Statement of Purpose

MS/PhD in Computer Sciences, Systems

The closeness of disparate areas in computer science is intriguing. I believe we should be cognizant of all these areas as important conclusions have been reached by applying the knowledge of one area to another. Hence, the area of interest is not what one wishes to learn about, but one, one wishes to enrich. I wish to enrich the area of Systems specifically operating systems and file systems. I wish to apply my graduate education to pursue my interest in teaching and systems research.

While my projects in college strengthened my fundamentals and sculpted my thought, my work at Oracle Corp. exposed me to the practical problems and resolution strategies for multi terabyte databases. Now at IBM as a developer on the General Parallel File System, I work within the strains of high performance, scalability and availability, imposed by large-scale systems. These experiences have left me with an insight and equipped me for advanced research work and yet, paradoxically, revealed the limits of my knowledge and driven me to seek graduate education.

In college amongst other systems projects I did, two, "Win2Lin: File System calls across FAT32 and ext" and "Efficient trigger implementations in MYSQL 4.0" stood out for the enhanced learning experience they provided. The former performed in the third year, exposed basic concepts and issues in file system implementations, while the latter, our final year dissertation project, dealt with deeper systems concepts like efficient data structures for metadata, locking, security and deadlock handling with an emphasis on response time and performance. This was a ground for experimentation; we worked out the detection of mutating triggers by modeling them using control flow graphs. Performance tuning included hard parsing trigger code for compiled code executions, pinning selective metadata to server memory and replacing memory hungry structures with reduced bitmap implementations. After handling the-building-blocks kind of problems in college, my work scaled to robust distributed systems that serve terabytes of data.

At Oracle Corp., I was confronted with live, large-scale production systems; specifically, a multi terabyte multiple database client. Here, I developed cloning and concurrent task execution systems for them. Additionally, I also dealt with designing solutions to live problems that arose regularly. The vast amount of data led to problems in archival, fragmentation and data corruption for disaster recovery instances. I developed a batch defragmentation solution for a 1.7 terabyte database. I also developed a self adaptive system[1] that would automatically predict and fix corruption issues for disaster recovery instances. On joining IBM I transitioned from working with live problems, to implementing a full-blown product, fulfilling design goals and predicting likely problems beforehand.

I now develop the General Parallel File System for the Solaris platform. I have designed and implemented the mount techniques, VFS layer operations, distributed statistics collection and kernel memory structures. Thinking Distributed is a superset of thinking Systems. My thoughts grew from response times, deadlocks and starvation to high-performance, scaling and fault tolerance. A feature

[1] Awarded by Oracle Corporation for being a significant innovation for reducing recovery times and saving several man hours.

as basic as the mount operation was designed to be fail-safe and consistent across cluster nodes. The VFS layer operations balanced high capacity disk management while giving a high-performance. I also designed and implemented the tracing mechanism. The perceived "unity amongst nodes" did not restrict itself to resource and load balancing but also to error detection and recovery. Traces were generated across multiple nodes, to ensure debugging information for a crashed node was available on other nodes. Tracing in interrupt contexts, ensuring concurrent reading and writing of trace buffers, handling memory alignment and ordering issues and frequent kernel panics made the implementation challenging and non-trivial.

My experience in learning and enhancing GPFS has been educating and enriching. In large scale parallel systems trivial overheads grow exponentially, as do savings. This principle is the heart of each solution developed. Problems are not isolated from their applications. Trade-offs are closely weighed, different inputs mean differential treatments. The problem of concurrency for example, is solved with different locking protocols for data and metadata and the complexity of the synchronization methodology used is directly proportional to the size of the update. Having said this, I have developed an interest in and wish to pursue work in developing and applying ranking algorithms to the semantic cache methodology. I'm also interested in the undergoing standards development for parallel I/O, issues accompanying pedabyte storage and algorithms that leverage metadata for efficiency.

The knowledge gained by working through systems concepts in large scale databases and intricacies of designing scalable distributed file system has given me a foothold in these areas. The industry environment however restricts my aspirations of advanced learning with product boundaries and deadlines. After a detailed study of the graduate school brochure, I'm confident that the computer science department will give me the opportunity to explore advanced concepts and prototype existing ideas in my area of interest. I seek this opportunity and sincerely hope to be given a chance to perform and utilize my potential at your university.