Brian Kroth - Curriculum Vitae

Contact Info

Brian Kroth	
Email	bpkroth@gmail.com
Homepage	http://cs.wisc.edu/∼bpkroth
GitHub	https://github.com/bpkroth
LinkedIn	https://www.linkedin.com/in/brian-kroth-80141546
MSR	https://www.microsoft.com/en-us/research/people/bpkroth/

Overview

I started my career mostly as a self directed sysadmin with special emphasis on automation, monitoring, and performance. Over time I got involved in the research community and from there shifted into a more formal applied research role. I have a fairly diverse background and skill-set, but am mostly a systems nerd at heart, with a keen interest in building and measurement for the sake of performance, optimization, and understanding. I love all things distributed, storage, networking, kernel bypass, virtualization, memory, caching, etc. I've often focused on database systems through this lens. Based on my experiences, I have an ability and love to learn and adapt new technologies based on underlying principles. In the past I have taught myself to be a deep expert in underlying CPU caching details, some machine learning for optimization purposes, etc. I am a good communicator and capable of leading teams, but really love to get my hands dirty. In particular, I love to operate in a Linux and `git` first oriented environment. Some technologies I'm particularly interested in working with in the future include `io_uring`, `ebpf`, and `k8s`.

Education

01/2010-05/2014	University of Wisconsin-Madison Master's degree in Computer Science
09/2003-05/2007	University of Wisconsin-Madison Bachelor of Science, Theoretical Mathematics and Computer Science

Research

GSL — Various applied research projects. Some examples include:

- Autoscaling analysis of SQL databases in Azure for improved resource management and utilization (policy).
  Fast scaling of local storage SQL databases in Azure Service Fabric (mechanism).
- Mechanisms for communicating fine grained application resource demands across a virtualization layer (vsock).
  Analysis of power utilization improvements by controlling CPU frequencies using the same.
- Several storage and hardware emerging hardware investigations (NVMe, NVM, AVX, FPGA, etc.).
- Several efforts into learned and adaptive indexes and data structures, both in-memory (VIP hashing) and on-disk (Learned LSM).
- MLOS: started and helped lead a large effort on machine learning based auto tuning for applications and OS tunables, and low level software data structures for several products in Azure Data and beyond. Many related efforts in the benchmarking, observability, workload characterization and similariy, etc. spaces.
  OSS, Paper: https://aka.ms/MLOS

Seminar: https://aka.ms/MLOS_Seminar
Guest lecturer at MIT, DSAIL

- "Noise Cancelling Canary Application for Improved Performance Evaluation and Autotuning in the Cloud"
  A method for identifying and removing performance interference noise from an application benchmarked in shared infrastructure that does not rely on hardware counters through careful L3 cacheline coloring (patent pending).
- code-book: an interactive notebook experience for large scale code search and analysis
  Used for an updated version of analyzing the millions of notebooks for updated versions of "Data Science through the looking glass and what we found there" finding tunables in source code for MLOS, and several product guidance and engagements including log message frequency analysis for Azure SQL DB.

Student — Class project research:

01/2014-05/2014	An analysis of UW-Madison Moodle platform performance characteristics http://cs.wisc.edu/~bpkroth/papers/moodle-performance-analysis
01/2013-05/2013	Bin Packing: A Survey and its Applications to Load Balanced Job Assignment and Machine Allocation http://cs.wisc.edu/~bpkroth/papers/bin-packing
09/2010-12/2010	Checksumming Software RAID http://cs.wisc.edu/~bpkroth/papers/md-checksums
01/2010-06/2010	OpenSAFE - A distributed IDS system using OpenFLOW. http://cs.wisc.edu/~bpkroth/papers/opensafe-imc10.pdf

UW — A large amount of data driven self directed research and analysis done in the course of work as a systems engineer. Some interesting examples include:

