

Research Statement

Ben L. Titzer

I enjoy and intend to pursue systems research in software, from compilers and language implementation to operating systems and virtual machines, including verification, instrumentation, optimization, and analysis. I have pursued research projects concerning these topics with a focus on embedded systems where software challenges are the most severe—and I believe—the most clear, and this has always served to inform and expand my wider interest and willingness to develop and apply such techniques for other areas, from the smallest of software systems to the largest.

Program analysis and optimization. Program analysis has advanced considerably over the past 15 years and is increasingly important in software development for both software quality and performance. Based on my experience in the embedded domain where software errors are relatively more dangerous and dynamic instrumentation is more difficult, I believe that advances in verification such as static type systems and software model checking represent real progress on the grand challenges of software development, and will be increasingly important in the next decade. Further, the changing landscape of desktop and server computing is placing new demands on optimization technology, particularly memory latency and concurrency. Few people can reliably predict the directions in which computing will develop in the next decade, but I believe that across the spectrum of software, program analysis will have a large role to play, from utilizing the full resources of the next generation of processors and memory hierarchies, to sophisticated space optimizations for the smallest of systems.

Virtual machines and runtime systems. The value of dynamic languages and their implementations cannot be underestimated. Languages such as Java and C#, employing and building on the pioneering technology of Self, Smalltalk, Lisp, and Scheme, have finally established beyond doubt the value of virtual machines to industry and the wider software development community. I have always had a research interest in improving the efficiency and architecture of virtual machines, with particular focus on reducing complexity without reducing functionality. I believe that many of the new tasks for which virtual machines are being deployed, such as high performance computing, large data sets, and embedded devices, may require fundamental redesign and reengineering, both at the subsystem level and at the architectural level.

Language design and tools. Software development is a complex process with tensions between concerns such as flexibility, complexity, and efficiency, and the choice and usage of programming languages inevitably affects all factors. I don't believe that language design is fully art or fully science, only that by constant design and evolution we can find more efficient ways to develop and maintain software. One thing is clear; that our challenges—and our imaginations—are constantly outstripping our languages and tools. I believe that both developing new languages and applying existing language technology in new ways holds the promise of better productivity and more reliable software systems.