

Felix S. Klock II

42 Pinecliff Dr, Marblehead, MA, 01945 USA, phone: +1 857.472.3757, e-mail: felix.klock@gmail.com

OBJECTIVE	Innovative software development, leveraging my experience in compiler and language runtime technology, in a challenging environment with an enthusiastic, smart, and respected peer group.
SOFTWARE DEVELOPMENT SKILLS & INTERESTS	Programming languages: runtime design, memory management, JIT, static analysis Software engineering: functional programming, debugging tool design, CS education Languages: Rust, C/C++, Scheme/Lisp, Java, Python, C#, assembly (x86, ARM).
EDUCATION	Northeastern University , CCIS, Boston, MA 2003 – 2010 <i>Doctor of Philosophy in Computer Science</i> Massachusetts Institute of Technology , Cambridge, MA 1996 – 2001 <i>Bachelor of Science in Computer Science</i> , 2000 <i>Master of Engineering in Electrical Engineering and Computer Science</i> , 2001
PROFESSIONAL EXPERIENCE	Mozilla , Paris, France and remote 2012 – present <i>Staff Research Engineer, Rust language</i> Roles: (1) Rust compiler team co-lead, (2) Rust compiler developer, and (3) Language design team member. Project home page: https://www.rust-lang.org/ Project history: http://github.com/rust-lang/rust/ Achievements: Compiler team management, triaging issues and chasing down bugs. Spun up working groups: <i>wg-prioritization</i> (delegating compiler bug triage) and <i>wg-incr-comp</i> (improving incremental compilation in Rust). Oversaw non-lexical lifetimes (NLL; Rust RFC 2094) development and staged migration from lexical to NLL. Designed future-incompatibility reporting (Rust RFC 2834), user-defined destructors with type and lifetime parameters (Rust RFC 769) and Rust's user-defined allocators (Rust RFC 1398). Adobe Systems Incorporated , Waltham, MA 2010 – 2012 <i>Computer Scientist, Actionscript 3 Virtual Machine (aka Tamarin) for Flash Runtime</i> Roles: (1) Memory Management and Garbage Collection (GC) expert, (2) JSON, Array, ByteArray libraries, (3) integration lead, and (4) cross-platform build. Project history through May 2012: http://hg.mozilla.org/tamarin-redux/ Achievements: Improved telescoping GC inverse load-factor, reducing overhead from 20× to 4× mark/cons ratio. Implemented Native JSON integrated with AS3, and extended with serialization of public members of AS3 classes. Added JIT-support for efficient indexing of “simple dense” <code>ArrayObject</code> , yielding 10–20% speedup. Northeastern University , Boston, MA 2005 – 2010 <i>Developer and maintainer of Larceny Scheme compiler and runtime system</i> Project home page: http://www.larcenists.org/ Project history: http://github.com/larcenists/larceny/ Achievements: Designed regional GC, with formal bounds on mutator utilization (MMU). Revised Intel x86 backend to emit machine code in-heap. Developed and evaluated four alternative x86 calling conventions, yielding $\geq 10\%$ speed boost. Implemented dynamic in-heap .NET bytecode emission for Common Larceny.

	<p>Northeastern University, Boston, MA 2003 – 2009 <i>Instructor of Record/Teaching Assistant</i></p>
	<p>Green Hills Software, Santa Barbara, CA 2001 – 2003 <i>Software Engineer for End-User Compiler Product Development</i> Achievements: Implemented move-coalescing register allocation, data-load optimizations, and reassignment of zero-initialized and uninitialized arrays to blank static segment (bss).</p>
	<p>Massachusetts Institute of Technology, Cambridge, MA 2000 – 2001 <i>Teaching Assistant</i> Assistant for 6.170, <i>Laboratory in Software Engineering</i>, Spring, Fall 2000, Spring 2001. Head Teaching Assistant Fall 2000.</p>
	<p>MIT Laboratory for Computer Science, Cambridge, MA 1999 – 2001 <i>Undergraduate Research Assistant for Computer Architecture Group</i> Assisted with implementation of <i>FLEX</i> compiler for transforming Java byte-code to machine code .</p>
	<p>MIT Media Laboratory, Cambridge, MA 1997 – 1998 <i>Undergraduate Research Assistant for Software Agents Group</i> Helped develop <i>Footprints</i>, a tool for visualizing navigation of users on Web.</p>
RESEARCH PUBLICATIONS	<p>Felix S Klock II, “A Declarative DSL for Customized Rendering of Text-Based Art”, in <i>Proceedings of the 2017 International Symposium on Practical Aspects of Declarative Languages</i>, PADL 2017, Paris, France January 2017.</p> <p>Felix S Klock II and William D Clinger. “Bounded-latency regional garbage collection”, In <i>Proceedings of the 2011 Dynamic Languages Symposium</i>, DLS 2011, Portland, OR, 24 October 2011, pages 73-83.</p> <p>William D Clinger and Felix S Klock II. “Scalable Garbage Collection with Guaranteed MMU”, In <i>Proceedings of the 2009 Workshop on Scheme and Functional Programming</i>, Northeastern University, Boston, MA, 22 August 2009</p> <p>Felix S Klock II, “The Layers of Larceny’s Foreign Function Interface”, In <i>Proceedings of the 2009 Workshop on Scheme and Functional Programming</i>, Victoria, British Columbia, 20 September 2008</p>
PRESENTATIONS	<p>“Rust: Fearless at all levels” Algolia Search Party 2018 https://www.youtube.com/watch?v=9QuI2Z0stbs&t=2755 “Subtyping in Rust and Clarke’s Third Law” Rust Fest 2016 https://www.youtube.com/watch?v=fI4RG_uq-WU “Rust: A type system you didn’t know you wanted” Curry On 2015 https://www.youtube.com/watch?v=Q71QCgnNWU0&t=152s</p>
HONORS AND AWARDS	<p>Teaching Award, Northeastern University, 2008 Northern Telecom/BNR Digital Systems Laboratory Project Award, 2001</p>
INTERESTS	<p>Cooking; reading, especially historical discussions of mathematics, logic, and language development; skiing; graphics programming</p>