Realizing a Java Virtual Machine with SEAM

Final Presentation

Patrick Cernko

cernko@ps.uni-sb.de

September 16, 2004

Programming Systems Lab,

Saarland University

Motivation

- Java Virtual Machine
  - Wide spread, object-oriented VM
  - Published specification [Lindholm and Yellin, 1999]
- SEAM [Brunklaus and Kornstaedt, 2002]
  - Pluggable components & generic services to build VMs
- Existing JVM prototype
  - Naive prototype implementation using SEAM
- Questions to answer
  - Efficiency/Overhead to comparable JVMs

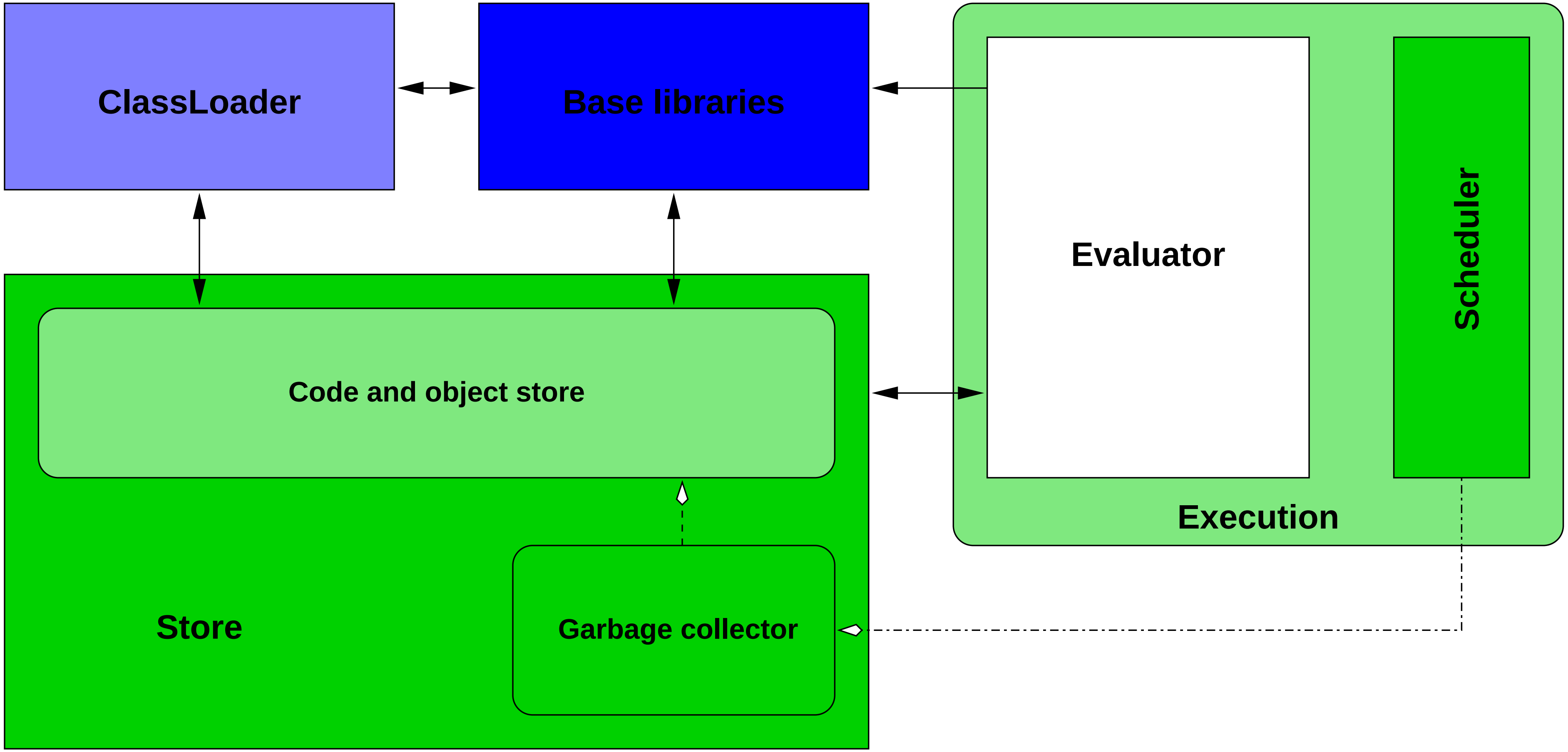
Aproach

- Refinement of existing prototype
  - Analysis of object model
  - Analysis of bytecode execution
- Reuse components from other JVMs
- Evaluation of refined JVM using standard benchmarks

