

---

# Ali Çehreli

[acehrel@gmail.com](mailto:acehrel@gmail.com)

## Skills

- *Programming languages*: Extensive experience with C++, C, and D since 1997, 1989, and 2009, respectively. I am familiar with C++11, C++14, and Python as well.
- *Technologies*: User-mode C, C++, and D programs on Linux systems; configuration management systems (**TMS** and proprietary others); **Boost**, libevent, libxml, and other libraries; DDoc, GNU make, BoilerMake, distcc, gdb, valgrind, and other tools
- *Program correctness*: Unit testing; exception-safe C++ API design and implementation

## Employment

### **Weka.IO, June 2016 - present**

*Principal Software Engineer*

- Helping build *Radically Simple Storage™*.

### **Riverbed, June 2007 - May 2016**

*Member of Technical Staff, Device management framework*

- Designed and implemented C++ library APIs to improve program correctness and developer productivity over existing C libraries. Introduced the Boost libraries to the development environment. Improved developer productivity by automating device configuration tree registrations through XML and C++ APIs, and automatic upgrades and fallbacks of device configurations.
- Improved program responsiveness by enabling more kinds of non-blocking tasks and by implementing an event-based process startup dependency graph.
- Improved software quality by integrating a unit testing framework (Unittest++) into the build process, wrote a mock file system library to enable program and library testing at build time. Introduced Doxygen to the code base. Wrote the coding guidelines document, acted as a resident C++ expert, gave presentations on C++ exception safety and the D programming language.
- Interviewed 200+ engineering candidates.

*Member of Technical Staff, SteelFusion*

- Worked on the configuration and management system of *SteelFusion*. Implemented configuration modules like *Hypervisor Image Vault*, which manages hypervisor ISO images.

### **VMware, 2006 - 2007**

*Member of the Technical Staff, VMware Converter*

- Member of the team developing VMware Converter, a client/server physical-to-virtual computer migration system. Implemented the job scheduler to manage the execution of asynchronous machine conversion jobs, implemented job status message transfers from the server to the user interfaces, implemented the archiving of job status information as XML files. Helped incorporate Converter to VMware's Virtual Center product.

## **PayPal, 2004 - 2006**

*Staff Software Engineer, Chinese PayPal site*

- Participated in *the UTF-8 strings* project that integrated the [ICU libraries](#) of IBM to hundreds of source files to make the PayPal site Unicode-aware.
- Worked on the Chinese PayPal site by localizing various existing features of PayPal to the Chinese language and culture. Implemented the Chinese site signup and referral bonus programs.
- Worked on various other parts of the PayPal system from the back-end financial batch processing to front-end web interfaces.

## **NetContinuum, 2002 - 2004**

*Senior Software Engineer, Application security gateway*

- One of the primary contributors of the team that developed the configuration agent of the *application security gateway* product. Redesigned and implemented the statistics collection framework; optimized the execution speed and reduced development and maintenance efforts. Redesigned the XML messaging framework; improved the communication with the user interfaces. Abstracted the external representation of the configuration tree. Designed and implemented a simple and safe output stream wrapper to increase the speed of string concatenations by a factor of ten. Improved the quality and robustness of the software; refactored the source code. Removed memory leaks; used valgrind and constantly inspected the source code for potential developer errors.
- Acted as the in-house C++ expert. Gave presentations on C++ good practices. Introduced the use of modern C++ idioms, design patterns, and libraries. Increased the overall understanding of the safe and robust software practices.
- Improved the build environment. Automated target dependency generation in the make files. Tripled the build speed by introducing distcc.

## **Inviscid Networks, 2001 - 2002**

*Lead Software Engineer, Packet forwarding engine ASICs*

- Principal designer of network processor verification software and network processor development tools.
- Lead contributor to architecture, design, and implementation of full suite of integrated network processor development tools.
- Contributor to the architecture of network processor ASICs.
- Developed run time libraries for table management and hardware abstraction layer for a family of network processors.

- Designed and implemented an unlimited-width and platform independent bit-field class to represent the varying widths of hardware memory blocks, implemented patricia trie, hash table, and other data structures.
- Abstracted the hardware layer structures as logical operators and functional blocks of a graphical decision tree.

