Gary Hoo

637 Liberty Street
El Cerrito, CA 94530
(510) 526-5426

gjhoo@lbl.gov
http://www-itg.lbl.gov/~hoo/

Objective

A software engineering position developing networked/distributed systems.

Experience

Computer Science Engineer
1998- present

Lawrence Berkeley National Laboratory, Imaging and Distributed Computing Group

- Modified and extended Java applications that create certificates for use in a fine-grained access control infrastructure (Akenti).
- Designing and implementing a reservation-based bandwidth brokering system that will use the Akenti access control system to control premium bandwidth allocation to users and processes in a wide-area, heterogeneous, distributed computing environment.

Graduate Research Assistant
1993- 1997

Lawrence Berkeley National Laboratory, Imaging and Distributed Computing Group

- Designed and implemented (in C++) an object-oriented, multithreaded application to store and to replay MBone audio and video sessions using LBL's Distributed-Parallel Storage System (DPSS). Application implements the Real-time Transport Protocol, version 2 (RTPv2), including the RTP Control Protocol (RTCP), to perform MBone communication. Multithreading via POSIX threads facilitates both synchronized playback of associated audio and video streams, and parallel access to DPSS. (Master's thesis project.)
- Designed, implemented and maintained various DPSS client applications for debugging, as sample code for other client developers, and for network load testing and analysis. Client applications write data to and/or read data from the DPSS, allowing stress-testing of the DPSS, its client-side library interfaces, and the underlying network.
- Co-designed, implemented and currently maintaining low-level C-language API ("ISS API") to DPSS used by Multidimensional Applications and Gigabit Internetwork Consortium (MAGIC) project collaborators. ISS API hides network-level details of client-DPSS interaction (e.g., byte ordering) while permitting fine-grained control of data transmission and receipt if desired. XDR extensively used to encode complex data structures exchanged between client and DPSS.
- Designed, implemented, and currently maintaining higher-level C-language API to DPSS ("ISSFS"), built on ISS API. ISSFS API emulates Unix filesystem semantics, allowing non-MAGIC developers to use a less complex and more familiar interface. High-energy nuclear physics researchers and medical technicians use ISSFS to access DPSS as a high-speed cache for both raw experimental data and intermediate analysis results.
- Co-author of several papers about the DPSS and its uses in MAGIC and other environments. See http://www-itg.lbl.gov/~hoo/CV.html#publ for more details.

Laboratory Consultant
1992-1993

San Francisco State University Advanced Computing Laboratory

Student Researcher
1991-1992

San Francisco State University Computer Science Dept.

Implemented high-level language interface in lex and yacc for specifying neural network connection types. Interface served as front end to neural network application, allowing user to specify network configuration (e.g., feedback types and number of feedback levels) using a simple, custom programming language.

Paralegal
1987-1990

Hancock, Rothert & Bunshoft (San Francisco)

Areas of Knowledge

Languages
- C, C++, Objective C, HTML, csh, sed, awk

Network Protocols
- TCP/IP, IP multicast, RTP

Unix Variants
- SunOS 4.x, Solaris 2.x, DEC OSF/1, IRIX 5.x and 6.x, FreeBSD

Education

M.S. *expected 1998*
San Francisco State University
Computer Science

A.B. *1987*
Harvard College
English and American Literature and Language