Suresh Iyer

158 National St
Santa Cruz, CA 95060.
Tel : 1-831-331-1204.

Jack Baskin School of Engineering,
1156 High St., Santa Cruz, CA 95064.
suresh@cs.ucsc.edu

Education

- Ph.D (ongoing) Computer Science from September 2006, University of California, Santa Cruz (UCSC), California USA
- Master of Science in Computer Science, University of California, Santa Cruz (UCSC), California USA, June 2008
- B.Tech. Indian Institute of Technology(IIT), Kharagpur, India 1997

Highlights

- Strong storage industry experience in Storage resource management and other data center automation softwares.
- Good knowledge of SAN (storage area network), Fibre Channel internals, FCP, SCSI, Clustering technologies, Volume Management, File Systems, Operating systems.
- Over 9 years of work experience in programming in C/C++.

Interests

File systems, Operating systems, Storage systems , Object based Storage, Real time systems.

Graduate Courses

Advanced Operating Systems, Advanced Programming Languages, Storage Systems, Distributed Systems, Computer Networking, Analysis of Algorithms.

Research projects

- Looked at improving the read throughput for ceph, a petabyte scale distributed object file system which is under development at UCSC. Implemented read balancing and exclusive caching for OSDs (Object storage devices). Implemented flash crowd handling for OSDs by allowing reads from replica caches, and handling write consistency during replica reads.
- Implemented the RBED (Resource-Based Earliest Deadline scheduling) real time scheduling algorithm on the L4 micro kernel. My current research includes looking at the separation of user and kernel space for scheduling in L4. Using L4 IPCs, I am looking into the possibilities of a user space dispatcher with minimal kernel support.

Work experience

- Member-Technical Staff(Intern) at Network Appliance(NetApp) , Sunnyvale , CA, USA (06/2008-09/2008)
- **Senior Software Engineer** at R&D center, Symantec Previously known as VERITAS Software India Limited, Pune, India (06/2000-08/2006).
- Senior Software Engineer, Infosys technologies Limited, Pune India (04/1998-05/2000).
- Software Engineer, CMC Limited, Pune India (05/1997-04/1998).

Projects at NetApp

Worked on the NetApp Write Anywhere File System(WAFL) . Removed the resize phase for the client operations to increase the manageability and performance. This included modifying the core client operations for cases like large deletes and a good understanding of parts of WAFL code.

Projects at Symantec/ VERITAS

Product Group: SAN Access Layer (SAL)
C/C++ , cross platform (Solaris, Redhat Linux , AIX, HP, Windows)
Responsibility: Overall design as well as design and implementation of some modules, right from the product inception stage

SAN Access Layer (SAL) is the back end layer of the VERITAS Storage area network management tool namely VERITAS CC Storage. SAL implements discovery, automatic visualization of the SAN topology as well as management of the SAN including zoning, LUN Masking. SAL exports an object model of devices through an XML interface to a client such as the CC Storage GUI . SAL is composed of various device discovery modules called agents and supports an ever evolving huge hardware compatibility matrix.

- Designed and developed the following modules, apart from contributing to the design of the product infrastructure.
  - The **HBA(Host bus adapter) agent** discovers the devices and storage on the host and correlates to the devices on the SAN. The HBA agent uses interfaces of various vendor device drivers as well as the SNIA HBAAPI to discover objects such as the HBA, the FC ports on the HBA and the device paths and the LUNs visible through the HBA. These agents run on various hosts and send the discovery data to a primary database which is run on a Raima database engine. I contributed to the design of HBA agent as well as the implementation of the same on AIX and Linux.
    The Linux HBA agent implementation consisted of discovering devices using the OS interfaces by scanning the */dev* directory for devices and performing ioctls on to them to get the controller,target,LUN information. The SNIA HBAAPI was used to discover the devices and the attributes from both discoveries are merged on the objects. For AIX, the OS specific command line interface was used to discover the devices, along with SNIA HBAAPI.
    I also added the IDE/ATA disk discovery capability to the HBA agent.
  - The **GS3 agent** discovers the topology and manages the SAN using the Fibre channel common transport protocol. I was fully responsible for the design and implementation of the the GS3 agent. The GS3 agent sends in band CT (common transport) commands to the switch. The commands are sent using the pass-through interface of the HBA driver or the SNIA HBAAPI pass-through interface. The GS3 agent discovers the FC(Fibre channel) fabric, various FC switches, FC nodes and ports. The objects are correlated to other objects discovered by other agents using unique naming schemes. In additon, the GS3 agent also supports a management interface to perform zoning on the fabric.
  - The **Tape agent** discovers tape devices, tape enclosure devices behind an FC-SCSI bridge. Using Netbackup API library, this module discovers the tape enclosure device. In addition, the tape drives of the enclosure are discovered and correlated to the HBA agent tape devices.
- In addition, I also worked on the SAL core agent architecture. Core agent handles the client interface and inter host communication.
- I worked on customer escalation issues for CC storage and solved quite a lot of visible customer issues for SAL.
- To uniquely identify different LUNs visible from different hosts and also to correlate the LUNs which are visible through array specific discovery agents, **I devised a unique way of naming LUNs the SAN**. This is being developed as a utility library called the VERITAS device identifier(VDID). With this, various HBA agents and array specific discovery agents create only one object for a LUN object. For HBA agents, SCSI inquiry data was used to construct this. For example, for an EMC clarrion LUN, I used the SCSI inquiry page 83 UUID as the unique id for a VDID.
- **Published an Invention disclosure titled** *A method for dynamic LUN configuration in a SAN environment for Linux (2.4 kernel) hosts.* As a part of the design of SAN access layer, I was developing a framework to create device paths on a host, related to management operations like Zoning and LUN masking. This was dependent on the host operating system's ability to create device paths without reboot. However Redhat Advanced Server

2.1/3.0 (Linux 2.4 kernel) required a reboot for creating new device paths. My invention solved the problem of having to require a reboot of the host. The idea involved polling of the SNIA HBAAPI target mapping information to find out the new targets which are visible to the host and using the /proc interface of Linux to add a new controller,target,LUN triplet to the OS.

- Added high availability support to SAN access layer by integrating SAN access layer with VERITAS Cluster Server(VCS). This involved identifying the resources on which SAL access layer depended upon. I also added code in SAL to discover VERITAS cluster server installations in hosts and their storage dependencies.

PROJECTS AT INFOSYS CMC

- DHL offline scanners using a proprietary protocol called ACK/NACK. (Platform: C/C++ on HP-UX)
- Design/Development of a windows RAS client, which would dial up and ftp files to a remote server. (Platform: C/C++ win 32 SDK, Windows NT.)
- Worked on Hydraulic Analysis and Simulation Module which does mathematical modeling of a water supply network to provide monitoring and management of the network (Platform: C/C++, win32 on windows NT)

REFERENCE

Available upon request.