CV of Srinivasan Kannan (alias) Ka.Shrinivaasan (alias) Shrinivas Kannan

Permanent Address:

Srinivasan Kannan,

S/O. P.R.ES.Kannan,

172, Gandhi Adigal Salai,

Kumbakonam–612001, TamilNadu, India.

e-mail :   ka.shrinivaasan@gmail.com

shrinivas.kannan@gmail.com

kashrinivaasan@live.com

Mobile :   9789346927, 9791165980,

9003082186

Name spellings in

employer/academic

records :   Srinivasan Kannan (academics, BaaN,Sun

Microsystems,Verizon),Shrinivas Kannan(webMethods and

CMI), Ka.Shrinivaasan (Global Analytics)

Personal website(research) :- https://sites.google.com/site/kuja27/

Krishna iResearch Open Source: https://sourceforge.net/users/ka_shrinivaasan,

https://www.ohloh.net/accounts/ka_shrinivaasan

Date of Birth as per records: 19 May 1977

Marital Status: Unmarried, Single

Passport Number: A9373811 (expired in 2011, pending passport renewal application

file number – TR1071503012712)

COMPLETE ACADEMIC AND WORK HISTORY

ACADEMICS

1. Elementary Schooling, RC Morning Star, Kumbakonam (1982-1987)

2. Higher Secondary, Town Higher Secondary School,Kumbakonam (1987-1994) (SSLC -470/500 and Plus Two 1115/1200)

3. Bachelor of Engineering (Computer Science & Engg) (1995-99), PSG College of Technology, Coimbatore-641004 INDIA (CGPA/Percentage: 8.8/87.75)

4. M.Sc (Computer Science), Chennai Mathematical Institute, Chennai (2008 – 10) CGPA 8.08

5. Ph.D (Junior Research Fellow in Computer Science), Chennai Mathematical Institute, Chennai (August 2010 – August 2011) – resigned

---

COURSES DONE IN M.Sc and Ph.D and Self–study

M.Sc courses [CMI and IIT-Chennai] and self-study (2008-2010):

Haskell, Operating systems, Distributed systems, Theory of computation, Databases,Logic-1, Complexity-1, Principles of Programming Languages(Java and lambda calculus), Algorithms, Graph theory, Cryptography, Datamining-1, Information Retrieval, AutomataConcurrencyTimedSystems

Ph.D courses[CMI and IMSc] and self study (August 2010 - October 2011):

Complexity-2, Topics in data mining(Recommender Systems,Streaming Algorithms), Randomized algorithms(including PTAS), Logspace computation, Program Verification, Program Analysis, Program Slicing, Computational number theory and algebra, Computational geometry, Expander graphs, Combinatorics(Generating functions), Probabilistic method, Communications Complexity, Linear Programming And Combinatorial Optimization and Computational Biology (BioInformatics) algorithms for sequence alignment.

PUBLICATIONS and PUBLICATION DRAFTS

1.(During PhD) **Decidability of Existence and Construction of a Complement of a function with Prof.Meena Mahajan, IMSc, 2011**
(http://arxiv.org/abs/1106.4102)

2.(Master's Thesis) **Few Algorithms for Ascertaining Merit of a document (Citation graph Maxflow, Recursive Gloss Overlap Algorithm and Interview algorithm applying either of the previous two with applications of them) with Profs. Ravindran(IIT Chennai) and Madhavan Mukund (CMI) in 2009-2010 –** http://arxiv.org/abs/1006.4458.

3. (During PhD) **Evaluated NIST TAC 2010 dataset (Summarization track) for Update Summarization by applying Interview Algorithm – appeared in proceedings of TAC 2010** at:
http://www.nist.gov/tac/publications/2010/participant.papers/CMI_IIT.proceedings.pdf

4.**IntegerPartitions and Hash Functions (insufficiently reviewed)**
(https://sites.google.com/site/kuja27/IntegerPartitionAndHashFunctions.pdf?attredirects=0)

5.**Interview Algorithm is in IP=PSPACE (insufficiently reviewed)**
(https://sites.google.com/site/kuja27/InterviewAlgorithmInPSPACE.pdf?attredirects=0)

6.**Few Non-trivial questions and Shell Turing Machines (insufficiently reviewed)**
(https://sites.google.com/site/kuja27/UndecidabilityOfFewNonTrivialQuestions.pdf?attredirects=0)

