

## Peter M Klausler

**Objective:** Architecture and development of high-performance computers and systems software

*Sr. Principal Engineer* – Cray Research → SGI → Cray Inc. (Minnesota)

1992 →

- Discovered, validated, and implemented new routing algorithm for arbitrary networks that is minimal, balanced, fault-tolerant, and deadlock-safe without additional channels.
- Designed and implemented routing scheme for large “fat tree” network of the Cray x2.
- Architect of Cray x2 operating system design. Led port of Linux 2.6 kernel to Cray x1 and x2 architectures; wrote most architecture-specific and assembly code.
- Developed high performance simulators as Cray x1 and x2 software development platforms using on-the-fly binary translation (x86, MIPS, SPARC) and distributed processing.
- Wrote complete software development kits for Cray x1 and x2 architectures including: optimizing and vectorizing C compiler compatible with GCC, assemblers, ELF linker, and a simplified multiprocessor UNIX kernel and runtime.
- Primary software representative on team that designed the Cray x1 instruction set architecture, encoding, and system interfaces. Wrote x1/x2 ISA summary paper.
- Developed and debugged microkernel code and firmware for Alpha-based Cray T3E system.
- Researched future architecture and compiler technologies: built prototype compilers, models, simulators, and binary translators. Invented technique of incremental global register coloring of SSA code interleaved with instruction scheduling. Participated in port of Alpha code generator to SPARC.
- Implemented Fortran 90 array processing in the Cray CFT77 compiler.

*Sr. Software Engineer* – Cray Computer Corp. (Colorado Springs)

1989 – 1992

- Principal architect and developer of a novel automatic vectorizing compiler for Fortran and C on the Cray-3. Invented symbolic data dependence analysis techniques, partial and conditional vectorization, a structural data flow analysis framework, and an intermediate/internal program representation for a modular compiler design.
- Developed a new runtime control library for fine-grained microtasking.

*Sr. Programmer/Analyst* – Cray Research (Minnesota)

1983 – 1989

- Developed new polycyclic scheduling technique for Cray-2 vector loops.
- Implemented vectorization, instruction scheduling, and new language features in the Cray Pascal compiler. Retargeted compiler to support multiple architectures.
- Implemented Cray Prolog interpreter/compiler hybrid.

*Intern* – Sperry-Univac (Roseville, Minnesota)

1981 – 1983

- Compiler test development for various programming languages.

### Education, etc.

- Bachelor of Mathematics with distinction (3.55 GPA), University of Minnesota Institute of Technology, Minneapolis 1981 – 1984
- GRE scores: verbal 730, quantitative 780, analytical 800, CS subject 770, December 1986
- U.S. citizen, no security clearances
- Zertifikat Deutsch (German), some French and Italian
- U.S. Patent 6,308,250 for a vector machine instruction
- Winner, 2005 Int'l Obfuscated C Code Contest, *Worst Abuse of the Rules* category