A CRITIQUE OF BLOCKCHAIN/BITCOIN

Excerpts from
BITCOIN: The Magnet Snare

Kaiser Hugan

CONTENTS

PREFACE

The single most important role computer technologies have played (and should in the future as well) for the humankind, in my humble opinion, has been about getting work done faster and more efficiently, and solving problems humans would not be able to otherwise.

The need for performance and efficiency in computer technology and implementation is paramount. It has been ingrained in my head since studying "Algorithm Design and Analysis" and "Computational Complexity" tens of years ago.

As the lead architect for several large projects in supercomputers and telecom equipment over the years, and as the Chief Architect of a mid-sized international hi-tech corporation with staff in thousands before my retirement, I strove to achieve efficiency in hardware and software design both for lowering product cost and for lowering operational cost for the users, while balancing the need for productivity in R&D.

Over the years, I've had a good share of encountering designs and implementations where performance and efficiency didn't get where they should be. While I've worked with a lot of extraordinarily bright and talented engineers, it was not uncommon for me to stumble on a system-wide performance issue caused by some individuals.

I've come across implementation performance deficiency problems as simple as individuals not knowing when and how to use a hash table properly for maintaining context data structures. Believe or not, I had to investigate system wide performance issues on numerous occasions, where some individuals used hash tables to maintain context data structures intended for constant-time access on average (rightly so for those specific cases) but failed to set proper table sizes, leaving the default table size of 64 of the hash table Template code utilized and as a result leading to hundreds of thousands, even millions in some cases, of context data structures serially linked in a linked list to a single hash table bucket. You can imagine what kind of operational performance issue this would cause. I've seen this same problem time and again on a few R&D projects of hundreds of designers each, caused by different individuals.

When I was interviewing software engineer candidates for a project where performance was critical, out of a few hundred candidates each with well over ten years of development experience and having a degree up to PhD, only a couple were able to correctly answer this simple question: Can a small C++ inline function with just a few lines of simple code run slower, much slower, than its non-inlined counterpart? If so, under what conditions that would happen and why? These few lines of simple code of the function are just simple arithmetic operations and in the pure execution cost sense would not cost anywhere close to that for the stack operations related to a regular function call. If you are reading a book on the C++ language, you will almost certainly find that the book tells you this is a perfect case for inlining the function—*precisely the kind of scenarios the inline function feature is for*.

While the thousands of software engineers I worked with over the years had good education in Computer Science/Engineering from good schools, I found many were not able to connect the dots between the different subjects they learned in school unfortunately, many of which had not been directly used in their work since university.

Using all kinds of middleware, though helping with productivity, made many lose the top-to-bottom view of a system. It's not uncommon to find software engineers unable to make a connection of the performance of their code in a high-level programming language to cache performance, instruction pipeline, etc. Many were not even able to have a good grasp of the impact of system calls made from their programs directly or indirectly.

I felt I could share some of the experiences to help the new generation of engineers to have a better system view while developing individual components of a system, a large system in particular.

But what actually made me start writing late in 2019 was the exposure to Blockchain and Bitcoin. I heard about them years ago but never actually looked into them. I took a look into Blockchain after my retirement. My immediate impression was that anything based on Blockchain could not possibly scale to serve the whole world as a cash transaction system.

Anybody with experience in distributed systems should know this would never scale. Fundamentally, it's replicated centralized processing, despite the huge number of miner nodes in the Bitcoin network giving some a false perception of distributed and parallel processing.

I tried to look for technical critiques but all I saw was a deluge of praises and ravings. People were only looking at how the chain helps increase the credence of the transaction records as it grows, not to mention many being deluded by the myths surrounding Blockchain/Bitcoin. But what about the cost? Will it even be able to scale to serve the world, hypothetically giving it a green light to replace all currencies?

I was disappointed at not finding any balanced technical critiques. Maybe there were some, but at least not many, not easy to find.

I've seen many hypes in the industry and academia over the years. But this one has been the craziest to me, perhaps because it's directly related to money—investment speculation.

So, I decided to write up something to counter the main-stream opinions, which I viewed as highly inflated hypes.

I thought about writing an academic paper to submit to a conference or journal, but decided against it in the end. The Blockchain/Bitcoin hype has been very much about myths created by the Blockchain/Bitcoin creator and misconceptions by investors who have been misled. So, I wanted to reach more people. Plus, I wanted to help the new generation of engineers get a better system view and be better prepared for design for performance as I mentioned earlier. So, I decided to write up a fictional story to allow more people to understand.

My initial book was titled "B Coin: The Architects", made available on Amazon in June 2020. Shortly after, I made some minor update and changed the book title to "Bitcoin: The Magnet Snare", again made available on Amazon.

It's encouraging seeing the following comment by a reader of the book in his/her review:

> "... Just searched what the per-transaction cost is like for Blockchain. A 2017 report said it consumed the same amount of energy for a household for a week. How could that justify for the payment processing cost for purchasing a coffee if Bitcoin is to be used as a payment system for regular purchases? I don't know what else can be more absurd.
>
> You folks working on any Blockchain based software really need to sit back and use your brain to think."

To reach more people with no Computer Science/Engineering background, I cut all technical content that was not essential to the understanding of the issues with Blockchain/Bitcoin and created a lighter version "Bitcoin: The Snare" and made it available on Amazon.

As a critique of Blockchain/Bitcoin, this book has been created to allow readers to get directly to the core issues of Blockchain/Bitcoin without being bogged down by the lengthy fictional story. I have taken four chapters from the book "Bitcoin: The Magnet Snare" and written up some bridging text to connect these excerpts. The result is a much condensed short story about three computer professionals with different perspectives with regard to Blockchain/Bitcoin.

I welcome all comments and feedback, positive or negative. What I am trying to get is to spur healthy technical discussions and debate among computer professionals.

BRIDGING TEXT

It was April 28, 2011.

Congruous Computing, a supercomputer startup based in Boston, had run into financial challenges in the aftermath of the 2008 financial crisis after its founding in late 2007. It had been teetering on the brink of bankruptcy, despite the superior scalability—the most important attribute for a supercomputer—of its advanced supercomputer product *Symphony*. It had been forced to let the entire engineering team go half a year earlier, left with only 8 key people looking for opportunity to rebound.

It had been a roller-coaster ride for Huffman, the Chief Architect of Congruous. The night before, he was informed by the CEO that the Board of Directors had decided to shut the company down the next day. But when Huffman arrived at the company office in the morning, he was surprised to learn that the plan to close down the company had been reversed. The company had received a major order for Symphony from its key customer, the National Labs.