7. **Research Statement** - https://sites.google.com/site/kuja27/ResearchStatement2.pdf?attredirects=0

8. **Research Proposal with some proof sketches (includes a new timeout manager algorithm)**

- https://sites.google.com/site/kuja27/PhDThesisProposal.pdf?attredirects=0

Following are publication drafts (not final versions) – in progress:

9. **(Draft) Lower Bounds for Majority Voting and Pseudorandom choice –**

https://sites.google.com/site/kuja27/LowerBoundsForMajorityVotingPseudorandomChoice.pdf

10. **(Draft) Circuits for computation of error probability in majority voting –**

https://sites.google.com/site/kuja27/CircuitForComputingErrorProbabilityOfMajorityVoting.pdf

11. **(Draft) Indepth analysis of a variant of Majority Voting with relation to ZFC -**

https://sites.google.com/site/kuja27/IndepthAnalysisOfVariantOfMajorityVotingwithZFAOC.pdf
?attredirects=0

12.(Draft) A Chaos theoretic Parallel Pseudorandom generator in RNC For Majority Voting
and Pseudorandom Choice (https://sites.google.com/site/kuja27/ChaoticPRG.pdf?
attredirects=0)

13. (Draft) Analysis of a Randomized Space Filling Algorithm and its Linear Program
Formulation (https://sites.google.com/site/kuja27/Analysis%20of%20a%20Randomized
%20Space%20Filling%20Algorithm%20and%20its%20Linear%20Program
%20Formulation.pdf?attredirects=0)

14. **(Draft) Discrete Hyperbolic Polylogarithmic Sieve For Integer Factorization (version 1) -**

https://sites.google.com/site/kuja27/DiscreteHyperbolicPolylogarithmicSieveForIntegerFactorization.pdf?attredirects=0&d=1

15. (Draft) Discrete Hyperbolic Polylogarithmic Sieve For Integer Factorization – with Interpolation Search (version 2) –

https://sites.google.com/site/kuja27/DiscreteHyperbolicPolylogarithmicSieveForIntegerFactorization_updated_interpolation_search.pdf?attredirects=0&d=1

16. (Draft) Discrete Hyperbolic Polylogarithmic Sieve For Integer Factorization – with Interpolation Search (version 3) –

https://sites.google.com/site/kuja27/DiscreteHyperbolicPolylogarithmicSieveForIntegerFactorization_updated_interpolation_search_30June2013.pdf?attredirects=0&d=1

17. (Draft) Discrete Hyperbolic Polylogarithmic Sieve For Integer Factorization – with Interpolation Search (version 4 and version 5 with handwritten illustrations and calculations) –

http://sourceforge.net/projects/acadpdrafts/files/DiscreteHyperbolicPolylogarithmicSieveForIntegerFactorization_updated_interpolation_search.pdf/download

18. (Draft) Informal notes on Implication Graphs, Error probability of Majority Voting and P Versus NP Question
http://sourceforge.net/projects/acadpdrafts/files/ImplicationGraphsPGoodEquationAndPNotEqualToNPQuestion_excerpts.pdf/download

19. (Draft) <u>Discrete Hyperbolic Polylogarithmic Sieve For Integer Factorization – using Rectangular Binary (or) Interpolation Search</u>

http://sourceforge.net/projects/acadpdrafts/files/DiscreteHyperbolicPolylogarithmicSieveForIntegerFactorization_updated_rectangular_interpolation_search.pdf/download

20. (Draft) <u>Informal Notes on Derivation of Upperbound for Discrete Hyperbolic Factorization with Stirling Formula - using Rectangular Binary or Interpolation Search</u> (http://sourceforge.net/projects/acadpdrafts/files/DiscreteHyperbolicFactorization_Upperbound DerivedWithStirlingFormula_2013-09-10.pdf/download)

21. (Draft) Discrete Hyperbolic Polylogarithmic Sieve For Integer Factorization – using Rectangular Binary (or) Interpolation Search and Upperbound derived with Stirling Formula http://sourceforge.net/projects/acadpdrafts/files/DiscreteHyperbolicPolylogarithmicSieveForIntegerFactorization_updated_rectangular_interpolation_search_and_StirlingFormula_Upperbound.pdf/download
(a C++ implementation of this algorithm is at:
http://sourceforge.net/p/asfer/code/HEAD/tree/cpp-src/miscellaneous/DiscreteHyperbolicFactorizationUpperbound.cpp and factorization result logs at: http://sourceforge.net/p/asfer/code/HEAD/tree/cpp-src/miscellaneous/)

