Course Information

In this seminar we shall read technical papers on the design, definition, and implementation of the Java (TM) programming language. Our focus will be on those features that distinguish Java from previous mainstream languages. We will discuss how these features make Java attractive to programmers, and what challenges they pose to the language implementor.

Students will use the recently announced Open Runtime Platform from Intel to explore Java implementation alternatives hands-on.

Themes.

- Language design and definition
  - Java in the context of other object-oriented languages (historical and research perspective)
  - The design of the Java core language
  - The development of the Java environment and class library
  - Byte-code compilation
  - The structure of the Java Virtual Machine
- Language implementation
  - Realizations of the virtual machine for bytecode interpretation, from naïve to highly optimized
  - Optimizing compilation in general
  - Dynamic and Just-in-time compilation
  - Garbage collection
  - Thread management
  - JVM Security model
  - Case studies of state-of-the-art systems, including HotSpot and Jalapeño

In a future seminar, tentatively in the Fall 2001 term, we will revisit some of the implementation topics in greater depth, including control flow, type, alias, pointer, and shape analyses, security models, concurrency models, and mobile code.

Assignments: weekly readings and oral presentations, a large programming project.

Prerequisites in detail: Knowledge of the Java language and experience programming in Java. Understanding of the basic issues of language implementation, such as compilation and code optimization. Understanding of computer architecture. Understanding of operating systems structure. Permission of the instructor. (Please come and talk to me before the semester begins.)

UNM CS courses CS 257L - *Nonimperative Programming*, CS 351L - *Design of Large Programs*, CS 441 - *Modern Computer Architectur*, CS 548 - *Advanced Computer Architecture*, CS 454 - *Compiler Construction*, CS 552 - *Advanced Topics in Compiler Construction*, CS 481 - *Operating Systems Principles*, CS 550 - *Programming Languages and Systems*, and CS 453 - *Topics in Program Correctness* provide appropriate background, but you need not have taken all of them.

Seminar meetings: Wednesdays 5:30–8:15, Tapy Hall 218

Instructor: Darko Stefanovic, office FEC 345C, phone 2776561, email darko@cs.unm.edu — office hours Mondays 5:30-6:30, Tuesdays 11-12, Wednesdays 8:15-9, or by appointment

Reading material:

In addition to research papers (see below), we will use these two books:

James Gosling , Bill Joy , Guy Steele , Gilad Bracha: Java Language Specification, 2nd edition, 2000, Addison-Wesley, ISBN 0-201-31008-2.

Tim Lindholm, Frank Yellin: The Java Virtual Machine Specification, 2nd edition, 1999, Addison-Wesley, ISBN 0-201-43294-3.

These books will be available in the bookstore; on-line versions can be viewed at http://java.sun.com.

Grading: You must attend all class meetings, read all assigned papers, and participate in class discussion. Each attendee will present at least two papers (depending on the number of participants). Projects will be individual, as agreed between the students and the instructor. Larger projects may be undertaken by teams of students. Graduate students will be expected to undertake and complete particularly ambitious projects. You should come up with project ideas by the end of February, and have a written proposal by March 10; complete projects and reports will be due at the end of the finals week, on Friday, 11 May.

The grade will be determined as follows:

Class participation 30%

Oral presentations 30%

Project 40%

Detailed Lecture Plan

The list of papers we shall read, and who will present them. Two or three papers per meeting, usually.

17 January: Organizational meeting

24 January: Language Design

- Dionysius W. Vigil: Andrew Black: Object-Oriented Programming: Regaining the Excitement, ECOOP 1999.
- Hajime Inoue: Guy Steele: Growing a Language, OOPSLA 1998.
- Hajime Inoue: An overview of Java virtual machine structure

31 January: Class Files

- Jiaye Zhou: William Pugh, Compressing Java Class Files, PLDI 1999
- Dino Dai Zovi: Chandra Krintz, Brad Calder, and Urs Hölzle: Reducing Transfer Delay Using Java Class File Splitting and Prefetching, OOPSLA 1999

7 February: Compilation I

- : Ole Agesen and David Detlefs: Mixed-mode Bytecode Execution, SMLI TR-2000-87, June 2000
- Hajime Inoue: Allan Heydon and Marc Najork: Performance Limitations of the Java Core Libraries, JavaGrande 1999

14 February: Compilation II

- Niannian Modisette: Tia Newhall and Barton P. Miller: Performance Measurement of Dynamically Compiled Java Executions, Java Grande 1999
- : Manish Gupta, Jong-Deok Choi, and Michael Hind: Optimizing Java Programs in the Presence of Exceptions, ECOOP 2000
- Dino Dai Zovi: José Moreira, Samuel Midkiff, and Manish Gupta: From Flop to Megaflops: Java for Technical Computing, ACM TOPLAS, Vol. 22, No. 2, March 2000

21 February: Compilation III

- Dionysius W. Vigil: Ali-Reza Adl-Tabatabai, Michal Cierniak, Guei-Yuan Lueh, Vishesh M. Parikh, and James M. Stichnoth: Fast, Effective Code Generation in a Just-In-Time Java Compiler, PLDI 1998
- Trek S Palmer: Vasanth Bala, Evelyn Duesterwald, and Sanjeev Banerjia: Dynamo: A Transparent Runtime Optimization System, PLDI 2000

28 February: Compilation IV

- Irina Smirnova: Jeffrey Dean, Greg DeFouw, David Grove, Vassily Litvinov, and Craig Chambers: Vortex: An Optimizing Compiler For Object-Oriented Languages, OOPSLA 1996
- : David Detlefs and Ole Agesen: Inlining of Virtual Methods, ECOOP 1999