The President of the United States had issued an executive order and put some emergency funding into strengthening the competitiveness of the United States in supercomputing, because the Chinese had recently unseated the US from the top position in supercomputing per TOP500 ranking. On the other hand, the TOP500 ranking of supercomputers had been dismissed by Huffman as nothing more than "Smoke and Mirrors".

Huffman was tasked by the CEO to recruit back the engineering team, most of whom had gone back to Ottawa. They were Ex Ottawa employees of the former Northern Research, one of the world's top telecom equipment makers.

Huffman had been making calls and sending emails the whole day, trying to set up a meeting in Ottawa over the weekend with the former Congruous engineers.

On the top of the list of people Huffman trying earnestly to get back was Vincent LeBlanc. Brought in by Huffman shortly after the founding of Congruous, LeBlanc had been the team lead of Computing Platform and Data Path before being laid off half a

year earlier when the whole engineering team of Congruous Computing was let go. Huffman had not been able to get a hold of him. He had called him a few times in the morning and in the end left him a voice message to call back.

CHAPTER 21

Huffman gazed out the window. Nightfall had come but LeBlanc had not called him back yet. Huffman's mind drifted back to the last serious conversation he had had with LeBlanc at Congruous Computing.

It had been the day before the last layoff. Huffman had learnt about the Board's assessment of the supercomputer market. He had anticipated an unwelcome, major change to descend on the company shortly, likely the closing down.

LeBlanc walked into his office and sidled up to him. "Can we have a talk in the meeting room?" LeBlanc whispered while pointing at the meeting room across from Huffman's office.

"Right now?" Huffman asked.

"Yes, please."

Huffman followed LeBlanc into the meeting room. LeBlanc closed the door right away and locked it.

Huffman found himself a little uneasy. *Did Vinny hear something about the Board's assessment of the HPC market? Is he wondering what will happen to the company? I cannot say anything or confirm anything. I don't even know how things will unfold myself. Plus I could not possibly disclose company confidential info entrusted to me. Vinny has to wait to hear from his chain of command. Plus, Vinny can find a good job in an instant by himself. I could make recommendations or reference if he needs.*

Huffman was not expecting to hear what LeBlanc was to tell him. What he heard left him relieved. "Kyle, I am very sorry to tell you I am going to resign. I have really liked and enjoyed working with you all these years. Learned a lot from you. We have made a fantastic product in HPC, so much more scalable than our closest competitor's. But the economy and the market have not been kind. The company has been faltering."

"Where are you going?" Huffman asked.

"I am going into Bitcoin mining. Bitcoin is a cryptocurrency based on the Block Chain technology. A friend of mine has been mining Bitcoin and made tons of money.

He wants me to join him and to help turn the Bitcoin mining operation into parallel processing. Have you heard about Block Chain/Bitcoin before?"

"Yes, I did."

"Oh, that's great. Would you be interested in doing that? We could do something together, if you are interested. With your expertise in algorithm design and parallel processing, we can make truckloads of money very quickly." LeBlanc enthused.

"Actually, I am not interested in doing that. Thanks for asking. I heard about it, not because of the putative P2P money transfer service Bitcoin is supposed to provide, but rather because of the claim of a solution to the Byzantine Generals problem. A friend of mine mentioned that the Block Chain scheme provided an elegant solution to the Byzantine Generals problem."

LeBlanc had never heard of the Byzantine Generals problem before. "What's the Byzantine Generals problem about?"

"I came across the Byzantine Generals problem in the early 1980s. Our Discrete Math prof introduced us to the problem of distributed agreement making. It's about how you can reach an agreement amongst a number of people who may have very different ideas."

"I certainly know about distributed agreement making or distributed consensus building. It's about how people can reach an agreement through talking with each other. For a distributed computer system, it's about how the distributed computers can reach an agreement through messaging over the network connecting them. Many of the networking products we did before involved distributed leader selection for various kinds of functionalities, and active unit selection in our redundancy schemes. They were special kinds of distributed agreement making." LeBlanc interjected.

"Exactly." Huffman continued: "To make this topic sound more interesting, my prof started describing a city under siege by troops under the command of a few Byzantine Generals. The Byzantine Generals needed to coordinate and agree on a common time of attack. They had different ideas about when it's the best time to attack. Some believed midnight was the best time, others believed an hour before daybreak was the best; still others … The Generals were with their own troops and separated from each other. They could communicate with each other only through messengers. The question was how you would plan the messaging and create a way for the Generals to reach a consensus on the common time to start the attack. My prof asked us to model this problem using Graph Theory and to design a distributed algorithm as homework. My prof also attributed the Byzantine Generals way of describing this distributed agreement making problem to a then new research paper. I am aware of a few other variations of the Byzantine Generals problem, including some in fault tolerance. So, I went to check out what the Block Chain scheme had to offer in that regard." Huffman explicated.

"So, what did you find? Is Block Chain an elegant solution to the Byzantine Generals problem you talked about?" LeBlanc inquired.

"No, not at all. It's just an egregious claim. It does not even involve a generic consensus-building component, where each party has an individual stand to start with. The Block Chain scheme, in concept, is more of a distributed validation or certification of versions of the same event presented by the participating parties, mostly honest guys but possibly some fraudsters. There is no such thing as each individual having a legitimate individual opinion and needing to come to a consensus, which may possibly be different from the initial opinion of every party involved. It has a closer resemblance to the fault-tolerance version of the Byzantine Generals problem."

"In that case, is Block Chain a good solution to the fault-tolerance version of the Byzantine Generals problem?" LeBlanc asked.

"I don't think so. It's at best a tenuous claim. I don't even see any attempt at achieving a semblance of efficiency but rather deliberate efforts to waste computing power and energy. We could not have possibly used that kind of methodology in any of the networking products we've made before, say for network level diagnosis. What we needed was real-time performance, besides the required logic functionality. We've had plenty of experience doing that for real products. The Block Chain methodology is even slower than a snail crawl. It's like a snail continuously trying to jump up without even crawling along. The snail is just exhausting itself by continuous attempts at jumping at its original spot. It's inconceivable that anybody would use this methodology in a networking system."

"I see. Fervent protagonists of new technologies almost always stretch the truth. They always try to claim a new technology could be applied to where it couldn't. It always happens at the inception of a new technology. Unfortunately, too many people like to follow fads. Dust will settle in the end." LeBlanc had seen a few industry hypes before.

"That's absolutely true." Huffman agreed.