22. Individual Invention Disclosure for Survival Index Based Transaction Timeout Manager (Sun Microsystems) -
https://sites.google.com/site/kuja27/SurvivalIndexBasedTxnTimeoutManager.pdf?attredirects=0

23. AstroInfer Open Source Product Design that includes an experimental non–statistical inference model - http://sourceforge.net/p/asfer/code/HEAD/tree/AstroInferDesign.txt

24. VIRGO Open Source Product Design that includes an experimental Pseudorandom Number Generator based Loadbalancer – http://sourceforge.net/p/virgo-linux/code-0/HEAD/tree/trunk/virgo-docs/VirgoDesign.txt

25. King Cobra – Byzantine and CAP based request servicing design – https://sourceforge.net/p/kcobra/code-svn/KingCobraDesignNotes.txt

26. An NC algorithm and some Sequential Search Algorithms for Discrete Hyperbolic Polylogarithmic Sieve For Factorization using Binary or Interpolation Search with Stirling Formula and Logarithmic Sorted Tile Merge in PRAM model – http://sourceforge.net/projects/acadpdrafts/files/DiscreteHyperbolicPolylogarithmicSieveForIntegerFactorization_PRAM_TileMergeAndSearch_And_Stirling_Upperbound.pdf/download

---

TEAM PATENTS

---

1. Simultaneous global transaction and local transaction management ...

US Pat. 7610305 - Filed 24 Apr 2003 - Issued 27 Oct 2009 - Sun Microsystems, Inc.

2. Read/write lock transaction manager freezing

US Pat. 7739252 - Filed 14 Jul 2003 - Issued 15 Jun 2010 - Oracle America, Inc.

3. Utility for configuring and verifying data sources

US Pat. 7134008 - Filed 4 Sep 2003 - Issued 7 Nov 2006 - Sun Microsystems, Inc.

4. Transaction optimization of read-only data sources

US Pat. 7165061 - Filed 31 Jan 2003 - Issued 16 Jan 2007 - Sun Microsystems, Inc.

5. Common transaction manager interface for local and global transactions

US Pat. 7743083 - Filed 24 Apr 2003 - Issued 22 Jun 2010 - Oracle America, Inc.

6. Specifying transaction manager type at various application levels

---

US Pat. 7082432 - Filed 24 Apr 2003 - Issued 25 Jul 2006 - Sun Microsystems, Inc.

7.Transaction manager freezing

US Pat. 7640545 - Filed 14 Jul 2003 - Issued 29 Dec 2009 - Sun Microsytems, Inc.

8.Identity for data sources

US Pat. App 10655346 - Filed 4 Sep 2003 – Sun Microsystems, Inc.

9.Common transaction manager interface

US Pat. App 10422453 - Filed 24 Apr 2003 – Sun Microsystems, Inc.

DETAILS OF WORK (1999 – Present)

C/C++/Java/Python on Windows and various Unix flavours

PRODUCTS

Netscape Application Server 4.0, iPlanet/SunONE Application Server 6.x, SunONE Web/Proxy Server 3.6/4.0, Apache web server 1.4.x/2.0.x, webMethods Broker Messaging Server 5.x/6.x/7.x, ASFER Classification and Inference Software for Large Data Sets (10.0,12.0), USBmd 1.0, VIRGO Linux Kernel Extensions for Cloud (5.0,6.0,7.0,8.0,10.0,12.0), Global Decision Platform(GDP) 2.3.1/ 2.5.0/ 2.5.1/ 2.7 /2.7.1/3.0, Automated Modelling Platform(AMP), Python 2.4.3 and 2.7 source code, Linux kernel [3.2.0, 3.7.8, 3.9.x], Maitreya's Dreams 7.0.3, STL, BOOST, wxWidgets library, Swiss Ephemeris API (based on NASA JPL DE406 Ephemeris).

