

Anshu Veda

anshu@it.iitb.ac.in

Telephone: +91-22-9323648022

M.Tech Student

KReSIT, IIT Bombay.

Mumbai -400076

Summary

Computer science graduate; M.Tech from IIT Bombay; 1 year experience in J2EE development at TCS, Mumbai; also proficient in Java, C, C++.

Education

Master of Technology (IT) - IIT Bombay [July 2004-June 2006]

With major subjects including Advanced Algorithms and Complexity, Object Oriented Systems, Distributed Algorithms, Performance Evaluation, Datamining, Mobile Computing. *CPI-9.20*

Bachelor of Engineering (Computer Science) - SVITS Indore [July 1999-July 2003]

BE (honors) with major subjects like Operating Systems, Discrete Structures and Networks. *Aggregate-78.91%*

High School - Sri Sathya Sai Vidya Vihar, Indore [April '1999, CBSE Board]

With Mathematics, Science and Economics as major subjects. *Aggregate-81.2%*

Secondary School - Sri Sathya Sai Vidya Vihar, Indore [April '1997, CBSE Board]

Aggregate-79.2%

Work Experience

Teaching Assistant at IIT Bombay [July 2004 - December 2005]

During my M.Tech, I was teaching assistant for the Affordable Solutions Lab, KReSIT where I evaluated a number of projects for Red Hat Scholarship and for Mobile Computing Course where I guided a number of course projects involving J2ME-Bluetooth-tomcat programming at IIT Bombay.

Software Engineer at Tata Consultancy Services, Mumbai [September 2003-April 2004]

I have worked for eight months with TCS, Mumbai for duration of 8 months. Here I worked over 2 major projects, E-quiz – A tool for handling creation, conduction, and evaluation of different subjective and objective quizzes in various domains for different level of hierarchy in the organization and Six Sigma Process Improvement tool. I was a part of the team that worked over design and implementation of several modules. The project involved JSP/Java/backend programming with Tomcat at the server end and Oracle as the backend.

Master's Project [In progress]

Automatic Partitioning of Java programs

[Details: http://www.it.iitb.ac.in/~anshu/wiki/doku.php?id=wiki:mtech_project:mtech_project]

Partitioning implies splitting up an application into components (possibly interacting) that run on different network locations, while maintaining consistency with the original program which was written for a standalone system. Our motive is to separate distribution concerns

from the application development. Writing program considering distribution aspects beforehand increases the complexity of program to a great extent. Modifying existing sources to enable execution in a distributed environment involves a lot of re-coding. In addition, we consider the scenario where the deployment settings can vary time and again. Given an existing Java code, designed, coded, debugged and tested for executing on a standalone computer we want to partition it in a transparent fashion. This is desirable for usual benefits of distributed systems like those of efficient resource utilization and reduced network traffic. Automation of such process will reduce the burden of developers to a great extent I have been working under the guidance of Professor Sridhar Iyer on this problem.

Course Projects [During the M.Tech]

A Virtual Classroom [Systems Laboratory Course]

This is an ongoing project at IITB being sponsored by *Intel*. This project aims at putting together an integrated e-Learning environment for a university student. This will mainly be used by distance education programs offered by institutes, in order to enable students to avail of the academic facilities from any computer connected to the internet. As a part of first phase team for this project we were involved in the design and set up of the server side architecture as well as server configuration which involved setting up JBoss4.0.1 and Postgres8.1 configuration. We coded a number of modules. We used server side EJBs and integrated it with Java -Swing client.

XML Diff and Merge tool [Foundation Laboratory Course]

This project involved the development of a tool that could generate and display the differences between two xml files and if needed merge them via a web-interface. It identifies the differences to a specified nesting level. The tool was built using java, XML, X-path APIs and JSP with tomcat server.

Implementation of Distributed Phase Commit [Implementation of Database Management Techniques]

Here we implemented a multi-threaded version of 1-2 Phase commit algorithms to enable consistent commit in Distributed Database scenario in C.

Solving Feature Interaction Problems using Java Filter Objects [Object Oriented Systems]

Feature Interaction problems refer to scenarios that arise when support of one feature in a system conflicts with other features. We designed solutions to some such problems and implemented the same in translator for Java filters (TJF) - an extension to java that supports filter objects, developed at IIT, Bombay (<http://www.cse.iitb.ac.in/~rkj/filter.html>).

Intrusion Detection System [Datamining Course]

The goal was to come up with an anomaly-detection based IDS. We tried to incorporate both network-based and host-based detection techniques using different clustering (K-Means) and classification (KNN, Decision trees) approaches to suggest an appropriate algorithm for IDS. Our system was tested on IITB network logs.

Simulation of Distributed Fair Queue Scheduling in WLANs [Performance Evaluation Course]

Here we simulated the execution of a published scheduling algorithm for 802.11b WLAN scenario in Java. This gave us a feel of different aspects of simulation including the maintenance of virtual clock, simulating queues and analyzing the system for parameters like throughput, fairness etc, in addition to the understanding of how simulating a scenario is different from actual implementation.

Seminars and Presentations [During the M.Tech]

Localization and Root Cause Isolation of Faults in Large-scale systems [Seminar M.Tech Course]

Localization involves mapping of faults to components, given some logs say for example the set of requests failed, whereas root cause isolation involves mainly dependency analysis over the set of failed components to determine the actual faulty component. Here we surveyed all the available techniques in literature for dealing with this problem which involved – using clustering, decision trees and some other datamining techniques for localization as well graph algorithms for dependency analysis.

Search Mechanisms for *Peer-to-Peer* Systems [Distributed Algorithms Course]

Searching and information retrieval is an important problem in peer to peer systems due to their *decentralized* nature. We wrote a term-paper comparing the existing techniques-variations of BFS (e.g. iterative deepening), dynamic search, and multi-agent system based algorithms.

Data Aggregation, Query Processing and Routing in Sensor Networks [Mobile Computing Course]

We wrote a term paper surveying the available protocols for data aggregation, query-processing and routing, their usability and tradeoffs.

B.E. Project [July 2002-March 2003]

Framework for Bayesian Learning and Inference

Bayesian Belief networks are graphical models used as an important classification technique in datamining algorithms. Given a large dataset representing the problem domain, we tried to implement an efficient learning algorithm to learn the Bayesian belief network representing the dependencies between the system variables and their probability distributions. This learnt graph could then subsequently used by the *inference* engine where the effect on distribution of other variables could be calculated given some evidence. Such a framework was targeted for problems like those of *Market-Basket Analysis*.

Skills and Interests

Languages

C/C++, Java, VB.

Scripting languages

Bash scripting, Perl, PHP, CGI.

Other tools and technologies

J2ME, JSP, EJB, ns-2, OPNET, Qualnet.

Platforms

Linux, Windows, J2EE

Interests

Distributed Systems, Object Oriented Systems, Wireless Networks, Datamining and AI.

References

Available on request