"But what do you think about Block Chain and Bitcoin from the perspective of a distributed cash system as a technology? I know there are people who do not accept Bitcoin and even believe Bitcoin is worth nothing. They all came with an economy or finance related background, famous economists and Nobel Laureates included. Some Nobel Laureates called Bitcoin a speculative bubble. But I think they are just not able to comprehend new technologies. They have never learned Computer Science. These old guards have been left in their antiquated castle and are afraid that their moat will be breached. This has been a Copernican turning point of the monetary system. The world has moved on beyond their imagination. I don't give a shit." LeBlanc raved about the new cash system and lambasted the economists who did not buy in.

LeBlanc hoped Huffman would share his thoughts on Block Chain/Bitcoin as a technology. *Kyle may find a way to greatly improve the performance of Bitcoin mining, or better still, find a way to improve the Bitcoin scheme. Or create a better variation altogether.*

"From the system design perspective, what I have seen is dismal scalability and wretched computing inefficiency. It could at best be used in a limited scale, such as for inter-bank transactions between a few banks, minus the coin-value assurance problem which is for an economist to figure out."

Huffman's face started to turn stern. "It's beyond me why people have proclaimed this to be a utopian distributed cash system for the world, transcending all boundaries. Think about the population of the world, several billions. Assume a single person makes a number of transactions a day, buying coffee, meals, groceries and other things. We are looking at over one hundred billion transactions a day. The massively replicated central processing of the Bitcoin network will crash way before it has reached even just a thousandth of that scale, or …"

LeBlanc had known Huffman for a long time. It did not surprise him or faze him in any way when he heard Huffman criticize and deplore the deficiencies in the scalability and performance of Block Chain/Bitcoin. *Kyle, of all people, does that all the time. Kyle believes every computer system or computerized system must be efficient and scalable. He has made his life dedicated to making systems more scalable and more efficient. That is who Kyle is.*

LeBlanc saw opportunity, big opportunity. *Kyle will make it so much better, so much more efficient, and so much more scalable.* But when he heard Huffman characterize the distributed processing of Block Chain network as *massively replicated central processing*, a shudder ran through his spine. Before he could react, Huffman unleashed a fusillade of bullets leveled at the new technology he had recently fallen in love with:

It is paradoxical to claim distributed processing while it is actually massively replicated centralized processing with no bound on replication or on exploding per-transaction processing cost.

It is paradoxical to claim that user transactions do not place any trust on any third party while regular user wallets in practice implicitly put trust wholly on a small number of mining nodes it can connect but may not even know who they are.

It is paradoxical to claim user privacy and security while all transactions are explicitly made public and the full money trails are explicitly revealed by the Block Chain.

The proof-of-work scheme for deciding the granting of a coin is nothing more than a competition to see who has wasted more computing power and energy in gambling than everybody else has, with no meaningful work done.

You need to go back to ancient Greece to find a Sophist most deft at specious reasoning to smooth up the paradoxes.

LeBlanc found himself shaken by an earthquake right at the epicenter. *Kyle has just strafed almost every single core value of Block Chain and Bitcoin with a salvo.* LeBlanc's head was turned into a cauldron of conflicting thoughts.

Huffman was his idol, the brightest and most ingenious person he had ever met. To him, Huffman was a veritable demigod. Huffman had a preternatural aptitude to see things other people could not. It had been Huffman's métier to spot submerged dangers ahead and steer the R&D teams around them. Huffman's uncanny foresight and acumen had been proven time and again in R&D projects LeBlanc had worked on at Northern Research and Congruous Computing. *The other people just do not have the sagacity to discern them, regardless of how much R&D experience they have and academic degrees they have received*. LeBlanc had worked with over a thousand people in his career over the years—architects, engineers, and development managers, but no one could be compared to Huffman.

Block Chain was different. It had been accepted and heralded by so many people, far beyond the number of people LeBlanc had worked with in his career. Some even hailed it as the future of the society, bigger than the arrival of the Internet. So many computer engineers and computer science professors had been proponents. So much research on various applications of Block Chain. B this, B that, B pig, B cow … *How could so many people B So-Wrong?*

The more LeBlanc fought it, the denser the fog in his head became. He knew he was in over his head. *I need more time to digest and think about this.* LeBlanc was a great OS expert and he knew what the best way of dealing with this monstrous task was—lower its priority and the OS would automatically select other tasks of higher priority to work on.

Instead of responding to Huffman's comments, LeBlanc changed the subject, "Kyle, can I borrow or buy the prototype computing boards with GPUs we made before? Maybe you can put in a word for me with Brian. They have been sitting on the shelf doing nothing but collecting dust."

A GPU vendor had tried to lobby Congruous Computing to create a new computing board with their GPUs for Symphony to increase the FLOP power. The VP of PLM, Miguel Palacio, tested the idea with a few potential customers. The applications of those potential customers either had a large amount of matrix operation or were Embarrassingly Parallel. Those applications were a better fit for a supercomputer with GPU processors as co-processors on their computing boards.

However, none of them would commit to buying such computing boards. They worried about the extra software development costs required to make use of this capability. Their software had been developed over many years and they could not justify putting a large investment into re-coding, especially in this bleak economy.

In the end, The GPU vendor offered to cover all the cost of developing the prototype boards up front. Prototype board development and testing ensued. Congruous Computing simply depopulated 3 processors from its 4-socket computing board and added a daughter card to house 2 GPUs. Unfortunately, no customers bought in, despite the intensive marketing efforts by Congruous Computing and the GPU vendor.

"What do you need them for?" Huffman asked.

"Block Chain *Proof-of-Work*, the main task for mining Bitcoins. I can have many nonce values tried in parallel with a SIMD processor of a GPU. It's just a tiny little hash code I need to modify. Not like our targeted customers who have a lot of legacy code to contemplate. Between the SIMD processors intra or inter GPUs, there are absolutely no interactions required. All I need to do is partition the data range of the nonce values for trials between the SIMD processors."

LeBlanc grinned devilishly, "This solution scales infinitely. It's an apotheosis of Embarrassingly Parallel computation, the kind of parallel computing Congruous has deemed to be for dummies."

* * *

The next day, LeBlanc was laid off. His resignation letter was still sitting in the desk drawer of the Director of Software Development, not even passed up to the VP of Engineering for approval yet. Both the Director of Software Development and the VP of Engineering were also let go.

Ironically, it turned out better for LeBlanc than he asked for—he received a severance package, which he had never bargained for. Better still, he took home all four prototype computing boards each with 2 GPUs for only $400. Brian Hailey, the CEO, did him a favor of letting him take the four computing boards practically for free, partly atoning for his guilt in laying off one of his best guys.

That $400 LeBlanc paid, later proved to be the best investment LeBlanc had ever made in his entire life.