LANGUAGES/PLATFORMS/TOOLS

C/C++, Java, J2EE, Python, Haskell, Solaris 6-10, Windows 2000, HP-UX 11.23 (PA-RISC and IA64),AIX, Linux, Sun Studio, CORBA (Visibroker), VAX FORTRAN, VAX VMS, GNU collection, MS Visual Studio .NET C++, WinDbg, DDD, Visual Basic and ASP, OptimizIt, Eclipse, EclipseCDT, Code::Blocks, ActiveState Python, IDLE, Haskell FP, Insight, GreatCircle, Valgrind, Callgrind, Cloud Computing Mapreduce Tools (Dumbo Mapper-

reducer on Hadoop), Python NLTK, Hadoop, PiCloud, RabbitMQ, Pyutilib workflow, PyF workflow, Linux kernel – [3.2.0, 3.7.8, 3.9.x], Maitreya's Dreams 7.0.3, wxWidgets library, Swiss Ephemeris API (based on NASA JPL DE406 Ephemeris), BioInformatics Sequence Alignment Tools (BioPython, ClustalOmega), ROOT(CERN), Geolocation Python API, Beautiful Soup

---

January 2013 – present – Krishna iResearch (self-started, not-for-profit, open-source research initiative) (https://sourceforge.net/users/ka_shrinivaasan)

Working on Research, Design and Development of open source projects started by self licensed under GPL v3:

1. acadpdrafts - All publications and drafts, important documents and Photo ID proofs are uploaded at http://sourceforge.net/projects/acadpdrafts/

2. (Experimental) iCloud - Linux kernel USB device driver module for debugging USB issues possibly using machine learning algorithms for fault detection (USBmd) USBmd – (Latest Code at sourceforge repository: https://sourceforge.net/p/usb-md/ and periodically updated Design Details at: http://sourceforge.net/p/usb-md/code-0/HEAD/tree/USBmd_notes.txt). Presently USBmd version 1.0 has been released and future version is in development. Part of iCloud platform.

3. iCloud - Asfer - AstroInfer astronomy and astrology machine learning inference open source software which uses algorithms viz.,

> **2.1   Bayesian Classifier,**
>
> **2.2   Support Vector Machines,**
>
> **2.3   Decision Tree Classifier,**

2.4   Pairwise String Sequence Alignment (Needleman–Wunsch) ,

2.5   Multiple String Sequence Alignment(BioPython,ClustalOmega),

2.6   Powerset construction

2.7   Parsers and Encoding features for USGS and NOAA HURDAT2 data

2.8   Automated Generation of Training and Test set for Classifiers that
segregate the datasets into Event Classes and Automated generation of encoded
astronomical object files specific to each Event Class.

2.9   Sequential implementation of Discrete Hyperbolic Factorization and
PRAM design for publication drafts in https://sites.google.com/site/kuja27/

Goal of AstroInfer is to develop an inference system for Big Data, especially, massive
encoded string and numeric datasets and to mine patterns in them. Presently implemented
for mining patterns in astronomical datasets(degrees of astronomical objects viz planets,
constellations etc.,) and prediction based on rules and execution of those rules, but works
for any dataset. Design started in May 2003. Presently ASFER version 12.0 has been
released. Part of iCloud platform.

(Latest Code at sourceforge repository: Asfer – http://asfer.sourceforge.net and periodically
updated Design Details at:

http://sourceforge.net/p/asfer/code/HEAD/tree/AstroInferDesign.txt)

4. iCloud – VIRGO – VIRtual Generic Os – Cloud Platform – Linux Kernel Extensions for
cloud – kernel modules, system calls implementing cloud at linux user thread creation –
clone() – step itself rather than at application level high up the stack with :

3.1   Config file support

3.2   Psuedorandom Generator Based Loadbalancer

3.3   Kernel space remote execution

3.4   User space remote execution with kernel upcall and pthread creation of

userspace library function or executable

     3.5   Example unit test cases

     3.6   Usermode output redirected logging feature for Kernel upcall to Userspace

     3.7   Intermodule Function Invocation in Kernel Space - through which any machine on cloud can be completely remote-controlled deep upto board and hardware cards through function names or commands sent through virgo_clone() calls.

     3.8   Multithreaded Kernel Service – Multi-kernel-threaded VIRGO cloudexec Kernel Driver Module for unrestricted service of virgo_clone or other client requests.

     3.9   Memory Pooling Driver (and a key-value store) and system call and userspace clients for it on Cloud nodes

Goal of VIRGO is thus to enable viewing entire cloud as a single "logical machine" upto hardware level and this differentiates from other cloud libraries which are in userspace mostly. Unification and Integration of AstroInfer and USBmd into VIRGO to make it a Cloud Platform with Machine Learning abilities, Implementation of a Cloud File System for VIRGO cloud platform, Integration of mobile operating systems in cloud are, among other things, long term goals for VIRGO. Part of iCloud platform.
(Latest Code at sourceforge repository: VIRGO - https://sourceforge.net/projects/virgo-linux/ and periodically updated Design details at: http://sourceforge.net/p/virgo-linux/code-O/HEAD/tree/trunk/virgo-docs/VirgoDesign.txt)
Design started in 2008. Presently VIRGO version 12.0 has been released.

