

Xing Lin

CONTACT INFORMATION 3490 Merrill Engineering Building (MEB) office: (801) 581 4285
50 S. Central Campus Drive mobile: (801) 647 1899
Salt Lake City, Utah 84112-9205 e-mail: xinglin@cs.utah.edu

RESEARCH INTERESTS Storage and file system, especially in deduplication and cloud storage systems
Operating systems and computer architecture, especially in multicore related designs.
Virtualization, and distributed systems
kernel programming in Linux

EDUCATION **University of Utah**, Salt Lake City, Utah, U.S.
Doctor of Philosophy **Fall 2009 – present**

- Research advisor: Robert Ricci

Nankai University, Tianjin, P.R.China

Bachelor of Engineering

Fall 2005 – June 2009

- Research advisor: Xudong Lee

RESEARCH EXPERIENCE **Nankai University**, Tianjin, P.R.China

Research Assistant

Oct. 2008 – Sep. 2009

Research on Freeze-drying and Resuming Processes

2008 Microsoft Windows Core Theme Project

This project mainly focuses on how to freeze-dry an executing process and how to load a disk core of a freeze-dried process into memory and resume it. All work is done based on Windows Research Kernel(WRK).

I implemented the function to freeze-dry portable executables(PEs) and dynamic linked libraries(DLLs). We succeeded to freeze-dry and resume a simple loop program and programs opening kernel objects such as files. I also investigated how virtual memory is managed and implemented the function to freeze-dry stacks, heaps and other virtual memory areas for a specific process. I implemented five policies to select a waiting process to run for semaphores. I also investigated the handle mechanism. Based on the above work, we published four papers about freezing and unfreezing for process with multi-file handles, semaphore, handle and virtual memory management mechanisms in WRK.

University of Utah, Salt Lake City, UT, U.S.

Research Assistant

Sep. 2009 – present

High-Performance Disk Imaging with Deduplicated Storage

Apr. 2011 – Jan. 2012

Integrated a deduplication storage system called Venti into Emulab's high-performance disk imaging system called Frisbee. We solved several challenges to achieve efficient deduplication and high-performance disk imaging performance simultaneously.

Understand and optimize the read throughput in deduplication storage systems

Apr. 2011 – present

Disks in deduplication storage systems have a significant performance degradation because deduplication fundamentally fragments stored data. We propose to design a complementary prefetching and adaptive block-reorganization framework to improve the disk I/O performance.

Refining the Utility Metric for Utility-Based Cache Partitioning

Feb. 2011 – June. 2011

the Advanced Computer Architecture course, taught by Rajeev Balasubramonian

Cache misses have varying impacts on IPC across programs because of varying levels of latency tolerance in each program. In this work, we discover a non-trivial number of cases where the use of MPKI is sub-optimal. We then propose a simple mechanism that uses two IPC samples and a miss rate curve to compute an IPC curve. This allows us to better quantify marginal utility in terms of IPC and arrive at performance-optimal cache partitions.

OpenVZ Container Migration in Emulab

Dec. 2009 – present

Implemented an instant container migration mechanism for OpenVZ. LVM-tt is used to transfer the delta of the local file system and VLANs are synchronized with snmpit during migration.

Linux physical memory de-duplication

Dec. 2010

The purpose is to implement content-based page sharing for Linux. A kernel module which calculates a hash for every physical page is implemented for both 32-bit and 64-bit NUMA Linux. Another kernel module which exports the content of specified page is also implemented. After we found Linux has already implemented this feature, we stopped this project.

Integrating OML into Emulab

Sep. 2009 – Dec. 2009

The main purpose is to facilitate users to gather their experimental data distributed in Emulab. Four testbed extension commands are added as OML interface to users. Emulab can automatically generate OML related C structures and functions, which facilitates users to use OML.

COURSE
PROJECTS

Graduate Courses:

Database Kernels and Large Data Management from Feifei Li

Implement B+ tree dynamic insertion and buffer management with love-hate page replacement policy.