BRIDGING TEXT

After leaving Congruous Computing, LeBlanc joined his friend Nick Galliano in a Bitcoin mining venture in Gatineau, Quebec.

LeBlanc brought in with him the four Symphony prototype computing boards with GPUs. By splitting the range of nonce values for trials for Proof-of-Work amongst the SIMD processors of the GPUs, he managed to speed up the nonce value trials big time.

His Bitcoin mining operation with the four computing boards with GPUs turned out to be a huge success, far surpassing his friend Galliano's original 32 regular servers.

Their Bitcoin operation helped them generate significant wealthy just in months. LeBlanc went on an extended vacation to Australia, while maintaining his servers remotely.

While vacationing in Australia, LeBlanc met an Aussie Bitcoin miner Peter on Moreton Island near Brisbane in late February 2011. Peter told him a rumor that Satoshi Nakamoto, the creator of Block Chain/Bitcoin, had effectively left Block Chain/Bitcoin development. LeBlanc was shocked by the news.

CHAPTER 29

Sitting in the business cabin of Air Canada flight AC 36 from Brisbane to Vancouver, LeBlanc was tired. He had not slept well for two nights straight, since hearing the news of the departure of Block Chain/Bitcoin's creator. He returned the rental boat early, as soon as he had gotten back to the Manley shore. He had called Air Canada to change his return flights to the next day. He had abruptly brought his vacation to an end.

The disturbing news of Satoshi Nakamoto's departure had sent his mind into a tailspin. He could not fathom why Satoshi Nakamoto had departed Block Chain and Bitcoin.

He told a flight attendant that he was to skip the meal and reclined his seat to a flat bed. He managed to have a nap but a voice woke him up from his shallow slumber:

It is paradoxical to claim distributed processing while it is actually massively replicated centralized processing with no bound on replication or on exploding per-transaction processing cost.

It is paradoxical to claim that user transactions do not place any trust on any third party while regular user wallets in practice implicitly put trust wholly on a small number of mining nodes it can connect but may not even know who they are.

It is paradoxical to claim user privacy and security while all transactions are explicitly made public and the full money trails are explicitly revealed by the Block Chain.

The proof-of-work scheme for deciding the granting of a coin is nothing more than a competition to see who has wasted more computing power and energy in gambling than everybody else has, with no meaningful work done.

You need to go back to ancient Greece to find a Sophist most deft at specious reasoning to smooth up the paradoxes.

Once again, he had heard the voice of Kyle Huffman criticizing Block Chain and Bitcoin's core values. He had intended to think about those points carefully later. But he had totally forgotten about it.

He had been busy getting the servers set up and working initially. Then he had gone to set up an account with B$Xchange, the Bitcoin exchange of Nick Galliano's father. It had allowed him to exchange his Bitcoins for Canadian currency, when the rate was good. After he saw everything worked smoothly, he had gone on a long vacation to Australia.

The news of Satoshi Nakamoto's departure had perturbed him so much. It had rewired the neuron network in his brain and replayed Huffman's voice.

* * *

"*It is paradoxical to claim distributed processing while it is actually massively replicated centralized processing with no bound on replication or on exploding per-transaction processing cost*." He started to think about the first point in Huffman's criticism.

He had thought the Bitcoin network was a distributed processing system, as many others had. There were a lot of miner machines geographically distributed all over the world. *That means the machines are distributed. But Kyle has called it (massively replicated) centralized processing*.

LeBlanc thought about how a distributed system usually worked. A distributed system usually had distributed processing elements, either for distributing different types of jobs to specialized processing elements or for distributing different (possibly similar) workloads to distributed processing elements, possibly for geographical distribution of processing. In either case, a distributed processing system bore the characteristic of processing-load sharing over the distributed processing elements.

LeBlanc had also worked on systems that had replicated processing elements, mainly for fault tolerance. But he knew that the replications had always been limited, usually to 2, due to cost-effectiveness considerations.

He thought about what these miner machines were doing. They were all doing the same thing—basically accounting and arranging the records into a block. Each was taking *all* transaction data from *all* over the world and trying to create a transaction block. They were not sharing the workload but rather taking the entire workload for the whole world. *That means each miner machine is a central processing center, processing all transactions distributed all over the world. It is not any different from other centralized processing centers.*

There had been a lot of miner machines. LeBlanc had heard that the number of miner machines might approach a million soon. *That is truly massive, even just counting*

the number of active miner machines today. They were all doing the same thing—meaning *massively replicated processing*.

Putting the facts of centralized processing and the massive replications together, *we see massively replicated centralized processing*.

It's like having a million accountants doing the replicated accounting work, each having to do the entire accounting work for the whole world. Only one of the accountants will get his work recognized for a transaction block, while the remaining accountants in this pool of one-million see their work discarded and wasted. How could a single accountant be able to serve the whole world? No wonder Kyle said the Bitcoin network would crash way before it even reached the scale of serving just one thousandth of the need of the world.

LeBlanc then thought about whether there was any bound on replication of this centralized processing scheme. *There is clearly no bound on replication at all*.

For each transaction, it got processed on every single miner machine. Just with 1000 miner machines, the same transaction would be processed 1000 times. When the number of miner machines grew to a million, a single transaction would be processed a million times. *That's an exploding per-transaction cost*. Since there is no bound on replication, *there is no bound on the exploding per-transaction cost*.

LeBlanc came to the conclusion:

It is indeed massively replicated centralized processing with no bound on replication or on exploding per-transaction processing cost.

Kyle was right. Distributed processing in the Bitcoin network is a myth.

* * *

It is paradoxical to claim that user transactions do not place any trust on any third party while regular user wallets in practice implicitly put trust wholly on a small number of mining nodes it can connect but may not even know who they are.

LeBlanc moved on to the second point Kyle Huffman had made in his criticism of Block Chain/Bitcoin. LeBlanc knew removing the need for trust in a third party, the trust with one's bank in the traditional banking paradigm for example, was one of the central tenets of Block Chain/Bitcoin. It had been claimed by Satoshi Nakamoto that peer-to-peer transactions over the Block Chain based Bitcoin network did not put any trust in a third party.

Block Chain/Bitcoin had been built on the premise that the majority of miner nodes were honest, not fraudulent. If this assumption was not true, everything would fall apart.

For a user wallet to make sure its transactions had been handled correctly, it would have to talk to at least a majority of the miner nodes, even in the best case where it received completely consistent views.

However, it was not practical for a user wallet to connect and communicate with such a large number of miner nodes. A regular user wallet could only connect to a tiny number of miner nodes and some in fact only connected to a single miner node. These miner nodes could be all fraudulent. They could collude with each other and never put the transaction on the longest Block Chain at all.