## **Berkeley Networks (Fore and Marconi by acquisitions), 1996 – 2001**

*Senior Software Engineer, Network switches and routers*

- Fundamental contributor from the earliest design stages of a family of network processor ASICs. Contributed to the development of a family of application aware Ethernet switches. Proposed optimizations that were later implemented in successor chip development.
- Implemented low level protocols and tools for ASIC verification, manufacturing diagnostics, and fault isolation. Implemented low level protocols to communicate with line cards over both serial and ethernet ports. Developed tools for ASIC verification, manufacturing diagnostics, and fault isolation. The test methods and programs developed were leveraged from early phases of ASIC design and verification to the eventual deployment into 3rd party manufacturing process flows.
- Developed a functional test program to run on a single inexpensive PC to stress test the routing and bridging protocols of a multi-Gigabit switch. Lead developer of diagnostic stress tests utilizing avalanche traffic generation methods within the switch. Hardware multicast, COS queue scheduling and multiple programmable loop-back interfaces were used to advantage. The test program could automatically isolate faults to specific chips and interfaces within the system by dynamically adjusting the program parameters based upon incorrect returned packet data. Stimulus and verification data could be injected and returned at program controlled rates on a single link to enable simple test bench environments suitable for manufacturing.

## **ITA - Nokia Monitors, 1994 - 1996**

*Software Engineer, Order-entry and invoicing system*

- Programmed a proprietary order-entry and invoicing system. Integrated the system with FedEx's shipping applications. Sole maintainer and improver of a proprietary order-entry and invoicing system written in the MS FoxPro database language. Integrated the system with FedEx's shipping applications to automatically update the status of orders in the database. Designed, configured, and maintained a WAN connecting the branch offices. Connected the system to company's remote offices via modems and routers for automatic data entry.

## **Opera Computer, 1991 - 1994**

*Founder and Software Engineer, WordPerfect products*

- Co-founder of the WordPerfect reseller in Turkey. Localized the products of WordPerfect Corporation. Translated the word processor, database, spreadsheet, and other applications to Turkish. Provided customer support and product training.

## **Karel Computer, 1989 - 1990**

*Software Engineer, Digital Research products*

- Localized the products of Digital Research Corporation. Translated the DR DOS operating system and the entire set of GUI products of Digital Research Corporation, as well as some of the user manuals of those products to Turkish. Patched pre-compiled object files to support Turkish keyboard layout and to enable copy protection for applications.

## **Other experience**

### **Mill Computing, May 2014 - present**

*Software Engineer, Mill CPU*

- Worked on the simulation environment, syscalls, runtime, and various tools of the Mill, a revolutionary CPU architecture.

### **IoTone, May 2016 - present**

*Software Engineer, IoT devices*

- Added publisher/subscriber support to Tiny-Redis, a D module for Redis communication.

## **Personal projects and accomplishments**

### **C++ and C Programming Languages**

- Former president of [Silicon Valley Chapter of ACCU](#) (2012-2015); former membership coordinator (2002-2015); four-time speaker (2004-2015)
- Author of the Trends column and several C++ articles for the ACC++ent newsletter (2001-2003)
- Former moderator of [the C and C++ forum](#) on Ceviz.net (2002-2014)
- Attendee of [Silicon Valley Patterns Group](#)
- Author and translator of [C++ articles](#)

### **D Programming Language**

- Author of the books [Programming in D](#) (textbook at University of Minnesota and [Utah Valley University](#)) and its original [D Programlama Dili](#)
- Secretary and board member of The D Language Foundation
- Main organizer of [Silicon Valley D Lang Meetup](#) group
- Speaker at [DConf 2013](#), [DConf 2014](#), [DConf 2016](#), and D Lang meetups
- Founder of [Ddili.org](#)
- Moderator of D forums on [Ddili.org](#)
- [Personal projects](#) including the generation of ddili.org as well as an experimental Unicode library for collation and capitalization of various writing systems

## Education

- M.S., Physics, [Istanbul Technical University](#), 1994. Developed tools and emulation programs to study fractals, dynamical systems, neural networks, and cellular automata.
- B.S., Electronics Engineering, [Istanbul Technical University](#), 1989. Designed microprocessor controlled stopwatch for track and field sports.