7 March: Dynamic Class Loading

- Dino Dai Zovi: Sheng Liang and Gilad Bracha: Dynamic Class Loading in the Java Virtual Machine, OOPSLA 1998.
- : Scott Malabarba, Raju Pandey, Jeff Gragg, Earl Barr, and J. Fritz Barnes: Runtime Support for Type-Safe Dynamic Java Classes, ECOOP 2000

14 March: no meeting - spring break

21 March: Security; Threads and the Memory Model I

- Victor Winter and/or Dino Dai Zovi: Christopher Colby, Peter Lee, George C. Necula, Fred Blau, Mark Plesko, and Kenneth Cline: A Certifying Compiler for Java, PLDI 2000
- More papers on security?
- Irina Smirnova: Andreas Krall and Mark Probst: Monitors and Exceptions: How to implement Java efficiently, JavaGrande 1998

28 March: Threads and the Memory Model II

- Trek S Palmer: William Pugh: Fixing the Java Memory Model, JavaGrande 1999
- : David Bacon, Robert Strom, and Ashis Tarafdar: Guava: A Dialect of Java without Data Races, OOPSLA 2000.

4 April: Synchronization

- Irina Smirnova: Jeff Bogda and Urs Hölzle: Removing Unnecessary Synchronization in Java, OOPSLA 1999
- : Jong-Dek Choi, Manish Gupta, Mauricio Serrano, Vugranam Sreedhar, and Sam Midkiff: Escape Analysis for Java, OOPSLA 1999
- : Erik Ruf: Effective Synchronization Removal for Java, PLDI 2000

11 April: Example Systems I

- Trek S Palmer: Dan Ingalls, Ted Kaehler, John Maloney, Scott Wallace, and Alan Kay: Back to the Future: The Story of Squeak - A Usable Smalltalk Written in Itself, OOPSLA 1997
- Darko Stefanovic: Mario Wolczko, Ole Agesen, and David Ungar: Towards a Universal Implementation Substrate for Object-Oriented Languages, OOPSLA99 VM Workshop
- : Antero Taivalsaari: Implementing a Java Virtual Machine in the Java Programming Language, SMLI TR-98-64, March 1998

18 April: Example Systems II

- Dino Dai Zovi: Antero Taivalsaari, Bill Bush and Diug Simon: The Spotless System: Implementing a Java System for the Palm Connected Organizer, SMLI TR-99-73, February 1999
- Trek S Palmer: HotSpot white paper (http://www.java.sun.com/products/hotspot/whitepaper.html)
- Hajime Inoue: Michal Cierniak, Guei-Yuan Lueh, and James M. Stichnoth: Practicing JUDO: Java under Dynamic Optimization, PLDI 2000

25 April: Example Systems III

- Niannian Modisette: Michael G. Burke, Jong-Deok Choi, Stephen Fink, David Grove, Michael Hind, Vivek Sarkar, Mauricio J. Serrano, V. C. Sreedhar, Harini Srinivasan and John Whaley: The Jalapeno Dynamic Optimizing Compiler for Java, JavaGrande 1999

- : Bowen Alpern, C. R. Attanasio, John J. Barton, Anthony Cocchi, Susan Flynn Hummel, Derek Lieber, Ton Ngo, Mark Mergen, Janice C. Shepherd, and Stephen Smith: Implementing Jalapeño in Java, OOPSLA 1999
- : Matthew Arnold, Stephen Fink, Michael Hind, David Grove, and Peter F. Sweeney: Adaptive Optimization in the Jalapeño JVM, OOPSLA 2000.

2 May: Language Extensions

- : Martin Odersky and Philip Wadler: Pizza into Java: Translating theory into practice, POPL 1997
- : Gilad Bracha, Martin Odersky, David Stoutamire, and Philip Wadler: Making the future safe for the past: Adding Genericity to the Java Programming Language, OOPSLA 1998
- : Gilad Bracha, Martin Odersky, David Stoutamire, and Philip Wadler: GJ:Extending the Java programming language with type parameters
- : Gilad Bracha, Martin Odersky, David Stoutamire, and Philip Wadler: GJ Specification

Note: we may not have time to read papers on memory management in Java and a few other topics that were in the draft syllabus. If there is enough interest, we shall work them in, or plan a broader seminar on memory management for next year. These are the papers in question:

- Sylvia Dieckmann and Urs Hölzle: A Study of the Allocation Behavior of the SPECjvm98 Java Benchmarks, ECOOP 1999
- Ran Shaham, Elliot K. Kolodner, and Mooly Sagiv, On the Effectiveness of GC in Java, ISMM 2000
- Wim De Pauw and Gary Sevitsky, Visualizing Reference Patterns for Solving Memory Leaks in Java, ECOOP 1999
- James M. Stichnoth, Guei-Yuan Lueh, and Michal Cierniak: Support for Garbage Collection at Every Instruction in a Java Compiler, PLDI 1999
- Tamar Domani, Elliot K. Kolodner, and Erez Petrank: A Generational On-the-Fly Garbage Collector for Java, PLDI 2000
- Richard L. Hudson, J. Eliot B. Moss, Sreenivas Subramoney, and Weldon Washburn: Cycles to Recycle: Garbage Collection on the IA-64, ISMM 2000
- Darko Stefanovic, Kathryn S. McKinley, and J. Eliot B. Moss: On Models for Object Lifetime Distributions, ISMM 2000
- Al Globus, Eric Langhirt, Miron Livny, Ravishankar Ramamurthy, Marvin Solomon, and Steve Traugott: JavaGenes and Condor: Cycle-scavenging Genetic Algorithms JavaGrande 2000

  `http://www.nas.nasa.gov/~globus/papers/JavaGrande2000/JavaGrandePaper.html`