The wallet put absolute trust in this small number of miner nodes. An ordinary user of a wallet generally had no idea who he had put the trust in. The miners were not accredited in any way by any authority. They could be any fraudster on the Internet. For a client of a regular bank, the client at least knew with what kind of bank he was doing business. The Block Chain based Bitcoin network was actually much worse in this respect.

Clearly, Kyle was right and Nakamoto had made a false claim. No trust placed on a third party for a Bitcoin transaction is a myth.

* * *

It is paradoxical to claim user privacy and security while all transactions are explicitly made public and the full money trails are explicitly revealed by the Block Chain.

LeBlanc had known all along that all transactions were made public. He had believed the user transaction anonymity through self-generated Bitcoin addresses would give the user privacy as claimed by Satoshi Nakamoto. Nakamoto had also claimed that the Bitcoin scheme gave a user privacy equivalent to that of stock market trading.

After a more careful consideration, LeBlanc realized the Bitcoin network disclosed and made readily available the full money trail for all transactions. For any money transferred, LeBlanc was able to trace it all the way back to where the coin, fully or partly used in the transaction, had originally been created. Not a single transaction on this path escaped.

The use of multiple addresses provided little cover. They could easily be lumped into a virtual account through money trail analysis. Law enforcement and any third party could easily follow the money trails.

The Bitcoin transactions had been mostly associated with the underground economy and investment speculation. There had been very limited transactions in payment for normal purchases that were above board. Many of the Bitcoin transactions involved local currency exchanges. The parties in these transactions

could be easily identified, and so could the parties in those transactions for normal purchases. Parties of transactions along the money trail could then be identified one by one.

For those fraudsters using Bitcoin as the vehicle for collecting their ill-gotten gains, such as ransoms, the money they had collected illegally could be easily flagged. They could be flagged as illegitimate property and treated just like stolen goods. Whenever they used the ill-gotten fund to make a payment, they could be easily caught. Even people who accepted Bitcoin payments with illegally obtained Bitcoins could be charged if they ever use such Bitcoins later, as they were in a way selling or in possession of stolen goods.

LeBlanc wondered why law enforcement bodies of the world had not moved on these criminals. *Either they are incompetent or, maybe for some unspeakable reasons, they do not want to take actions*. The jurisdiction boundaries did present some challenges for law enforcement but they were not insurmountable.

LeBlanc also realized a stock trade by an ordinary trader was provided with privacy beyond the anonymity of a transaction posted on the stock market. What were seen on the stock market were not stock trades of individual ordinary traders at all. Their trades had been combined into aggregate trades by financial institutions before being posted to the stock exchange. The aggregation provided another level of privacy. *What Satoshi Nakamoto has claimed is false*.

Anonymous address, multiple Bitcoin addresses and the stock trade analogy are just lame attempts at covering up the dearth of privacy with a coat of varnish. LeBlanc had seen clearly the fallacy of the arguments.

Kyle was right again and Satoshi Nakamoto had again made a false claim. The claimed privacy of Bitcoin transactions has just been an illusion and myth created by Satoshi Nakamoto.

* * *

The proof-of-work scheme to decide the granting of a coin is nothing more than a competition to see who has wasted more computing power and energy in gambling than everybody else has, with no meaningful work done.

LeBlanc had known exactly how the proof-of-work component worked. He had parallelized it over a cluster of SIMD processors, those embedded in his GPUs. LeBlanc knew even his friend, a University of Waterloo dropout, had gotten the gist right, although his analogy had not been accurate. *It is just like lottery gambling. If you buy more sheets of lotto tickets and make sure the selected number combinations are distinct, you will*

have a higher chance of winning. LeBlanc's GPUs had helped him win a lot of times in gambling on nonce values for hashing.

The creator has created a time-window based gambling scheme, where a gambler is allowed to make draws continuously until there is a winner. It's a supply-demand balance scheme. There is a fixed supply of one coin available up for grabs for each time window. The difficulty of winning is automatically adjusted per demand. If the demand is low, the difficulty is lowered. If the demand is high, the difficulty is increased. It's like adjusting the number of balls to be matched for a lottery win. If the gambling competition is low, you only need to match a lower number of balls to win. If the competition intensifies, you need to match a higher number of balls to win.

LeBlanc recalled the economists' criticism of Bitcoin, "*a speculative bubble*". He felt shame for his previous comment on economists. *They had seen this as a gambling scheme all along. Maybe they do have knowledge of Computer Science and have read the code for Proof-of-Work. They may have seen this as a bubble not only economically but also technically.*

Looking at the energy consumed by his servers, not to say the energy consumed collectively over the whole Bitcoin network, he knew his GPUs were big consumers of power and big heat generators.

He had to agree with Kyle Huffman.

Kyle was right again.

* * *

Block Chain/Bitcoin is a sublime fable of a technology, emblematic of packed myths inflated with hydrogen. It's a quintessence of a technology hype. The dichotomy between what is claimed and what is reality is irreconcilable. People have been deluded, including myself. I was infatuated. LeBlanc mused and felt sunburnt from within.

Even the title of Satoshi Nakamoto's high-level document, "Bitcoin: A Peer-to-Peer Electronic Cash System", is misleading. The Bitcoin network is not a peer-to-peer cash system at all, not as portrayed by Satoshi Nakamoto. LeBlanc conjured up a new picture of the Bitcoin network in his head:

There is a conglomerate of a massive number of central banks of questionable credence, each contending for its honesty in accounting.

A client of the conglomerate was led to believe he could make and receive payment reliably through any one of the central banks. "You don't need to put trust in any one of the banks. We make majority voting to make sure our transaction records are genuine and reliable and to weed out fraudulent

transactions. Any error is automatically corrected by our massively redundant voting system." He was told.

The client had received a payment through one of the central banks. He later went to make a payment using the money he had received. The bank through which he had received the payment had been closed. He went to another bank. But the other bank told him he did not have the money in his account, because the transaction for the payment he had received had not passed majority voting.

"It's your fault. It's your responsibility to check if the majority have voted for your transaction." He was told.

"How do I do that?" he asked.

"You check with enough number of banks to see if the majority have voted for your transaction."

"How could I possibly check to make sure the majority have voted for my transaction? My Bitcoin wallet does not do that."

With a bona fide peer-to-peer cash system, a receiver of a payment is concerned only with the validity of the payment he has received. He wants to make sure he could later on spend the received payment. That is all he cares about.