- Ceph cluster investigation and benchmarking and load testing, especially with respect to optimal OSD storage organization for providing RDB backed VM images with small file support and NFS access for mixed small and large file support. Some options explored were things like using SSDs for write-through bcache or OSD journal devices for several hard drives vs a Ceph cache tier and the administrative ease of maintaining each solution.
- In conjunction with bin packing work done in course work, wrote a simulator to determine the number of active Apache worker processes or threads at any point in time depending on the type of MPM in use by replaying access logs.
- Lots of performance analysis and testing in the course of maintaining general web platform system for several UW-Madison campus wide Moodle instances. Among a number of interesting results includes a 66% reduction in latency due to switching from BIOS controlled to hypervisor controlled power management policies. Others include a comparison of various data management systems with relaxed transaction durability options, separate virtual disks to separate IO queues for MySQL VM logs vs data for reduced query latency, SSL cipher suite optimizations as well as entropy keys/daemons, HTTP proxy caching (disk with open file handles vs in memory) optimizations, application cache system (eg: memcache vs mongodb vs NFS vs local disk cache) for performance vs coherency (and developer ease) tradeoffs.
- Wrote a syslog data analysis system to compute accurate distributions of message size, rates, spread, etc. for use in modeling and benchmarking new system redesigns (eg: battery backed hardware RAID with nvram vs software RAID with write through bcache SSD). Used it to help identify metal fragments in a drive slot when

hardware RAID mode wasn't delivering expected performance and software wasn't able to introspect the individual drive statuses.

- Heavy MySQL 5.1 to 5.6 transition research, optimization, automation, and deployment.
- Evaluation of EMC Celerra and later VNX implementation and integration of NFSv3, NFSv4 with Kerberos authentication, and Windows Active Directory systems. Encountered and helped debug a number of locking and permission mapping related bugs.
- Tracked down intermittent IO performance issues through the stack and eventually determined to be caused by dynamic inode allocation failure in a highly fragmented OCFS2 volume used for mail storage. During the course of this I also developed a distributed IMAP benchmarking client and evaluated virtual guest software iSCSI and VMware's VMI paravirtualization.
- EqualLogic performance analysis and rudimentary IO modeling.
- In depth comparison of the behavior and performance of a number of clustered filesystems and their corresponding cluster stacks including: OCFS2, GFS, GFS2.

Publications

Cloudy with high chance of DBMS: A 10-year prediction for Enterprise-Grade ML

Lessons learned from the early performance evaluation of Intel optane DC persistent memory in DBMS

From WiscKey to Bourbon: A Learned Index for Log-Structured Merge Trees

Optimizing databases by learning hidden parameters of solid state drives

MLOS: An Infrastructure for Automated Software Performance Engineering

Technical Skills

Languages	Bash, Python, Perl, C, C++, C#, SQL, PHP, XML, HTML, JavaScript, Java, Golang, Python, LaTeX, and others.
Platforms	ESX, libvirt + KVM, Docker, Azure
OS	Linux, Solaris, FreeBSD, Windows
Applications	Git, systemd, Nagios/Icinga, Collectd, Sysstat, Cfengine, FAI, cloud-init, iptables, arpwatch, ndpmon, Syslog-NG, Exim, Sendmail, Clamav, Dovecot, Perdition, PhantomJS, Apache, Shibboleth, ModSecurity, memcached, Redis, MongoDB, MySQL, Oracle, SqlServer, OpenLDAP, Active Directory, OpenSSL, OpenVPN, BIND, Samba, CUPS, Heartbeat, Pacemaker, Condor, and others.
Protocols	TCP, IP, IPv6, 6to4, VLANs, Firewalls, VPN, HTTP, IMAP, Proxies, SMTP, SSL, DNS, DNSSEC, DHCP, LDAP, SOAP, iSCSI, NFS.
Storage	Ceph, LVM, ZFS, BTRFS, OCFS2, GFS2
Network	Cisco IOS/NX-OS

Work Experience

08/2017-Present	Gray Systems Lab, Microsoft https://aka.ms/gsl https://www.microsoft.com/en-us/research/people/bpkroth/ Senior Research Software Development Engineer