SEAM

- Core
  - Abstract data store
    - Garbage collector
  - Generic concurrent execution model
    - Evaluator abstraction
- Language layer
  - Language data modeled on top of store
  - Language services modeled using evaluators

JVM-SEAM: Architecture



JVM-SEAM: Components

- Base-Libraries
  - Completely reused from Prototype
- ClassLoader
  - Mostly reused from Prototype
- Store
  - Code and object store: JVM data model on top of SEAM Store
- Bytecode execution
  - Evaluator: Implemented as interpreter using engine from JVM kaffe

Execution Engine

Bytecode Interpreter from Kaffe-VM

- Well structured and documented
- Mapping
  - Bytecode → Micro-Language → accessors

  ⇒ Own implementation of accessors
- Some accessors didn't match JVM-SEAM model

  ⇒ Small changes in kaffe's code-base required
  & workarounds in Accessor-Implementations

Evaluation

Benchmark	JVM-SEAM	Prototype	kaffe
fib	1.9^{-1}	1.8^{-1}	1.0
tak	2.3^{-1}	2.1^{-1}	1.0
nrev	3.6^{-1}	4.6^{-1}	1.0
quickarray	1.5^{-1}	1.4^{-1}	1.0
queens	2.3^{-1}	2.4^{-1}	1.0

Speed normalized to kaffeVM

Benchmark	JVM-SEAM	Prototype	HotSpot^{TM}
fib	4.7	4.9	1.0
tak	4.3	4.5	1.0
nrev	4.7	3.7	1.0
quickarray	3.6	4.0	1.0
queens	3.2	3.0	1.0

Speed normalized to HotSpot^{TM}

Discussion

- JVM-SEAM beats Prototype in *fib*, *tak* & *quickarray* (integer arithmetics)
- Prototype still faster in *nrev* & *queens* (method invocation) → but kaffe performs uneven slower!

  ⇒ kaffe lacks performance in method invocation & object creation
- JVM-SEAM beats kaffe with the same interpreter with twice up to tree times speed!

Summary

- Integration of kaffe-interpreter in JVM-SEAM
- Refinement of prototype partly successful
- JVM-SEAM beats kaffe

⇒ **SEAM *is* usable for JVM implementations**

Future Work

- Object model layout
  - improved concurrency locking [Onodera and Kawachiya, 1999]
  - Space and time efficiency [Bacon et al., 2002]
- JIT compiler
  - kaffe JITs [kaffe, 2004]
  - Jalapeño compilers [Arnold et al., 2000]

References

[Arnold et al., 2000] Arnold, M., Fink, S., Grove, D., Hind, M., and Sweeney, P. F. (2000). Adaptive optimization in the Jalapeño JVM. In *Proceedings of the Conference on Object-oriented Programming, Systems, Languages, and Applications (OOPSLA 2000)*, pages 47–65. ACM Press.

[Bacon et al., 2002] Bacon, D. F., Fink, S. J., and Grove, D. (2002). Space- and Time-Efficient Implementation of the Java Object Model. In Magnusson, B., editor, *Proceedings of the Sixteenth European Conference on Object-Oriented Programming (ECOOP 2002)*, volume 2374 of *Lecture Notes in Computer Science*, pages 111–132, Málaga, Spanien. Springer-Verlag.

[Brunklaus and Kornstaedt, 2002] Brunklaus, T. and Kornstaedt, L. (2002). A Virtual Machine for Multi-Language Execution. Technical report, Programming Systems Lab. http://www.ps.uni-sb.de/Papers/abstracts/multivm.pdf (August 2004).

[kaffe, 2004] kaffe (2004). kaffe - A free Java Virtual Machine. http://www.kaffe.org/ (August 2004).

[Lindholm and Yellin, 1999] Lindholm, T. and Yellin, F. (1999). *The Java^{TM} Virtual Machine Specification*. Adison Wesley, 2 edition.

[Onodera and Kawachiya, 1999] Onodera, T. and Kawachiya, K. (1999). A study of locking objects with bimodal fields. In *Proceedings of the 1999 ACM SIGPLAN, Conference on Object-Oriented Programming Systems, Languages & Applications (OOPSLA ’99)*, volume 34.10 of *ACM Sigplan notices*, pages 223–237. ACM Press.