[redacted]
[redacted]@[redacted]
[redacted]

OBJECTIVE

Seeking a challenging position in research and development of software tools for engineers (performance monitoring, debugging, build, testing) and related areas.

SUMMARY

Demonstrated ability to define and carry out research projects, provide technical leadership, leverage user input, and productize results in a short timeframe. At Sun, initiated and led several research projects, all of which resulted in publicly available products or product quality technologies/tools. At Google, led a small team of tool developers helping a large (over 100 people) group working on a multi-million LOC Java application. Successfully addressed a number of challenges in build and testing caused by huge code base and very high change frequency. Most recently, worked on the new version of high-performance internal RPC library.

Expertise in Java performance monitoring technologies and tools, Java build and testing tools, and the Java programming language and platform. Ph.D. in Computer Science (University of Glasgow, UK), with 9 years of subsequent industrial experience. Numerous published articles and issued patents.

SOME PUBLICLY AVAILABLE PROJECTS THAT I STARTED AND LED

1. JFluid/NetBeans Profiler (now JDK VisualVM) - a CPU, Memory and Thread Profiler. Features selective bytecode instrumentation based on code hotswapping and automatic call graph discovery, semi-statistical memory profiling, continuous low-overhead memory leak monitoring, dynamic attachment to running JVM, etc. See https://visualvm.dev.java.net

2. Javamake. A command line Dependency Checking tool for Java, that can be used with any Java compiler, and can run as a task in the popular Ant system. See http://kenai.com/projects/jmake

3. Testar. A tool that reduces testing time for large Java unit test suites. It runs on top of JUnit and automatically selects individual tests to execute based on changes made to the code since the previous tool invocation. See http://sourceforge.net/projects/google-testar

4. HotSwap feature in HotSpot JVM and the HotSwap GUI Tool. Available in the in the Java HotSpot Virtual Machine starting from JDK 1.4. Allows programmers to modify code on-the-fly. HotSwap Tool is available at http://www.experimentalstuff.com/Technologies/HotSwapTool

EXPERIENCE

May 2005 - present
Google Inc.
Senior Software Engineer
- Currently working on the next version of the high-performance internal RPC system. Writing highly concurrent Java code, redesigning user-level APIs, redesigning and rewriting the internals to fully utilize multithreading and separate code layers and generally clean up the code, gathering feedback and providing support to users, etc.
- Designed and developed the internal Submit Queue system, which allows developers to compile and test their changes on dedicated machines before submitting them. That made change submission process much less painful and mostly solved the problem of chronically broken code base for the biggest projects at Google. The system is now utilized by a few hundred projects within the company.

- Led a small Development Productivity team assisting a large Java-based project. Designed and developed tools and components to address issues caused by very large code base, large number of tests and high frequency of changes: advanced dependency checking for Java to speed up build, automatic test selection based on code coverage and user's changes to the code, automatic finding of code reviewers based on various heuristics, collecting code coverage in production to find dead or unused code, etc. Also spent much time helping others to integrate various internal systems, gathering user requirements and feedback, maintaining builds, writing documentation, etc.

May 2004 - May 2005
Sun Microsystems, Inc. - Java Tools, Menlo Park, CA
Staff Engineer, SW
- Technical lead and architect for the JFluid/NetBeans Profiler project:
  - Led the group of 3 engineers working on this tool. Performed planning, coordination, communication with users and other groups, documentation reviewing, etc.
  - Worked on the tool's "engine" - the functionality for data collection, transportation, and processing/aggregation.
  - Together with engineers from the JavaSoft Serviceability group, redesigned the hotswapping functionality in the HotSpot JVM, to improve its performance and stability.
  - Filed 5 patent applications.

June 2001 - May 2004
Sun Microsystems, Inc. - Sun Labs, Mountain View, CA
Staff Engineer, SW
- Principal investigator for the JFluid research project.
  - Designed and implemented advanced hotswapping functionality in the HotSpot JVM, resulting in the experimental "JFluid VM".
  - Starting from scratch, single-handedly designed and implemented the JFluid profiler. That included such advanced features as: dynamic call graph discovery and instrumentation, semi-statistical memory profiling, continuous monitoring for memory leaks based on GC patterns, dynamic attachment to running JVM, factoring out time overhead, online processing of profiling data, etc.
  - Gathered, analyzed and responded to user feedback. Grew a large user base.
  - Closely collaborated with several product engineering groups at Sun.
  - Filed 11 patent applications.
  - Wrote several academic papers and reports on JFluid.

- Principal investigator for the Javamake project (http://www.experimentalstuff.com/Technologies/JavaMake) - a smart dependency checking tool for Java.
  - Designed and implemented the tool from scratch.
  - Successfully completed numerous intermediate releases of Javamake, adding features and fixing bugs on user requests.
  - Wrote and presented an academic paper on Javamake at the leading OOPSLA 2002 conference.

- Principal investigator for the HotSwap research project - the technology for modifying classes on-the-fly in a running JVM.
  - Completed the VM-side hotswap functionality in JDK 1.4.
  - Participated in the implementation of full speed debugging functionality in JDK 1.4.1.
  - Developed a prototype GUI HotSwap Tool (http://www.experimentalstuff.com/Technologies/HotSwapTool)
  - Wrote and presented an academic paper on this technology.

