+91 97 31 81 42 97 GNVS Sudhakar novell.sudhakar@gmail.com

Summary: **Expert** in Systems Programming (**Linux Userspace & Windows Kernel**).
27 US patents on cryptography, security, distributed computing & virtualization.
Deep understanding & application of the **halting problem**.
1 personal patent (submitted provisional) on **Sanskrit** technologies in **2025**.

Education:

- **IISc, Bangalore** | 2025 | 1 week | Campus
  Graphs, Spectra and Convex Geometry
  (Designing scalable, structure-aware systems and optimization algorithms)
- **IISc, Bangalore** | 2025 | 1 year | Online
  Deep Learning Foundations and Applications
- **NITK, Surathkal** | 2001 | 3 year | Campus
  Master of Computer Applications
  (Elective in Compiler Design. Seminar paper in August 2000 titled "Ancient Sanskrit language and modern computers")
- **SV University, Tirupati** | 1998 | 3 year | College
  Bachelor of Science
  (Mathematicss, Physics, Chemistry)

Experience:

SPGAMT: Bangalore (September 2025 - present)

- Role: **Principal Engineer & Co-Founder**
- Sanskrit Parallel Grammar Algorithms & Mathematical Technology
- Funding and strategic sponsorship-seeking phase

: Bangalore (July 2025 - Aug 2025: 2 months)

- **IISc, Bangalore**: Graphs, Spectra and Convex Geometry
- Filed provisional Indian patent titled " **Computational Methods Driven by a Sanskrit-Based, Self-Refining Wave-Particle Numeric Framework** " to address Halting Problem & Quantum Limitations.
- Continued research on Sanskrit

Akashx.ai: Bangalore (April 2025 - June 2025 : 3 months)

- Role: **Technical Architect**
- The code is based on Apache **Doris/StarRocks**. Worked on **bridge mode** between distributed / cloud-based **shared nothing mode** and **shared data mode** cluster architectures to get best of both worlds using S3 based object storage and hdd + ssd.
- Designed **Databricks Unity Catalog** permission model sync with StarRocks.
- Analyzed runs and after finding need for CPU instruction-level efficiency, demonstrated ‘repne scasq’ construct outperforming SIMD (**AVX2**) in specific cases; Proposed architectural changes to better use RAM.
- Provided analysis explaining why Intel-based VMs **parallel-ism** can not be beaten by ARM based VMs.
- Developed a multi-core code instrumentation and tracing/debugging framework for the starrocks_be big monolithic binary, leveraging eu-nm to resolve and demangle C++ symbols, mitigating the slow, single-threaded nature of traditional tools like nm and addr2line.
- Analyzed **TPC-H** benchmarking; found and formally reported a TPC-H **spec** bug to **tpc.org**

: Bangalore (July 2024 - March 2025: 9 months)

- **IISc:** Deep Learning Foundations and Applications
- Society giveback, various DIY projects (details given in next page)
- Continued research on Sanskrit

Novell / OpenText: Bangalore (June 2001 - June 2024: 22 years)

- Role: **Windows Kernel** Developer, **Linux** User Space Developer; Lead SW Engineer
- Developed a solid understanding of the halting problem and **shared relevant parts** for debugging, coding and design purposes.
- Analyzed **gigabytes** of **Wireshark** traces, Windows **Procmon** traces, and Linux **syslog** traces to find critical issues.
- **Advanced** usage of Procmon, gdb, Windbg, Valgrind and various GNU/Linux tools; capable of writing **on-the-fly C code** in the debugger in resolving customer issues.
- Handled **millions** of lines of code primarily in C, with some C++, all involving complex **multi-threading** code for features' additions and fixing issues, without KT.
- Frequently and **proactively** resolved critical issues like product **incompatibilities** in restoring previous backups, spearheaded **innovative** solutions, and **addressed challenging roadblocks** in architecture and security in OES product development along with **improving performance** of a few products by redesigning.
- **Simultaneous** responsibilities at the time of leaving:
  - Solely managing Novell DNS (derived from **ISC Bind** code) and Novell DHCP (derived from **ISC DHCP** code) products along with Windows NCP (CIFS & NFS like from Novell) with **21 Windows kernel drivers** and **Linux** client for NCP - all as 1 person team by the time I left company.
  - windows kernel: given details below
  - Developing XPLAT developer libraries for Novell NCP and XTIER smart OS layer.
  - Assisting Novell's **Domain Services for Windows** product team in maintenance.
- Earlier **Product** Responsibilities: **Storage** Management Systems Backup, developing **Cluster backup** services, innovative solutions in leap frogging team whenever product got stuck, **unearthing security design issues and developing solutions**, SSL-ising products. NetWare **Kernel Drivers**, Porting NetWare products to Linux user space, Work on design of **Snapshot** backups and updates to NSS **Distributed File System,** innovative enabling of the product for **IPV6** without code changes - to name a few.
- **Mentored** several teams on innovation and advanced debugging skills
- Demonstrated **removal of unconscious bias** in team **interactions** by developing innovative collaboration tools and methods based on **scientific** first principles.
- Worked as the only Inventions Committee member representing Novell IDC for 4 years.
- Took a break from job in June 2024 to **pursue** further education.

