Filip Jeremic

Contact Information	filip@jeremic.ca	https://jeremic.ca https://github.com/fjeremic

Interests

Compiler development (static and dynamic), programming languages (design and implementation), machine learning, data science, parallelization, computer graphics, software protection, reverse engineering, and malware analysis.

Technical Skills

C, C++, x86-64 Assembly, z/Architecture Assembly, C#, Java
Expert, 10+ years

These technologies are the ones which I have used the most over the years. Because of my field of interest, low level programming languages such as various flavors of assembly, C, and C++ have been my primary languages of choice. Working on a just-in-time (JIT) compiler for Java has made me deeply familiar with these languages both from the angle of software engineering and performance optimization.

Linux, Unix tools, Windows, z/OS
Advanced, 8+ years

My primary development platforms is Linux. Most of my current work involves development on non-desktop architectures so working on a remote machine via SSH is second nature to me. I am very comfortable within a Unix environment carrying out tasks such as instruction level performance investigations (perf), assembly level debugging (gdb, IDA, Ollydgb, etc.), and remote development using the technologies listed above.

Docker, Jenkins, CMake, Bash, Qt, Python, JavaScript, Perl, HTML, WPF, UWP
Experienced, 4+ years

These technologies I have used in various ways to aid in my work, however they are not my primary tools and I do not consider myself an expert in any of them. However, I feel that I am proficient in all of them and often use them to my advantage when they fit the task at hand.

Professional Experience

IBM, Toronto, Canada

Compiler development - Advisory Software Engineer **2017 - Present**

A natural progression of my previous position into a team leadership role within the backend area of the JIT compiler technology I had been working on for several years. During this exciting time we worked on open sourcing our compiler technology as part of the Eclipse OpenJ9 and Eclipse OMR projects at GitHub. As the open source communities grew, my role has been evolving into being one of the focal points for cross-platform backend JIT development and community management within the realms of my expertise. I am currently the top code contributor across both projects as per the GitHub statistics and I am heavily involved in reviewing contributions to both projects.

Compiler development - Staff Software Engineer **2015 - 2017**

An extension of my previous role with a broader focus primarily on performance acceleration of Java workloads on Linux and z/OS. Among development items this role involved instruction level profiling of Java applications and identifying bottlenecks which can be optimized in the JIT compiler. The performance investigation is done at an instruction level but common intermediate language (IL) level optimizations were developed as part of the experimentation such that all supported backends (x86, Power, z/Architecture, ARM, AArch64, RISC-V) can benefit.

Part of my role also involved sharing the burden of some of the team leads responsibilities in technical management of work items triaged to the rest of the development team, and the delivery of performance improvement targets for the release of Java 9.

Compiler development - Associate Software Engineer   **2013 - 2015**

Directly after graduation I took a position at IBM Canada's compiler group where I worked on the backend of the just-in-time (JIT) compiler for the IBM J9 Java Virtual Machine (JVM) in support of IBM platforms.

- Worked on a team developing the backend code generator for z/Architecture based processors
  - Designed and developed various compiler features ranging from intrinsics libraries, register allocation, instruction scheduling, and optimal instruction selection for the target processor
  - Carried out instruction level performance investigations on various workloads and implemented changes to the backend code generator to realize the performance gains
  - Postmortem core dump analysis of non-deterministic code generator bugs
- Developed a firm understanding the interaction between the various components of a dynamic runtime environment (VM, GC, JIT), particularly in the context of a JVM

PATENTS

- **Object load introspection using guarded storage**
  Patent No. US11080182B2   **Issued 2021-08-03**
- **Software-directed value profiling with hardware-based guarded storage facility**
  Patent No. US20210208927A1   **Issued 2021-07-08**
- **Copying and forwarding for concurrent copying garbage collection**
  Patent No. US10877884B2   **Issued 2020-12-29**
- **Multi-byte compressed string representation**
  Patent No. US10002010B2   **Issued 2018-06-19**

EDUCATION

McMaster University, Hamilton, Ontario, Canada

Master's Student   **2012 - 2013**
M.Eng., Computer Science

McMaster University, Hamilton, Ontario, Canada

Undergraduate Student   **2008 - 2012**
B.Sc., Honours Mathematics and Computer Science

References available upon request.