5. King Cobra – Byzantine Distributed Request Servicing Software with Queueing on either iCloud platform or Hadoop Cluster - https://sourceforge.net/p/kcobra/

October 2013 – December 2013 – continued work on my opensource products (Krishna iResearch - https://sourceforge.net/users/ka_shrinivaasan) and non-profit machine learning consultancy to PiQube.com, Chennai (http://piqube.com)

September 2012 - February 2013 – Krishna iResearch - Consultancy for Global Analytics(Chennai) WOP

1. Did the **Python AMP workflow implementation** as per the workflow specification for AMP using PyUtilib, RabbitMQ and Config files support.

2. Wrote **AMP workflow-over-cloud** specification in addition to AMP workflow spec.

3. Guided on GDP bug fixes and issues

Apart from the above worked on design and development of my nonprofit open source products- https://sourceforge.net/users/ka_shrinivaasan

January 2011 – March 2011 and July 2011 – August 2012 – Global Analytics(Chennai) – Senior Lead Software Architect (C/C++/Python)

1. Mentored and Managed a team of 3 people .

2. Released version 2.3.1 of Global Decision Platform(GDP) with fixes for crashes in production environment.

3. Did refactoring, many bug fixes and enhancements to Global Decision Platform(GDP)

version 2.5.0 for **master demo lead passthroughs** to work.

4. Implemented **acceptor-worker thread model** request handling in GDP 2.5.0 with single acceptor thread and configurable number of worker threads for Service Execution Manager(SEM) and Service components of GDP. This resulted in significant improvement in response time.

5. Designed and implemented **session based request-response** in GDP 2.5.0

6. Implemented modified timeout error logging with placeholders for errorcodes and request id and did fixes for frequent crashes in production environment for GDP 2.5.0.

7. Released GDP version 2.5.1 for which 2 minor binaries were developed to clear the queue and test the creation of queues (for System V message queues)

8. Released GDP version 2.7 which has important bug fixes for crashes in python container(needed for AMP runtime described later), few features for **graceful shutdown** of the SEM, option to reconnect to MySQL during exception in Python container etc.,

9. Redesigned GDP 2.5.0 **SEM loadbalancer for GDP 3.0** to route requests to service processes by periodically monitoring the service process load and removing a bottleneck due to wait/notify code. This resulted in 10x speed up of the request-response throughput time.

10. For GDP version 3.0, designed and Implemented a **new Session Timeout Manager** algorithm to timeout the request session based on user configured timeout value(simplified version of Survival Index Based Transaction Timeout Manager mentioned later).

11. For GDP version 3.0, worked on embedding **multiple python interpreters** in Service process with Python C API replacing boost::python calls. This involves facility to have a global dictionary across multiple interpreters (or) local dictionary per request (or) per-thread dictionary for each service worker thread (using thread local storage) and also creation of pool of interpreters from which interpreters are allocated per request and returned to the pool after the request is serviced. This obviated the need for boost::python since boost did not support multiple interpreters with multithreading and interpreter pooling. This involved fixing one of the most time consuming bug related to restricted mode python in 2.4.3. Since it was deprecated in python 2.7, fixed Python 2.4.3 source to circumvent restricted mode and also did another fix for the same problem with PyImport_ImportModuleEx()

12. Mentored the team to work on Dynamic Risk Tables (DRT) library of GDP to extend the database compatibility to MySQL and Oracle

13. Was involved in the Automated Modelling Platform(AMP) development to automate the predictive modelling from the production data. Fixed many crashes in python runtime during testing of AMP prototype realtime design. Suggested an alternative design for AMP realtime based on external user address-space message queue, which populates the in-memory queue from on-disk request log, periodically replenishes it from disk, reads requests from message queue and posts them in parallel to Realtime-subset and Realtime-superset python services, and gives a virtual view of the single physical queue to each queue client(queue clients are located in SEM) – meant to replace an existing queueing module with msgsnd/msgrcv which involves user and kernel address-space copying to remedy huge queue backlog seen in AMP realtime prototype. This tightly couples GDP Python runtime with the AMP runtime

14. For GDP 3.0, implemented a GDP pluggable queue named **GDAMPQ** designed for AMP above – Global Decisioning and Automated Modeling Platform Queue which is a simple in-

memory queue with optional disk persistence to replace an existing queueing module with msgsnd/msgrcv which involves user and kernel address-space copying. This in-memory queue in best testing circumstances is 30% better than the System V unix message queues.