It is not a concern or prerogative of the payment's receiver to know how much money the payer has or where the payer got the money. It is none of his business.

The transaction should be strictly between the two peers, because it is a 'peer-to-peer cash system'. It is none of the business of any third party. Nobody else should even know the payer has made a payment to the payee, let alone the amount.

Third parties could be used as part of the transaction-processing conduit but none of them, not even together, should be able to find out the occurrence of the transaction. They may participate in the transaction processing, perhaps in some way similar to the mutual authentication of the subscriber and the network in a 3G or 4G wireless network, where a network element is delegated with the authority of authenticating a subscriber but yet is not given the subscriber confidential info involved in the authentication.

LeBlanc's mind went wild on what a bona fide peer-to-peer cash system needed to be.

CHAPTER 30

LeBlanc's deep thoughts were interrupted by the voice of a flight attendant. The flight attendant had seen that he was awake for some time now. She inquired if LeBlanc would like to have any food.

LeBlanc's stomach growled in response. Before departure, he had only had some coffee but nothing to eat in the Air New Zealand lounge, the only Star Alliance Lounge at the Brisbane International Airport.

"What do you still have for lunch?"

"Everything on the menu, Sir." The flight attendant pointed at the menu still sitting on LeBlanc's table.

LeBlanc made his selection for lunch.

LeBlanc's thoughts went back to mull over Block Chain again, after the flight attendant left to prepare his lunch. He realized the only remaining one of the central tenets of Block Chain, which had escaped Kyle Huffman's criticism, was chaining of the transaction blocks.

LeBlanc had understood well the effect of chaining on building up the level of trust for the transaction blocks in the Block Chain. The concept of chaining had not been foreign to him. He had learnt chaining of block ciphers in the Cryptography course in university. He had also had a more personal experience with chaining of CRC codes over disassembled packets.

It was a long time before. A field failure had been reported of the product he was working on at Northern Research. The failure was identified as a masked transmission error, detected by the CRC and yet masked by a bit error in the received CRC field itself.

CRC—Cyclic Redundancy Check—was an error-detecting code widely used in telecommunication. It helped detect errors in data packet transmission. However, there were errors that the CRC was unable to detect, albeit occurring at a very low probability. The CRC and the hash used in the proof-of-work in Block Chain had a lot of similarities in many ways.

The product LeBlanc worked on involved disassembling a large packet into a number of smaller packets before transmission. The received small packets were reassembled at the destination into the original large packet.

Each small packet came with a CRC field for error detection. In the failure that occurred to the product in the field, one of the small packets was received corrupted but the error was not detected by the received CRC field. A bit in the CRC field was flipped during transmission, masking the transmission error of the packet data, which would have otherwise been detected.

This problem could be avoided if another CRC field was added to cover the entire packet before disassembly. The problem with this approach was that disassembled packets might arrive out of order. It made it impossible to calculate the overall CRC field over all the received disassembled packets as they arrived, causing a performance issue and implementation complexity.

Kyle Huffman solved this problem by chaining the CRC fields of the series of disassembled packets. He added an additional CRC field for a disassembled packet to the next disassembled packet to be transmitted. As a result, the same CRC field was carried both in the disassembled packet the CRC value was calculated upon and in the next disassembled packet. The only additional step at the receiving side was making pairwise-comparisons of the corresponding CRC fields after receiving both disassembled packets carrying the same CRC field. It avoided causing a real-time performance issue or increasing implementation complexity.

Kyle Huffman's solution was quite similar to the chaining of transaction blocks in Block Chain, as a hash and a CRC were quite similar. LeBlanc implemented this CRC chaining solution.

But Kyle did not make any positive comment on the chaining of transaction blocks in Block Chain. LeBlanc remembered something Kyle Huffman had reminded him many times before, "Don't go overboard on anything. Everything has a scope and limit."

Going for a single global chain with Block Chain has gone way too far, ruining the good concept of chaining, making it impossible to scale. It entailed many other evils. LeBlanc mused.

LeBlanc recalled his own share of going overboard. As a neophyte fresh from school, he had been young and had just started working at Northern Research. His co-op work at Northern Research had attracted the attention of Kyle Huffman. The Proof-of-Concept project he had helped complete for Huffman was done well. Huffman recommended his hiring as a result.

When Huffman told LeBlanc the R&D project, based on the POC project LeBlanc had helped complete, had been approved, LeBlanc proposed to Huffman that the product development continue to use the JAVA language as it had been used with the

POC project. LeBlanc had been enthusiastic about the new JAVA language and the use of JAVA had worked out well with the POC project. *JAVA is the best programming language. It has everything*. The young engineer was caught up with the trendiest programming language of the time.

But Huffman gave him a small test project and a real-time performance target. The little project required a lot of bit manipulations and dynamic memory allocations and deallocations.

LeBlanc quickly realized he could never reach the real-time performance target with JAVA. Bit manipulation with JAVA was handicapped and memory garbage collection of JAVA wreaked havoc by stealing spikes of processing time away, spoiling real time responses. Plus, the execution of JAVA programs was universally slower, thanks to the extra level of execution with the virtual machine.

When he went back to Huffman to report his result, Huffman advised that he learn to appreciate different kinds of technologies and the limitations of each.

"*Nothing is perfect. Nothing has ever been and nothing ever will be. If somebody tells you that some new technology is perfect and has cured all illnesses of the past, he is simply lying, knowingly or ignorantly. Fervent apostles of every single technology have lied, for their proclivity to inflate. That's a fact of life. For a technology, you cannot just have faith."*
Huffman continued, *"We need to embrace new technologies. The best way to embrace a new technology is to look for what that technology is not good at, what the apostles are not telling you, and what is not in the literature."*

"We learnt it the hard way at Northern." Huffman started telling him a poignant story of a fiasco at Northern Research.

It was in the early 1990s. The newborn OO—Object Oriented—design had been the trend in vogue at the time. Like products from any other company, the telephone switching products of Northern experienced field issues due to bugs and other technical problems.

A couple of world-renowned OO design experts of the time had descended on Northern. They managed to convince the Northern executives that all the problems experienced by Northern had been rooted in the traditional procedure-based programming, even though Northern software had been mostly message and event driven. *You need a paradigm shift to the new OO design. The new OO design methodology will cure them all,* the OO experts told the executives.

The executives retained them to preach the new OO religion. A new project named GSP, a Generic Switching Platform intended to be the new foundation for all switching products of Northern, was born and these OO experts were entrusted with full technical authority for the project. It was Northern's highest profile R&D project at the time.

The OO experts pushed for the purest and the most authentic OO methodologies in every aspect and every part of the system, peremptorily disregarding the vociferous dissent of Northern's native technology experts.