CORE WINDOWS KERNEL ACHIEVEMENTS:

- Entirely self-taught, no KT or prior training.
- Network File System Write Caching - Designed, coded, and tuned adaptive caching strategies for database and general workloads; optimized latency and data integrity; measurable WAN/LAN performance gains.
- Read Caching Redesign - Analyzed CIFS traffic patterns and redesigned NCP read caching logic for improved throughput.
- Customer Issue Resolution - Fixed multiple crashes and numerous high-severity customer issues in production kernel drivers.
- Secure Communication - Designed kernel to userspace packet shift for SSL encryption to overcome kernel SSL constraints.
- WAN Optimization - Developed ICMP ping cost-based nearest server selection for dynamic client routing.
- Command-line Expert - Automated debugging, analysis, and instrumentation workflows via CLI scripting (Windows & Linux)
- Trace Correlation Innovations:
  - Created method to link Procmon and Wireshark traces using portquery-inserted events for highly automated kernel-level debugging at code level with debug statements going directly into procmon trace with base64 encoding at assembly.

    - Instrumentation & Event Tracing - TraceLog GUIDs, ETW - when had problems, resorted to completely assembly level _penter _pexit instrumentation across 21 drivers with very low performance impact during debugging
    - Code-to-Network Mapping - Developed scripts inserting PortQryV2.exe events to a safe non-existent IP to generate correlatable Wireshark entries with Procmon trace for code-level mapping
- Common Information Model (CIM) - 2 U.S. patents, including filesystem interface to CIM (WMI is Microsoft’s implementation of CIM)

Research Contributions

- Google Scholar: Profile (https://scholar.google.com/citations?user=8apvvJsAAAAJ)
- Research Citations: Total = 832; h-index = 16; i10-index = 18.
- **27 patents** at USPTO on **Distributed Systems, Virtualization, Cryptography & N/W Security**.
- **1** personal patent (submitted provisional) on Sanskrit technologies
- Multiple hacking projects inside company; represented IDC for Inventions Committee

Skills: Assembly, C, C++; Windows Kernel, Linux Userspace, NetWare Kernel.

Additional Interests

Society Giveback

- **SV Ayurveda University, Tirupati (Aug 2024):** Developed a demo exhibit system using ESP32, Espressif & FreeRTOS libs, Android mobile programming, servo motors, in case of no movement in newborn, to simulate various scenarios for movement via a web interface.

- **SV Ayurveda University, Tirupati (Jun 2024):** Designed a milk testing and report generation system for lactating mothers using ESP32, pH, color, and temperature sensors.

- **SRK Math, Ulsoor, Bangalore (Jan 2020):** Built a PDF-to-text conversion system for uploaded files using Google APIs and giving regex search, all via a web interface.

Photography

Runner-up in IDC Photography Competition (2005)

DIY projects

Various electronic and IOT fun projects for smart home and domestic robotic devices

A few yearly hackathon projects:

- 2023: Affordable geriatric care in vision loss scenarios and big data generation for AI-driven care: Using nodemcu, push buttons, mobile phone programming for giving care, starting from simple daily tasks like checking thermometer (**demoed**) to replacing, say, Holter test eventually.