11/2014-08/2016	UW CloudLab http://cloudlab.us Unix Research Administrator
10/2008-08/2017	Computer Aided Engineering Center, University of Wisconsin-Madison http://www.cae.wisc.edu Unix Administrator and Manager
06/2006-10/2008	School of Medicine and Public Health, University of Wisconsin-Madison http://www.med.wisc.edu Network and Server Administrator
10/2004-06/2006	Computer Aided Engineering Center, University of Wisconsin-Madison http://www.cae.wisc.edu Programmer, Netware and Windows Administrator
05/2003-10/2004	Health Sciences Learning Center, University of Wisconsin-Madison http://www.hslc.wisc.edu Programmer, Netware and Linux Administrator
06/2002-12/2008	Devil's Head Ski Resort, Merrimac, WI http://www.devilsheadresort.com Programmer and Network Consultant
09/2001-09/2002	Dane County School Consortium, Madison, WI http://www.dcsc.org Youth Apprenticeship Web Programmer and Network Manager
06/2000-09/2001	Service Electronics Corporation, Verona, WI Youth Apprenticeship Computer and Network Technician
06/2000-01/2001	ECI Software, Verona, WI Youth Apprenticeship Programmer

Highlights and Achievements

GSL

- Part of a small Microsoft database systems lab partnered with a group of professors and students from the Department of Computer Sciences at UW-Madison.
- Involved with various team and individually led systems and database applied research efforts partnering with product teams in the Azure Data organization (primarily SqlServer and Synapse Spark oriented).
- Emphasis on data driven analysis and development with recent focus on machine learning approaches.
- Help students with research efforts in open source spaces (e.g. Linux, PostgreSQL, MySQL/MariaDB) and setting up good and careful measurement experiments that consider small details and effects of the full system.
  Mentored 7+ students (RAs and summer interns) to date.
- Help liaison between local professors and remote management.
- Some amount of local lab hardware/software maintenance (200+ servers and network equipment at one point) as well as efforts to transition most development work to Azure that I continue to help manage.
- Some amount of side efforts to help improve build and test pipelines for SqlServer.
- Worked on data analysis efforts to find autoscaling opportunities for SQL DB SLOs in Azure. Collaborated on some predictive modeling efforts for the same.
- Developed a mechanism to support fast in-place scaling for local storage DBs in Azure (Service Fabric).

- Developed a mechanism for communicating performance requirements between an application and a host through a virtual environment using vsock. Explored a possible use to allow a provider mediated way for applications to signal when CPU frequencies could be adjusted in order to reduce power consumption without giving up on performance.
- Helped mentor interns through several storage analysis research projects including NVMe placement to make use of internal device parallelism and in-depth microbenchmarking of Intel Optane DC persistent memory.
- Helped organize and co-lead a large effort, called MLOS, around applying machine learning techniques to tuning low level software data structures and algorithms used for database systems.
  Collaborated with Intel's Machine Programming lead on the topic.
  Later expanded it to be generally applicable to many systems (e.g. Postgres) by tuning externally exposed settings as well. Have also collaborated with members of the Azure Compute team to tune OS settings for specific workloads, resulting in roughly 25% reduction in latency and jitter for memcached workloads.
  Lead the open-sourcing of an initial version of MLOS to GitHub:
  https://github.com/microsoft/MLOS
  Helped run a virtual seminar class of 150+ at UW-Madison around this topic and lab of 20+ using the software in Fall 2020: https://aka.ms/MLOS_Seminar
  It was also picked up by MIT in Fall 2021, where I gave a guest lecture on the topic.
  Currently working on a refined version of the libraries and using it to help auto-tune the Spark Synapse product.
- Started a related side investigation into "noise cancelling" for improved performance benchmarking in the cloud that resulted in a patent application. The technique uses very carefully controlled L3 cacheline coloring that works through several virtualization layers to allow estimating interference through careful timing using TSC instructions, all without access to hardware counters. Combined with limited offline testing in a controlled environment to produce a model of noise-to-performance loss, we're able to estimate *true* performance when measured in the cloud more accurately, thus enabling auto-tuning at scale.
- Started a related side investigation with an intern into code search so we could find tunables using an interactive interface. The system we built, `code-book`, provides an interactive notebook experience that allows "query by example" style code search over millions of files in seconds for simple queries, and order of minutes for complex ones involving data flow analysis. We used this to enhance several on going efforts within Azure Data to characterize data-science workflows and inform product teams which developments to prioritize based on usage patterns in open-source and first-party scenarios. We are currently engaged in extracting log message templates from various repositories used for providing the Azure SQL DB related services in order to automatically construct Kusto queries for message frequency analysis alerting.