When the moment of truth came, the called telephone set did not ring, until the testers had come back from their coffee and chat in the cafeteria during the first call's lab test. The testers had waited for a long time without hearing a ring on the called telephone set and believed the telephone switch being tested was not going to work. They decided to go take a break. The incident became the laughingstock for Northern engineers instantly.

It was later determined that there was no remedy to cure the new sickness other than to dump the millions of lines of software code and start all over.

These world-renowned OO experts left unceremoniously, shortly after their moment of coming to Jesus.

It was not a laughing matter for Northern however. Northern flushed hundreds of millions of dollars down the toilet. *Lesson learnt*.

"*Deifying a technology or methodology and turning it into a religion, will only serve to blind sight, to narrow perspective, and to dogmatize and petrify thinking*." Huffman warned his young acolyte and protégé.

BRIDGING TEXT

He had been working for the NSA covertly as a part-time contractor for a long time. He was known as "Tom" at the NSA. It started after his first visit to the NSA in the 1990s.

He was a key architect of the PKI—Public Key Infrastructure—project at Northern Research. After Northern Research released the world's first commercially available PKI in 1994, he was invited to the NSA headquarters in Fort Meade to make a keynote speech on PKI.

Soon after his speech at the NSA, he realized his speech on PKI was not the true reason for the NSA invitation. The NSA was bent on recruiting him to help secretly put backdoors into the network products of Northern and to help sneak backdoor hooks into telecom industry standards.

He initially politely declined due to concerns on privacy infringement. But Bill Ames, the Director of Electronic Intelligence of the NSA, managed to convince him to join by making use of the tragic death of Tom's niece, Amanda. The 13-year old girl had taken her life due to the cajolery and extortion by a sex predator over the Internet. The sex predator was never identified or prosecuted.

He accepted Ames' view of sacrificing *a bit of* individual privacy for the greater common good and decided to help law enforcement and intelligence.

He had not only done what the NSA initially requested of him but also come up with his own initiatives. After seeing the difficulty in identifying targets for wiretap before any signs of suspicion were observed, he told Ames that, instead of looking for a needle in the haystack, they should create a magic magnet and make the needle come to the NSA of its own free will. He told Ames how the NSA could create an open source software forum behind the scene and develop a software platform for peer-to-peer file sharing to support sharing of music files and other pirated materials over the Internet. It would allow the NSA to monitor the activities of those individuals who downloaded the software to their computers.

Ames coined the strategy Magnet Snare and the project Magnet Snare 1.0 was born. It was a huge success for the NSA. Peer-to-peer file sharing took off and grew quickly over the world like a pandemic.

He had done a lot covertly for the NSA over the years while working his day job at Northern Research and later at a smaller US hi-tech company.

Several years after the 911 terrorist attacks, he was dismayed at the country's inability to get to Osama bin Laden for so long. He suffered a personal loss in the 911 terrorist attacks—his sister died in the collapsed World Trade Center. He wanted to help American Intelligence and law enforcement to track down Osama bin Laden.

He came up with the idea of having a money transfer system free from all governmental control to lure the terrorists and financiers to use it to transfer money. It would be designed in such a way to have the appearance of transaction anonymity so as to bait the terrorists. A key attribute of this money transfer system was that all money trails were fully and easily traceable. That would facilitate the tracking down of terrorist financing and might help lead the CIA to Osama bin Laden.

A NSA/CIA project called Magnet Snare 2.0 was then born and the Bitcoin network was put into operation in less than a year thereafter. He used a formula to convert his name into a pseudonym, Satoshi Nakamoto, to hide his true identity in creating the open source forum and his activities.

In less than two years since the Bitcoin network went into operation, the NSA/CIA tracked down Osama bin Laden to the compound in Abbottabad, Pakistan, when Abu Ahmed al-Kuwaiti, Osama bin Laden's personal courier, made the first Bitcoin withdrawal, believing Bitcoin money transfer was anonymous and safe, and beyond the hands of any government, the American government in particular.

CHAPTER 104

It was the afternoon of May 1, 2011. Tom had come to his office at the NSA headquarters in Fort Meade earlier that morning. Bill Ames had called him the night before to ask him to come. He had not wanted to come this weekend. He would have rather stayed home with his pregnant wife Juliette. Plus, he had been in Fort Meade just three days earlier. But Ames had insisted, saying it was important.

Earlier, Bill Ames had called him into his office to have a private meeting. Ames had thanked him for his work on Magnet Snare 2.0, the super LNX hack, and for playing a leading role in breaking the camouflaged self-decrypting cipher codes.

Up until now, Ames had not told him the camouflaged encrypted message had actually been from the leader of the al-Qaeda terrorist organization, nor about the link between Magnet Snare 2.0 and the Camouflaged encrypted terrorist message.

Ames felt he owed Tom the truth. More importantly, telling him now would not cause any harm, because Operation Neptune Spear had for all intents and purposes been completed.

Ames told him all his work had borne huge fruit now. He told him a top terrorist had been located, thanks to Magnet Snare 2.0 and his leading role in breaking the camouflaged terrorist cipher code. He also told him a field operation was under way.

Tom was thrilled by the news. Sitting in his office, he thought about all the work he had done for the NSA. Magnet Snare 1, the hack for stealing the secret keys of a huge number of SIM cards, the backdoor in open source TOR implementation, and now Magnet 2.0, the LNX super hack and the code breaking of camouflaged ciphers. They had helped law enforcement apprehend terrorists and criminals and thwart Islamic and domestic terrorist attacks.

He thought of his sister and niece. *It has all been for them, to punish the guilty and to prevent other innocent Americans from falling victim like my sister and niece.*

But Tom also had ambivalent feelings. He had been a perfectionist in his technical work. He had considered his work on Magnet Snare 2.0 technically a piece of shoddy work, pushed out by what he had called *Time-to-Market pressure*, the same as he had experienced at the private sector companies he had worked at.

Had it not been for the time-to-market pressure, he would not have released the Magnet Snare 2.0 software the way it was. Not even the architectural design. *The architecture behind the released Magnet Snare 2.0 is totally wrong*.

Tom had known the whole time a single global chain of the released Block Chain design would not scale and would cause a lot of problems. Like his work on Public Key Infrastructure at Northern Research, he had originally architected it as a distributed tree structure for scalability, instead of a single global chain.

But having the ledger record blocks arranged into a distributed tree structure would make it a lot harder to allow the CIA to monitor and collect transaction data as needed. With the massive replications of miner nodes with a single Block Chain, all the data was readily available at every single miner node. It greatly simplified the CIA's access to all transaction data and made the following of money trails a cinch for the CIA.