- 2018: **Data Leak Prevention** with cloud services: Used OpenCV to slice sensitive documents by recognizing boundaries, mixed them with others with high dilution, and used Google Cloud APIs for text recognition. Reconstructed documents and retrieved information using coordinate data.

- 2012: Porting to **ARM**: Ported and compiled a few Novell products to Toshiba AC100, the official reference device also for Ubuntu Linux ARM port.

Conferences:

- SUSE developer conference, 2007: Prague, Czech Republic; attended, presented on versioning file system.
- SNIA Storage developer conference, 2008: Santa Clara, CA, USA: paper got accepted.
- ICVGIP, 2024: IIIT Bangalore: Attended specific part when **data leaks** were discussed.

Appended to this resume are,

1. Small writeup on halting problem titled,

" An understanding of the Halting Problem for making super fast systems "

2. Article on Innovation titled,

" Innovation Sparks - 1 "

3. Writeup on an experiment about group level innovation titled,

" Randomised Control Trial (RCT) inspired forum experiment "

An understanding of the Halting problem for making super fast systems

Halting problem can, in practice, be understood by the failure to avoid queue in parallelism and the inability of goal oriented selection using machine learning. This affects quantum computers also.

Details:

My intention of following questions is to stimulate thought and encourage contemplation.

Question 1: I humbly request to see if there is any practical example of a parallel program, language, or software/hardware architecture that operates efficiently without relying on algorithmic queue. The utilization of algorithmic queue impose limitations on system parallelism and also demonstrate shortcoming in the algorithm.

Question 2: Could you please provide evidence of the functionality and effectiveness of goal-oriented selection using machine learning? I appreciate any context or resources you can provide, such as the article available at https://ncbi.nlm.nih.gov/ pmc/articles/PMC4186236/

We shall first explain the quantum computers part here.

The role of black body radiation studies in the end of 1850s drove progress in modern physics, due to initial difficulties in explaining the observed phenomena. The Heisenberg Uncertainty Principle, which states that one cannot accurately measure both the position and speed of a particle simultaneously, has led to conventional quantum theories like the "bootstrap theory" (of S-Matrix) with still unsolved equations, leading to abandoning it and developing alternate theories as several main stream ones.

John Von Neumann, the physicist who made significant contributions to computer architecture and programming, offered key insights into the Halting Problem. This problem, along with other aspects of quantum physics as per Von Neumann, Weber, and others, is a part of several open-ended questions.

Quantum computers aim to address the perfect parallelism from a unique perspective, but they also face limitations rooted in this problem.
I offer to summarize a few unproven aspects of quantum computers in both theory and application as follows:

1. Hadrons, as self-creating particles, present challenges, and conclusions on the data from the Large Hadron Collider (LHC) are debatable. Quarks and Gluons do not exist as per some theories and the indirect proofs can be explained using different theories also, not just the most popular one.
2. Existing practical applications like laser technology and quantum cryptography rely on 'conventional' quantum physics, despite unsolvable equations.
3. The practical use of quantum computers in theory is still several decades away, regardless of long running projects. They primarily compete against traditional computers in relatively narrow domains, although the scope is expanding.

4. Quantum computers are unlikely to have a suitable form factor for onboard integration.
5. Cost and energy considerations pose significant challenges.

For 1st question, examine these:

Bitcoin's no queue causing reliability issue; Uber-like apps' billing delays due to queuing; Memory limits in queue causing message loss in message queues; Databases recommend disabling OpLocks at file write caching level to address inconsistencies; Async protocols for WAN performance utilizing hidden queues; Mutex implementations using queues; Itanium's failure for compiler-handled concurrency; Erlang parallelism used in Whatsapp being limited by queue memory; asynchronous circuits being limited by memory of the queue (analogous to queue is different here); concurrency handling issues in Domain Specific Languages; inevitability to increase out-of-order queue size in practice for super scalar processor designs; queuing limits in real applications with GPU programming, and the list can go on and on.

In all modesty, the complexity of the second question goes beyond this relatively short format. To summarize consider this line of thinking: Skinner's work with pigeons and boxes, James Olds' rat experiments with those boxes, and the discovery of extraordinary pleasure centers that override basic animal instincts; triggering of unethical human brain experiments thereafter prompting a comprehensive ban; the 2000 Nobel Prize-winning discovery of non invasive brain study techniques rekindling interest in brain research; resulting 2003 'BRAIN Initiative' and the concurrent rise of machine learning.