- Academic collaboration:
  - Together with two academic researchers, organized and served on the Program committee of the First Workshop on Unanticipated Software Evolution (USE), held at the 16th European Conference on Object-Oriented Programming (ECOOP 2002). Member of the Program committee of the Second USE Workshop.
  - Reviewed and co-reviewed research papers submitted for several ECOOP and OOPSLA conferences.

August 2000 - June 2001
Sun Microsystems, Inc. - Sun Labs, Mountain View, CA
Student intern, SW (in parallel with my Ph.D. completion)

- Started working on the HotSwap functionality in JDK 1.4.
- Analyzed various policies for method replacement and other aspects of hotswapping that may make it suitable for updating critical applications running in the field.

August 1999 - November 1999
Sun Microsystems, Inc. - Sun Labs, Mountain View, CA
Student intern, SW

- Worked in the Persistent Java research project.
- Ported the (previously developed as a part of my PhD research) persistent class and data evolution technology/tool to the new major Persistent Java release based on the JDK 1.3.

TECHNICAL SKILLS:

- Programming languages: Java, C/C++ - active knowledge. Also used PHP, Bash, Lisp, x86 and Sparc assembler.
- OSs: Linux, Windows, Solaris.
- Tools: IntelliJ IDEA, NetBeans, Eclipse, Perforce, CVS, Teamware, JDK, Visual C++, OptimizeIt, JProfiler.

SELECTED PUBLICATIONS

1[redacted], "The First Experience of Class Evolution Support in PJama". Published in the "Advances in Persistent Object Systems - Proceedings of the Eighth International Workshop on Persistent Object Systems (POS-8) and The Third International Workshop on Persistence and Java (PJAVA-3)", R.Morrison, M.J.Jordan and M.P.Atkinson (Eds.), Morgan Kaufmann, August 30 - September 4, 1998, Tiburon, California.

2[redacted], M. Atkinson, "Evolutionary Data Conversion in the PJama Persistent Language". In Proceedings of the 1st ECOOP Workshop in Object-Oriented Databases□, Lisbon, Portugal, June 1999.

3.[redacted], C. Hamilton. "Towards Scalable and Reliable Object Evolution for the PJama Persistent Platform". Published in the Proceedings of the ECOOP 2000 Symposium on Objects and Databases, in Association with the 14th European Conference on Object-Oriented Programming, Springer Verlag, LNCS 1944, Cannes, France, June 2000.

4. M. Atkinson, [redacted] C. Hamilton and T.Printezis, "Scalable and Recoverable Implementation of Object Evolution for the PJama Platform". Published in the Proceedings of the Ninth International Workshop on Persistent Object Systems: Design, Implementation and Use (POS-9), Springer Verlag, LNCS 2012, Lilihammer, Norway, September 2000.

5.[redacted], "Safe Class and Data Evolution in Large and Long-Lived Java Applications". Ph.D. Thesis, Department of Computing Science, University of Glasgow, Glasgow G12 8QQ, Scotland, and Technical Report TR-2001-98, Sun Microsystems Laboratories, 2600 Casey Avenue, Mountain View, CA 94043, USA, 2001.

6.[redacted] "Towards Flexible and Safe Technology for Runtime Evolution of Java Language Applications". In Proceedings of the Workshop on Engineering Complex Object-Oriented Systems for

Evolution, held in Association with OOPSLA 2001 International Conference, Tampa Bay, Florida, USA, October 14-18, 2001.

7. [redacted], "Application of the HotSwap Technology to Advanced Profiling". In Proceedings of the Workshop on Unanticipated Software Evolution, held at ECOOP 2002 International Conference, Malaga, Spain, June 10-14, 2002.

8[redacted] "Language-Specific Make Technology for the Java Programming Language". In Proceedings of the ACM OOPSLA 2002 International Conference, Seattle, USA, November 4-8, 2002.

9[redacted], "Design of JFluid: a Profiling Technology and Tool Based on Dynamic Bytecode Instrumentation". Technical Report TR-2003-125, Sun Microsystems Laboratories, 2600 Casey Avenue, Mountain View, CA 94043, USA, 2003. Available at http://research.sun.com.

10. [redacted], "Profiling Java Applications Using Code Hotswapping and Dynamic Call Graph Revelation". In Proceedings of the Workshop on Software and Performance (ACM Press), Redwood City, California, USA, January 14-16, 2004.

11. [redacted] "Selective Profiling of Java Applications Using Dynamic Bytecode Instrumentation". In Proceedings of the IEEE International Symposium on Performance Analysis of Systems and Software, Austin, Texas, USA, March 10-12, 2004.

SELECTED PRESENTATIONS AND INVITED TALKS

1. I presented all of the conference papers from the list above at the corresponding forums.

2. "JFluid: Profiling Technology and Tool Based on Dynamic Bytecode Instrumentation". Presentation at the same-named Birds-Of-A-Feather session at JavaOne 2004 Conference, San Francisco, CA, July 2004.

3. "JFluid: What's New Since Summer 2003". Presentation at SAP AG Laboratories, Palo Alto, CA, June 2004.

4. "JFluid: How Code Hotswapping May Facilitate Advanced Profiling". Presentation at PERNET seminar at San Francisco State University, San Francisco, CA, October 2003.

EDUCATION

- Ph.D. in Computer Science, University of Glasgow, UK, 2001
- Diploma in Computer Science, University of Cambridge, UK, 1997
- M.Sc. in Structural Engineering, Moscow State Technical University, Russia, 1996