UW CloudLab

- Split time with CAE to build a small Ceph and KVM cluster managed with Libvirt for hosting internal infrastructure to manage testbed machines with Emulab.
- Developed scripts to manage hardware configuration for the 240+ testbed Cisco machines and switches. Open sourced the public version of it here:
  https://github.com/bpkroth/Cisco-IMC
- Helped develop simple profiles for students to use the testbed including paramaterized RSpecs with Python for variable VM sizes and simple OpenStack configurations.
- Worked with researchers, staff, and vendors to design, spec, and purchase, install, and configure testbed equipment.

CAE

- Part of a 3 member Unix and Linux systems staff that provided core platforms for College of Engineering faculty, staff, and students to do their research, teaching, and learning.
- Helped manage 16 node cross site ESX cluster and iSCSI backed SAN.
- Helped manage 200+ Debian Linux VMs and servers for College of Engineering services and workstations for Engineering students primarily using Cfengine, rsync, and a number of custom systems.
- Help manage a staff of student workers focused on Unix and Linux systems maintenance and especially monitoring via Icinga and automation via Cfengine and many custom DB driven systems. Mentored students via code reviews.
- Began redesigning many of our systems to take advantage of systemd with containers instead of VMs in mind for eventual hybrid cloud deployments. Particular areas and tools of investigation/interest include Vagrant, Puppet/Chef/SaltStack, rkt/Docker, Kubernetes.
- Redesigned secure, stable, and scalable web and database hosting infrastructures consisting of over 1500 vhosts. The system uses a central Oracle database of vhost/IP/DNS/user information populated with predominantly user provided information to group and segregate vhosts into separate "security domains" (essentially lightweight proto-containers) according to certain vhost attributes. A combination of complex join queries and views and object oriented libraries provide reasonable developer access to an otherwise old and complex schema. Each group is assigned and bound to its own unrouted IPv6 address and run as its own Apache process with separate uid/gid and filesystem namespace. Each Apache process is assigned to one or more backend VMs that provide service for that vhost group's type. Assignments are done in a load balanced way by using the bin packing and access log simulator research work mentioned above as a feedback mechanism to inform it. Vhosts are proxied through another Apache process running on a separate server that optionally performs caching and input security checks via ModSecurity. A similar load balancing assignment process occurs for frontend Apaches as well.
- Maintained, monitored, and optimized the web system above for several UW-Madison campus wide Moodle instances. Many excursions into system performance analysis and modeling for ensued in the course of this. Some results of which include a handful of Apache mod_proxy_balancer patches, a two-phase commit (presumed commit) protocol and system to synchronize NFS web data to local VM disks in order to avoid expensive `stat()` calls, and many SSL, caching, and SQL optimizations like separately tuned databases based on durability requirements of certain datum like session data and logs.
- Maintained OpenLDAP and MIT Kerberos portion of a complex identity management system derived from a central Oracle database. Have expanded its use to be general enough for College of Engineering wide use. Several bugs and patches submitted around poor application behavior regarding insensitivity for uid LDAP schema attributes used for authentication.
- Setup install (FAI), backup (custom written secure ssh tunneled rsync and ZFS snapshots system), and monitoring (Nagios/Icinga/Collectd) systems whose configurations are almost exclusively derived from Cfengine feedback database (also custom written) and a few service specific databases leading to a highly automated environment in which machines can be spun up in a matter of minutes and are automatically configured, backed up, monitored, etc.
- Created a system for automatically performing SAN based snapshots and fixing up the filesystems contained in them for backup purposes regardless of logical (eg: LVM) filesystem layout.
  http://cae.wisc.edu/~bpkroth/src/san-snapshots/