Supplementary Notes - 1:

The omission of Alan Turing's interpretation is due to some intricate discussions his interpretation would necessitate. Given the challenge of simplifying them, I've respectfully sidestepped it.

Supplementary Notes - 2:

The connection between physics and computer science becomes more apparent when examining the field of regex indexing.

Closing thought:

"As far as the laws of mathematics refer to reality, they are not certain; and as far as they are certain, they do not refer to reality"
  - Einstein

Humbly request utile real-world examples for any discussion.

Revision History:
2019 Aug 17: GNVS Sudhakar: Initial Version
2025 Mar 12: GNVS Sudhakar: Current Version

Article from Novell IDC Tech Journal

https://issuu.com/novidc/docs/idc_tech_journal_v4

GNVS Sudhakar

Done MCA from NITK and been with Novell for the past 13+ years. IC (Inventions Committee) member.

Innovation Sparks - 1

Introduction

I will start with a question. Usually the data on servers gets backed up to tapes which will be stored in a secure place. How do you logically delete these backups without touching them? Think for 2 minutes before going ahead and reading the rest of the article. Here is a coded answer (to decode: replace with next character a->b, b->c ....): Dmbqxos adenqd. Cdkdsd jdx. Well this is the first patent I studied. A nice, simple but very utile innovation, isn't it?

Innovations in the beginning are abstract. In Shreodinger's "*Image of Matter*" we find these lines: "*....the physicist lives in two worlds: He manipulates such tangible objects as coils and vaccum pumps, but his ideas must be appropriate to atomic dimensions. He must enter a world in which mutually contradictory hypotheses much supported by incontravertible evidence must both be accepted ...*" In a way, being continuously inventive is just about living in abstract uncommon thoughts but verifying them with the litmus test of utility.

As an example, suppose we say trees sneezing causes wind and support the statement by taking analogy from wind caused by sneezing, how will it be looked like? But when we say earth's attraction force causes fall since attraction force (magnetism as a better example) causes fall, we accept it now since it is verified. But to him who discovered, the struggle must have been there to prove. I would like to quote what Einstein has said: "Common Sense is actually nothing more than a deposit of prejudices laid down in the mind prior to the age of eighteen"

Practical aspects

Let me touch some practical aspects of the innovation in the context of the litmus test mentioned: Whatever may be the source of inspiration, we need to spend time to think. However, in a corporate setting, where do we get so much of time? Well there can be many reverse answers like questioning back how do we get time for so and so things like watching a cinema. But the problem is a real felt at least. There is no substitute to spending time and here is where the real innovation starts I guess. An idea can be as simple as taking a leave and spending your time at office(nobody can assign tasks)/home for the entire day towards innovation.

Broadness of the idea is another aspect I would like to touch. When the idea is broad enough the level of acceptance is more. A better acid test is, can you start a company of your own with this idea. We need to aim for such broader ideas even to result for apparently small innovations. An essential ingredient for broader ideas is security. There may be many working in software development but not all are well versed with Network security aspects. I would recommend to atleast implement SSL client and server and subscribe to openssh mailing list for some time atleast to fill the requirement. "Cryptography & Network Security" by William Stallings is a nice book which I would recommend.

The mechanism of the idea actually need not be worked out to a granular detail. The thumb rule is when it can be explained to a person in the same field, just so much of detail is enough. When we find an idea with sufficient detail, even without an implementation, we can create a White paper, product presentation, or file a patent. All these do not require you to implement your idea, though it may help.

Problem sources

But, how to get the source of the problems? The work we do is definitely a source. In my case, I have also found the text books and sites like Google scholar were helpful. Sometimes, a problem may appear to be uncrackable, but it might be fun to solve it. An example is to factor a prime number product. Though it is not solved till now in public knowledge, I took it up and it helped me understand and discover the properties of prime numbers, which resulted in two patents.

Reinventing innovation

Somebody said the best way to understand a wheel is to reinvent it. I agree with it. Also in doing so, I realized many truths about numbers. They are difficult to explain, but I would still try to share my understanding below. I feel we have not been taught the basics about the number system properly in our schools nor innovation.

