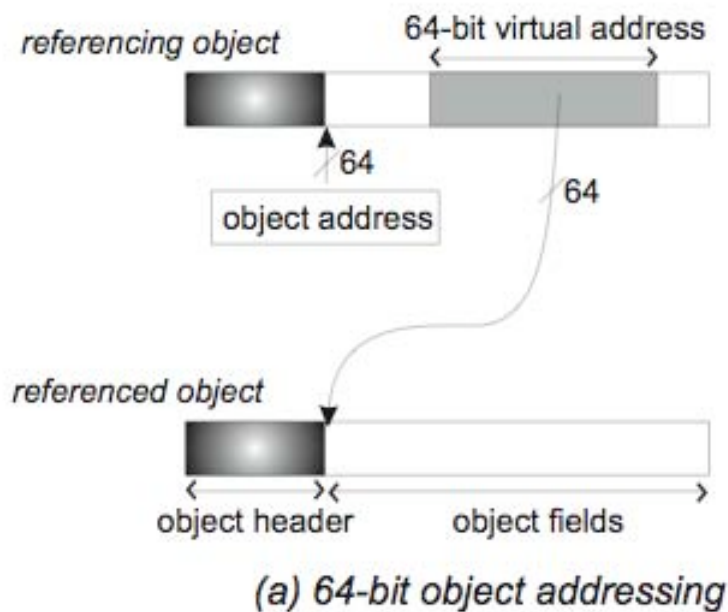
- 40% memory consumption increase from 32-bit JVM to 64-bit JVM
 - Pointer size increase
 - Increased object header size

[Venstermans et al., Software, Practice and Experience, '05]

Pointer size reduction: Object-Relative Addressing

- 10% reduction in memory consumption
- Virtually no overhead

[Venstermans et al., ECOOP'07]



Object header reduction: Typed Virtual Addressing

- 15% reduction in memory consumption
- Virtually no performance overhead

[Venstermans et al., CGO'06, ACM TACO'07]

- Key idea: allocate all objects of a given type in a contiguous memory segment
 - Object type is encoded in virtual address
 - No need for vtable/TIB pointers in object header

Performance analysis: Complex interaction

- Complex interaction between application, VM, input, heap size, GC strategy, etc.

[Eeckhout et al., OOPSLA'03]

- Phase-based profiling

[Georges et al., OOPSLA'04]

- Javana: Building customized vertical profiling tools

[Maebe et al., OOPSLA'06]

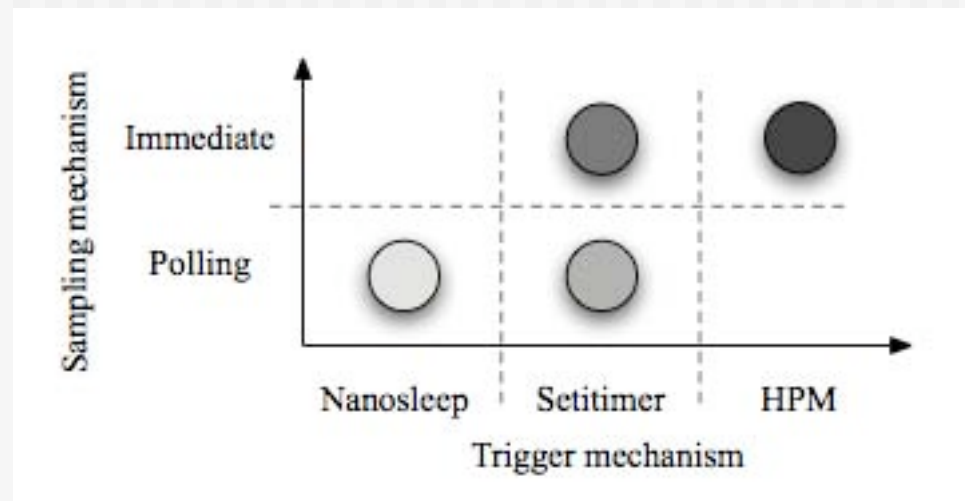
- Non-determinism due to timer-based JIT compilation, thread scheduling, etc. → statistics

[Georges et al., OOPSLA'07]

JIT compilation

- Timer-based JIT compilation using hardware performance monitors (HPMs)
- 5.7% performance speedup (up to 18%)

[Buytaert et al., OOPSLA'07]



Future research interests

- How to simulate VM workloads?
- Resource management / Performance isolation
 - Driven through analytical performance models
- Performance analysis
 - Rigorous methodologies: exp design + data analysis
- Workload characterization in system VMs
- Hardware support for system software
 - Per-thread cycle accounting architectures
- Co-designed VMs
 - For energy-efficiency