Miles Gordon Bader

Kawasaki, Japan

miles.bader@gmail.com

https://github.com/snogglethorpe
http://linkedin.com/pub/miles-bader/10/226/228

http://snogglethorpe.net/resume-links-201307

Education

Carnegie-Mellon University, Pittsburgh, PA;
B.Sc. in Applied Math / Computer Science, 1988

Experience

Systems Programmer — NEC / NEC Electronics / Renesas Electronics
November 1997 – Present (Tokyo, Japan)

▪ Designed and implemented a compiler infrastructure, which was the basis for a new C++ compiler at NEC. ▪ Ported Linux to the NEC V850 CPU architecture and various hardware environments using the V850; this included implementing many other necessary features and tools, such as fixing and extending V850 support in GCC, GDB, binutils, and the GDB simulator, a lot of work on uClibc and busybox etc. ▪ Ported GCC to the NEC 78K CPU architecture, and implemented an assembler, disassembler, and linker support for it.
▪ Wrote tools for benchmarking and analysis of various compilers for NEC processors.
▪ Wrote a highly-detailed simulation of the FlexRay embedded bus standard.

Systems Programmer — Free Software Foundation
February 1995 – November 1997 (Cambridge, Massachusetts, USA)

Part of a team designing and implementing the GNU Hurd operating system (an innovative multi-server OS running under the Mach kernel); wrote user tools, file-systems, low-level library support, tool support, and other infrastructure.

Research Associate — Human Communication Research Centre, Univ. of Edinburgh
February 1991 – February 1994 (Edinburgh, Scotland)

▪ Produced a CD-ROM of the Centre's large database of high-quality digitized speech, writing software to do high-quality sample-rate conversion, and user tools for doing rough segmenting.
▪ Ported CMU Common Lisp from Mach to SunOS. ▪ Implemented a machine independent declarative disassembler for CMUCL. ▪ designed various tools for linguistic analysis.

Systems Programmer — Information Technology Center, Carnegie-Mellon University
May 1987 – January 1991 (Pittsburgh, Pennsylvania, USA)

▪ Worked on the Andrew User-Interface Toolkit ("ATK"), from writing applications to low-level work on the underlying object system and toolkit internals.
▪ Designed and implemented a constraint-based WYSIWYG text layout object.
▪ Designed and implemented of OOPC, an object-oriented C front-end.

Computer Experience

* = Extensive experience

▪ Languages: C++*, C*, Lisp*, Lua*, Scheme*, Assembly*, Java, Objective-C, Perl, AWK, tcl
▪ Environments: Linux/Unix*, Windows, MacOS, DOS, TOPS-20, embedded platforms
▪ Fields/Subjects: ⋆ Highly experienced in compiler/programming-language and operating-system implementation ⋆ Very skilled at both high- and low-level (embedded, kernel, etc) debugging ⋆ Free/Open-Source Software (working on Emacs, Linux, GCC, binutils, and others) ⋆ Graphics (including Monte-Carlo pathtracing) ⋆ GUI toolkit design

Languages

Native English speaker; conversational Japanese

Hobbies

Photography, Reading, Travel, Writing Graphics and FOSS Software