15. Worked on Python 2.4.3 source code to analyze crashes in PyMalloc. Modified Python 2.4.3 source to add a flag DISABLEPYMALLOC to disable calls to PyMalloc and re-route them to unix malloc/free systemcalls. Also analyzed Python 2.4.3 source code for **restricted mode related** errors while designing multi-interpreters for GDP 3.0 and fixed them.

16. Worked on writing python configuration scripts for **automating the build configuration and installation** of GDP components like SEM, Service and Master .

17. Implemented a minor **encrypt utility** for master component of the GDP

18. Wrote Functional specification for **GDP 3.0 Automated Installer design**

19. Wrote Functional specification for **AMP Workflow**

20. Wrote Functional specification for **AMP Cloud deployment**

21. Studied python workflow packages for AMP workflow viz., Pyutilib.workflow, PyF and implemented a workflow prototype for AMP workflow in Pyutilib.workflow.

22. Implemented a **small prototype stub using Dumbo Mapper–reducer** for Cloud
 Deployment of AMP

23. Did 2 one-week training sessions for GDP versions

24. Did one-week training session for Python.

25. Implemented a **mini-GDP client and server** for debugging socket related errors in production.

26. Developed a **documentation knowledgebase** intranet website for GDP documentation, design documents, installation troubleshooting documents and debugging documents.

27. Implemented a **Build Version feature** for GDP 2.7.1 and GDP 3.0 which dynamically generates a buildversion.h headerfile which contains #defines for build machine and architecture information,SVN source tree version information for the build and is built at runtime by SEM, Service and Master binaries which include this buildversion.h header file **(In August 2012 resigned for personal reasons.)**

January 2011 – March 2011 – Consultant for Global Analytics(Chennai) and Krishna iResearch – (C++/Python)

Worked on my Open Source Products Design & Development (Krishna iResearch) and Did consulting and development for Global Analytics,Chennai (optimization, a smart-pointer reference-counted memory manager backed-up by an Object Pool, and refactoring) in Global Decision Platform (GDP)which is written in C++ with boost::python embedding.

August 2008 – June 2011 – Krishna iResearch (Open Source Project – ASFER in C++ - not-for-profit open-source research initiative)

Started working on open source project ASFER (http://asfer.sourceforge.net) as part of my non-profit open source initiative Krishna iResearch – ASFER is a rule miner and executer-

presently uses Vector space retrieval and Support Vector Machines with added Support for Naive Bayesian Multinomial Classifier and Decision Tree Classifier

March 2006 – July 2008 - webMethods (now Software AG) (Bangalore) (C/C++/Java)

Worked on WebMethods Broker Server 5.x/6.x/7.x– a heavily multithreaded messaging productwritten in C/C++/Java based on publisher/subscriber model – Clients can publish and/or subscribe to certain predefined types of messages called "Events". My responsibilities included development of new functionalities and fixing issues in core areas of Broker (Connection Layer, Territories/Gateways,Broker Admin Tool etc.,) and writing knowledgebase documents.

November 2005 – February 2006 – Krishna iResearch (self-started, not-for-profit, open-source research initiative)

Initial design work for Krishna iResearch open source products focussing on algorithms for web and BigData

August 2005 – November 2005 – Verizon Data Services India, Chennai

In Verizon India for brief period and then self-study in Mathematics and Computer Science.

May 2003 – Krishna iResearch (self-started, not-for-profit, open-source research initiative)

Registered on SourceForge.net and Initial design work started for open source products focussing on algorithms for web and BigData

February 2000 – July 2005 - Sun Microsystems (now Oracle) (Bangalore)

Ported SunONE Application Server to Red Hat Enterprise Linux Advanced Server 2.1 (2005)

Certified JWSDP 1.3 webservices pack on Sun One Application Server 7.1 (2005)

Studied feasibility of supporting SunOne Web/Proxy Server on Solaris 10 zones (2005)

On Academic Sabbatical for 3 months (February 2005–May 2005)

Implementation of a threaded ICP server and porting SOCKS server for Sun Java system web proxy server 4.0 (C/C++/Solaris/Windows/RHELinux) (2004–2005):
Implemented a threaded ICP(Internet Cache protocol) server functionality using NSPR threads.
Also ported a legacy SOCKS server to proxy server 4.0.