Operating System from Matthew Flatt

User-level Thread

Implement a timer, semaphore mechanism, a proportional share thread scheduler and a non-blocking socket calls, based on a user-level Thread

Page Replacement Algorithms

Implemented FIFO, second chance, LRU and optimal page replacement algorithms

Thread-safe File System

Support open, close, unlink, read, write, getpos, truncate and lseek for multi-thread processes

Network Security from Sneha Kasera

Implemented an expanded Needham Schroeder Mediated-Authentication Scheme

Implemented a simplified SSL

Implemented a network discrete event route simulation

Advanced Embedded Systems from John Regehr

Based on Codefire M52233DEMO chip

Power-On Self Test: RAM check and use a checksum to ensure the flash memory was loaded properly

Stack analysis: analyze lower bound and upper bound for worst-case stack depth

Accelerometer Tennis and Position estimation: use ADC to convert voltages in the accelerometer to implement a tennis game and position estimation system

Undergraduate Courses

Basic Memory Management

Allocate a huge memory space from operating system and then manage this memory pool for memory allocation and deallocation operations for test applications. We also performed boundary and checksum check for reads and writes.

UNIX File System

Used a file as a virtual disk and designed the layout of this virtual disk according to the traditional UNIX file system. Implemented functions to change/create/delete directories and create/delete/read/write files. Only 3 out of more than 120 students implemented a file system during that semester.

AVL Search Tree with Doubly Linked List

Implemented an AVL balanced search tree in C, which requires good handling of pointers, to keep the tree balanced when adding or deleting nodes

- PUBLICATIONS Xing Lin, Rajeev Balasubramonian, “Refining the Utility Metric for Utility-Based Cache Partitioning.”, *The 9th Annual Workshop on Duplicating, Deconstructing, and Debunking(WDDD 2011)*, held in conjunction with the 38th International Symposium on Computer Architecture (ISCA-38). San Jose, CA, June 5, 2011.
- Chunxia Zhang,Xing Lin(first student author),Shuguang Lin,Xudong Li, “Study of Handles Mechanism in WRK.” *The 2nd International Workshop on Education Technology and Computer Science (ETCS 2010)*. Wuhan, China. 6-7 March, 2010.
- Xudong Lee, Chunxia Zhang, Xing Lin(first student author), Shuguang Lin, “Introduce Satisfy policies into Semaphore in WRK.”, *International Conference on Computational Intelligence and Software Engineering (CiSE 2009)*. Wuhan, China. December, 2009.
- Xudong Li,Chunxia Zhang,Xing Lin(first student author),Shuguang Lin, “VAD: A Detail Investigation into Process’s Memory.” *2009 International Conference on Computational Intelligence and Security, (CIS 2009)*. Beijing, China. 11-14 Dec, 2009.
- POSTERS Raghuvveer Pullakandam, Xing Lin, Mike Hibler, Eric Eide, and Robert Ricci, “High-Performance Disk Imaging With Deduplicated Storage” *23rd ACM Symposium on Operating Systems Principles(SOSP ’11)*, Cascais, Portugal, Oct. 23, 2011.
- Xing Lin, Robert Ricci, Eric Eide, “A Survey of Computing Migration” *School of Computing Poster Competition*, Salt Lake City, Utah, U.S., March, 2010.
- Xing Lin, Robert Ricci, Eric Eide, “OpenVZ Migration in Emulab” *School of Computing Poster Competition*, Salt Lake City, Utah, U.S., March, 2011.
- HONOURS AND AWARDS Student grants from OSDI ’10, FAST ’11, ISCA ’11, Sep. 2010, Jan. 2011 and June 2011
National Encouragement Scholarship for excellent students, Nankai University, 2007-2008
Three Goods Student of Nankai University, Nankai University, 2006-2009
Second-level Scholarship of Nankai University, Nankai University, 2006
IBM Certificate of On Demand Business - Solution Designer V2, Nankai University, 2007
- COMPUTER EXPERIENCE Languages: C, C++, Java, Perl, MPI, OpenMP, shell, Tck, L^AT_EX 2_ε, SQL, .NET
Platforms: Linux, Windows Research Kernel, Emulab, LVM, OpenVZ