Here is a link about how pi was discovered: http://betterexplained.com/articles/prehistoric-calculus-discovering-pi/

A comment in the above site reads as "Why was this never explained like this in high school?" This is the whole point I would like to concentrate on. Mark Twain rightly said: "I've never let my school interfere with my education". We know children are very innovative. Growing up should not mean decreased innovation. The approach to mathematics can be made fun and fulfilling. For example, Srinivasa Ramanujan often said, "An equation for me has no meaning, unless it represents a thought of God". Einstein said, "But the creative principle resides in mathematics. In a certain sense, therefore, I hold true that pure thought can grasp reality, as the ancients dreamed". We can find more interesting examples in vedic maths. So we need to agree, it is possible to think differently.

When you innovatively solve a problem from a source, try again in a different way. This is another way to generate new ideas. When you find that your solution to a problem is already discovered, you can still consider it as a sign of success and progress.

Innovation Philosophy

Actually, in nature everything exists as one whole thing. Dividing itself is accepting unknown. Consider these examples: Which artist has drawn the colors of parrot like nature? We find Fibonacci arrangement in nature right upto arrangement in DNA. Aeroplane was designed after noticing birds. Music was inspired by cuckoos. Science + art + movement + ... exists as 1 in Nature. In nature we get everything together as one. It is we who divide it into different things depending on our limited range of perception.

We have seen different things about innovation. But why do we need to innovate in the first place? I would say it is seriously for the fun of it. No other reason really fits in there.

Inspiration for Ideas

If you think there is no inspiration for innovation, then reading so far proves you are interested and you have it! Inspiration for ideas comes from mystery. We call this by different names. For example, in case of science we call it as entropy. It can range from difficulty to compress random bit stream in computer science to the holographic theory in modern physics it is the entropy which is the inspiration to many things. Strength of secret for RSA algorithm comes from the fact that nobody knows (publicly claimed) how to factor large numbers. Entropy here is also not knowing how to factor. Stuff of this kind generally gives inspiration.

Innovation Definition

Innovation is uniting inside. Innovation comes from resolving conflict which can be identified through increased sensitivity to conflict. This comes from a "Not to fear" mindset. This is the most important quality needed for innovation and works at the subtlest of subtlest levels.

Areas for Innovation

I have seen patents in all imaginable areas in technology. So we cannot say that an area is not suited for invention. A good example is the email. I have observed many people trying to innovate the email. As it is the most used concept, many people generally tend to start there. While I do not discourage you, it is both easy and difficult to innovate with email because by the time an idea comes, it is already implemented since it is easy, somebody else already did it. Because of this it also becomes too difficult to innovate in email space. But, this shows any area can be taken invented when it becomes familiar.

Inventions review

When it comes to presenting an invention to the reviewing body, we need to understand that the entire review discussion is not typically conveyed back to you. In case the decision is NO, it does not mean the idea was not properly understood. The reason conveyed back to the inventor generally does not contain all details. This is true for most of the cases. It is also the responsibility of the innovator to convey the idea in a simple, easy to understand manner.

Prior art research means searching about your new invention related work in Internet to see whether it already exists. Generally 20 to 40 minutes of concentrated effort is enough in case of patent idea submissions. Also when understanding a patent, reading the first two claims and the abstract gives good idea about the patent. This method of understanding works for most of the patents.

Innovation Patterns

Some of the recurring innovation patterns I have seen are intellectual cross pollination, piercing through abstract layers, relaxing the original purpose, and so on. We can find many examples for first one. I read somewhere that IBM brings people from diverse backgrounds like physics, biology, chemistry and more to work on a single problem. For the second one though there are many, I would give this interesting example patent filed by a fellow Novellite from Bangalore: Raw sockets interface was used in a VPN product to provide a new feature otherwise not possible. For the third pattern we can find many examples, some are: Rsync, where integrity of data is relaxed or diluted to give faster synchronization, coating razor blades with plastic kind of material and decreasing sharpness to increase durability.