Load Balancer Module for Apache Web Server 1.3.27 / 2.0.47 (C/C++) (2003–2004) :
Designed and implemented a Load balancer module for Apache 1.3.27 which will route the HTTP/HTTPS requests onto SunONE Application Server instances by Round Robin Algorithm.
This module is part of Sun ONE Application Server 7, Enterprise Edition.Also ported the module to Apache 2.0.47.

Optimizations and Features for iPlanet/Sun ONE Application Server 6.x (Java) (open sourced at http://glassfish.java.net) (2002) :
a) Fixed bugs related to performance, admin-tool and security in iPlanet Application Server 6.x. Also fixed few CTS bugs in SunONE Application Server 7 Standard Edition.
b) Involved in optimization of SUN J2EE Reference Implementation (RI) to make the JTS transaction manager enterprise-class. This included introducing a new Component Level Transaction Attribute setting feature in iPlanet Application Server 6.5. With this, user can set the type of transaction (XA or Local) at J2EE application component level. [Patents granted – details in the end]
c) Added a Trace functionality for the JTS transaction manager in SunONE Appserver 7.0 Enterprise Edition (http://glassfish.sourcearchive.com/documentation/1:2ur2-b04-1/dir_512e7fccb5ab465c594742cd72317a0e.html).

Debugging and Monitoring framework (MagicDraw) (2002) :
Designed a Debugging and Monitoring framework for Sun ONE Application Server 7.0 Enterprise Edition.

Store Adapter prototype for High Availability(Java) (2002) :
Implemented a store adapter prototype for Clustra High Availability Database. This was to test the feasibility of using Clustra HADB as backend store for session data in SunONE Application Server 7.0 Enterprise Edition.

Minor Feature Addition to SunOne Application Server 6.5 SP1 (2002)

Added support for enabling/disabling TCP_NODELAY in SunOne Application Server 6.5 SP1

Survival Index based Transaction Timeout Manager for iPlanet Application Server (Java) (2002):

Designed and implemented a new Transaction Timeout Manager for iPlanet Application Server.This resulted in an overall speed-up of 10% for the application server. (Invention Disclosure done in 2002-2003)

Miscellaneous Bugfixes for customer escalations in the BillerXpert product which was dependent on iPlanet Application Server. This was one of the most involved bugfixes that required many days of debugging with JVM Profiling tools. (2001)

An automated Test suite for nightly build setup of iPlanet Application Server 6.0 (2001):

Designed and implemented a test suite which ran a set of regression tests on nightly builds of iPlanet Application Server - written on Korn shell.

JVM related bugfix for IBM's Java Virtual Machine team related to native thread and green thread implementations of JVM (2000)

Created a Resume Querying System using JDBC with Oracle Backend for Sun Microsystems Human Resources Department to track candidate profiles – named ResumeXpert (2000)

Official Training Courses (SunU)
1. Solaris internals
2. Multithreaded Application development in Solaris

3. SOLARIS DEVICE DRIVER INTERNALS

July 1999 – January 2000 – BaaN Infosystems (now SSA Global) (Hyderabad)

Developed a storefront for an e-commerce application called

E-Enterprise using Microsoft Internet Information Server/Site Server, Visual Basic, Active

Server Pages and Visual Interdev

January 1999 – May 1999 – BE Thesis and Project(team) – COBRA – Distributed Computing Framework based on CORBA(Visibroker) and JAVA.

April 1998 – June 1998 – BE Internship at Steel Authority of India Limited (Raurkela and Durgapur Steel Plants) – Enhancements to Automated Furnace Data Acquisition software through a GUI developed on VAX Fortran over VAX VMS

MISCELLANEOUS

1. Gold medal for proficiency from PSG Tech for ranking first in B.E.(CSE) – Received in February 2000

2. Praveen Uttarardh (Hindi) - 1988 - 92

3. Joint Entrance Screening Test(JEST) 2006 in Theoretical Computer Science - Rank 21.

4. Interests include solving puzzles, writing, reading non-fiction mostly on science, mathematics, religion and philosophy.

5. Participated in Winter School 2010 on Machine Learning and Computer Vision, Microsoft Research & CIFAR, IISc