- Developed a system for correlating users' access to services, to IPs, to MACs, to switch ports based on ideas presented in the Ethane paper (http://yuba.stanford.edu/ethane/pubs.html). The system uses a combination of arpwatch, ndpmon, and log, switch, and router scraping to gather data into a database where reports can be run as desired. The system now also includes integration with failed login records, iptables, GeoIP and Whois data in order to alert security officers of suspicious logins quickly. In process of converting the implementation of this system from a single VM with an 802.1q trunk to a distributed set of canary devices placed in various building IDFs.
- Created a self service system to create and manage various repositories (eg: svn, hg, git, bzr) integrated with Trac (pre gitlab availability).
- Converted old Cfengine CVS repository to use Git along with a slew of custom subcommands and server side push hooks for helping new student employees and other users learn and stay consistent with local workflow rules.
- Used Git repo to help facilitate rapid local machine development via Cfengine for better feature testing and rollout with the goal of bringing the systems development model more in line with standard software development practices (ie: DevOps).
- Patched 9 year old Cfengine v2 code to get more than 50% speed-ups for remote operations by batching `fsync` operations for non-critical metadata: https://github.com/bpkroth/cfengine2
- Custom built system for automating (as much as possible) DNSSEC key and algorithm rollovers complete with implementation that involved replacing single purpose keys for several zones with split KSK and ZSK keys. Integrated with existing custom IP/DNS/DHCP database.
- Implemented a custom system for dynamically generated iptables/ip6tables rules partially based on hostnames rather than addresses and derived from Cfengine classes and a local cache of recent DNS responses in order to account for DNS load-balancer services that may respond with different answers at different times and sites that may temporarily be unavailable. Combined with the login correlation system mentioned above, the system also serves as a sort of distributed fail2ban.
- Redesigned scalable mail infrastructure consisting of over 6000 active users. Redesign focused on segregating mail floods from both internal and external sources from impacting user visible services. We were then able to handle spikes of 10000s of messages occurring in the span of a few seconds.

SMPH

- Designed and implemented an XML-RPC protocol and database system for managing video encoding and publishing of lectures and presentations throughout the facility as well as support for mobile stations. Design details and screenshots available at: http://videos.med.wisc.edu/hvec/.
- Created custom Gentoo Linux encoder images for PPC64 and x86-64 architectures. Build servers maintained using VMWare ESX.
- Created highly available, semi load balanced web and database clusters using entirely recycled hardware and open source software such as Heartbeat, OCFS2, and Multi-Master MySQL.
- Managed Medical Physics 96 node GLOW Condor cluster. Worked with students and researchers to help them get the most use out of the system.
- Managed over 50 physical servers running OS-X, Windows, Netware, and various flavors of Linux such as Slackware, RHEL, Fedora, Ubuntu, and Gentoo.
- Monitored and maintained network, system, and information security for servers and clients using tools like Syslog-NG with SEC and LogWatch, Nagios, Cacti, Nessus, ModSecurity.

- Setup an Active Directory domain to consolidate Windows management and provided a single source of account and group information for all Windows, OS-X, and Linux machines.

DevilsHead

- Created a point of sale application written in Java using extensive MySQL and custom built low end Gentoo Linux machines as terminals. It is now used by Devil's Head and its sister companies to sell, print, and manage ski and golf tickets. Features included a simple DB synchronization protocol to make sure that orders could still be taken locally despite lack of consistently reliable networking in certain parts of the facilities.

Volunteering

01/2015-06/2015 East High School, Madison, WI
Math Tutor
Geometry, Algebra I, Algebra II, Precalculus, Calculus

Selected Personal Projects

03/2016 City of Madison tax assessment outliers analysis
Wrote a small pipeline to find outliers in tax assessment value by scraping the available tax and home/lot specs information off the City of Madison website and running some basic statistics and heuristic comparisons over them for various areas. Used it as the basis for a 12% reduction in my own property taxes and help a few friends and family as well.

01/2022 Quick and simple wordle solver using frequency analysis
https://github.com/bpkroth/wordle-solver

Interests

Computers Filesystems, Storage, and Databases, Virtualization, Networking, Security, Performance, Monitoring, Automation, various open source projects (eg: Gentoo Hardened, Debian, Libvirt, KVM, Fluxbox, GNU Screen, Mutt, Vim, Vimperator, and others)

Mathematics Group Theory, Ring Theory, Number Theory, Set Theory, Logic, Combinatorics

Other History of Science, Feynman, Physics, Quantum Computation, Camping, Canoeing, Hiking, Biking, Baseball, Skiing, Running, Jazz, Brass, various Electronica, and lots of other music

Madison, WI, January 31, 2022