Effort for Innovation

Another aspect about time is, do ideas take time or do they come in a jiffy? We know the case of Archimedes and the state in which he got his famous idea. Deeper contemplation is definitely needed. The flash moment may be at a different time but the mind must be tuned and hence spending time for innovation is definitely required. Mozart says he gets ideas for music when he wants. But we know the effort required to master classical music. Similarly kekule got the benzene structure in a kind of meditative state. The effort behind it can be understood. James Rothman, who won 2013's Nobel prize in medicine said he was "nuts" to attempt to reproduce the cell's complexities.

Summary

In the end, I would like to summarize as "Innovation is uniting inside. It starts with identified conflict. Identifying conflict requires the necessary "Not to fear" mindset. The "Not to fear" mindset comes from harmonious understanding.

Epiolgue

This is the first of the article series which I have planned about innovation. I will try to cover structured innovation and more in the next article. Please reach to me for anything in general. I will try to cover any shortcomings or requests in the next article.

Randomised Control Trial (RCT) inspired forum experiment as a solution for continued success

Abstract: RCT, which removed unconscious bias, is the reason behind the success of modern medicine. The same principles are applied in a corporate setting and a solution is proposed based on it for the problem of aligning brains. Encouraging results of a prototype forum are shared along with, unsolved problems.

Rational: For several reasons paradigm shift has happened: From individual to multitude at different fronts in research and development. This is not new. However achievable-ty of aligning all R&D brains is presented here.

Problems: 1. Some excellent fresh joiners who were innovative guarantee-ly when as child, after growing up and joining a good company do not speak up due to several factors. Also, for many employees, a higher in position need not mean high in excellence and innovation, especially in R&D. Similarly about low in rank. Yet the bias in decisions and arguments happens when the ideas are associated with the individuals (meaning when origin is known). This is the same problem which was solved in modern medicine, leading to its success.

2. For a given a problem, a perfect solution which is unchanged for at least long time will typically be unique. Depending on the experience, each can catch some part of it. When everybody shares (teaches) everybody else correctly, it helps all to come to common understanding correctly and discover the near perfect solution excellently with greater speed in a constructive and destructive/disruptive way together in a good company. When this is not done correctly, then with size, entropy increases exponentially. Entropy here is akin to not knowing how to use higher denomination currency notes pack and (effectively) using it as firewood.

Proposed solution: Quotes: "The most important discovery of modern medicine is not vaccines or antibiotics, it is the randomized double-blind test, by means of which we know what works and what doesn't"[Ref1]. "Honesty requires time....Those who were pressed for time lied; those who had time to think told the truth"[Ref2].

Proposed solution is a technical discussion forum where all can post as just 1 single user, say, "company". Nobody knows who has posted. Discussions happen by the worth of the idea and never by the influence of the individual even to an iota of a degree. One can post any relevant stuff to any degree as long as it is not a crime. All project discussions happen here, including all relevant managerial decisions posted 'appropriately'. A member of any team can get to see technology/product discussions of any team and contribute by answering or by questioning. The forum would mature over the time and evolve in a good company. Forum itself is any special techgroup like patents committee. To trace the contributors to the idea for patenting, the user can give an one time encrypted string at the end of the message to prove the identity when requested to reveal thus proving him/her to be the inventor.

Prototype results: Couple of years back a forum with above characteristics was setup in my team using custom modified myBB forum software. During discussions on the forum, we could get go to such depths that the initial discussion which apparently had no managerial interest finally turned to a level where assignments were given in the forum itself about further research and surveys to be done. Almost all threads lead to unique depths and showed progress which otherwise would not have been possible. Getting in writing gave more clarity. It was a different experience. Even today I do not know, who in the team has contributed to the thread except for the knowledge I got. Unsolved problems were: Inviting experts anonymously (integrating with email can be a solution) and maintaining a separate server than the corporate server (can be solved).

References:

1. The Chronicle Review. Volume 49, Issue 21, Page B20. (Robert L. Park, Director of public information for the American Physical Society, Physics Professor at UMCP, the author of Voodoo Science: The Road From Foolishness to Fraud (Oxford University Press, 2000))

   http://leptonica.com/cachedpages/bogus-science.html

2. http://www.scientificamerican.com/article/what-science-tells-us-about-why-we-lie

(for more details please contact GNVS Sudhakar) (Bangalore, 2014 Nov)