He had figured out how to do it with a distributed and reconfigurable tree structure but it required a lot of software development. It had needed much higher funding and much longer development time. Like with some of his product proposals for his companies in the private sector, his proposal to NSA/CIA had been rejected on the grounds of time-to-market and funding shortage.

For the proof-of-work to pick the winner of a coin, he had originally conceived something he called *proof-of-talent*. He had wanted the miner with the highest talent to win the coin—the miner who could prove he had used the least computing power and used the least amount of energy to do the job. He had always been conscious about computing efficiency and energy consumption. He had never liked to waste energy and cause huge CO2 emissions to pollute the environment.

But he had ended up settling on the simplistic gaming and gambling scheme of proof-of-work, because his development proposal for this part of design had also been rejected by the NSA/CIA on the grounds of time-to-market and funding shortage.

Thinking about where the Bitcoin network was now and what the concept of Block Chain had led to, he rued not having fought hard enough for the right design. Block Chain and Bitcoin had taken on a life of their own now, out of his control. He hated to see the ever increasing, astronomical amount of energy being wasted and the unconscionable amount of CO2 emissions generated as a result. A pang of guilt and regret swept across the pit of his stomach.

* * *

He recalled his visit to Beijing a month before. He had gone there to attend an international conference on Internet security. He had gone for dinner with a friend of his, a former classmate at Stanford who had been a Guest Professor at Tsinghua University in Beijing while on sabbatical.

His friend told him over dinner there was going to be a Block Chain Application and User Conference the next day, hosted by his Tsinghua colleagues. His friend asked him if he would like to attend, since his conference on Internet security was completed.

"Block Chain has caught on fire here in China." His friend said.

Tom had not told his friend he was the one who had invented Block Chain. In fact, he had never told anyone outside the NSA and the CIA. *It's a matter of national security*.

"I thought the Chinese government has banned Bitcoin," Tom said.

"Yes, but only the exchange and trading of Bitcoins. Bitcoin mining has been flourishing. So have research on Block Chain and its applications." His friend said. "The Chinese government has banned Bitcoin trading, fearing a hemorrhage of their banking system. The government controls who can make a change to foreign currency and how much." His friend explained.

"That's interesting."

The next day, the two of them went to attend the Block Chain conference.

There was a presentation by a young man from an organization named Block Chain Pig Research Institute.

"What does this research institute do? What's he talking about?" Tom was baffled by the name of the organization and wondered what the presentation might be about. *Block Chain Pig? Having pigs chained up? Using Block Chain in pig trading?* Tom could not imagine what the presentation might be about and why a whole research institute would be doing just that.

"I don't know. It's in Chinese. My Chinese is not good enough for this." His friend replied apologetically.

Then there was a female motivator, a parvenue. She talked about her luck from investing in Block Chain early. She urged others to follow suit. Tom heard thundering applauses and gales of laughter from time to time.

"Her speech seems to have garnered a lot of interest. What's it about?" Tom asked his friend.

His friend was equally baffled and asked one of his Tsinghua colleagues to help interpret.

"She's made an intuitive explanation of Block Chain to would-be investors. She said Block Chain was like a chain from grandfather to his grandson, to the grandson of his grandson, to the grandson of…" The Tsinghua professor explained.

"What?" Tom was dumbstruck. He could not make any sense out of it.

"This relates to a Chinese custom and the Chinese psyche, difficult for a foreigner to comprehend," the Tsinghua professor explained. "The grandfather is traditionally the patriarch of a family, the ruler. A grandson is at the bottom of the family hierarchy. You can hear kids sputter insults at each other in schoolyards all over China, 'SunZi, I'm your DaYe.' A kid calls another kid grandson and proclaims to be his grandfather. Even some girls do that as well."

"Isn't it better to be the grandson, being taken care of? I wouldn't take it as an insult, if somebody calls me grandson. It may actually endear the person to me." Tom asked.

"This has nothing to do with who takes care of whom. It's about the social hierarchy. China has always been a social edifice with distinct rungs. We've grown up in this tradition, wanting to be the ruler and treat others as our subordinates."

"I see. But what does that have to do with Block Chain?"

"It has a lot to do with Block Chain. The Genesis Block is *the grandfather* at the very top of the scale. It was the single one entrusted with the highest power. The level of trust decreases fast, going down the chain from the Genesis Block. The Genesis Block is the easiest to generate. The difficulty of generating a new Block, or a new coin, increases greatly with much higher cost as you go down the chain, because they are subordinates. So, it's better to be a grandfather at a higher level. The speaker was encouraging people to invest in Block Chain as early as possible, to be a grandfather at a higher level, instead of being sputtered the grandson, the grandson of grandson…"

Tom recoiled in dismay. He had seen people and companies jump on new technologies on shaky ground. The fear of missing the boat to paradise had been the Achilles' heel of many and perpetuated irrational behaviors. It had plagued the human psyche for thousands of years. But for the Block Chain technology of his shoddy work, as he had called it, this hype and absurdity had gone up a whole notch. It had even been trumpeted by some as the future of mankind. *It's inconceivable. People have gone totally nuts. It's all because of my stupid folly.* He felt sunburns from within.

He had created Block Chain and the Bitcoin network to draw in young computer geeks first who had been weaned on playing computer games. *It's a money game.* He had hoped the gaming and gambling psyche would bring in these young computer gamers. Then the drug dealers and the others in the underground economy might flow in to evade law enforcement monitoring. Then the terrorists would follow suit, believing safety and secrecy had been proven of money transfers through the Bitcoin network. It's been all about luring in the terrorists and following their money trails to apprehend them and to thwart future terrorist attacks.

Seeing all this Block Chain hype, Tom was horrified. The contrite man wondered how he could possibly help remedy the unintended side effects. He had been told that was just *collateral damage*. He felt sorry for those credulous people, who were led to believe their pigs should be chained up by the Block Chain. Who thought grandfathers and grandsons should be chained up by the Black Chain. Or who put their hard-earned money and life savings into Block Chain and Bitcoin, wishing to catch the boat to paradise.

* * *

Tom was surprised that the Computer Science academics had not caught up with him. *None of them has squawked. Perhaps none of them has bothered to look into the design of Block Chain*. He knew his architectural and algorithmic design of Block Chain and Bitcoin would get a failing grade from any reputable professor of a course in *Design and Analysis of Algorithms* or in *Computational Complexity*.

The per-transaction cost should have been *O(1)*, Order 1—meaning the computational cost per transaction was upper bounded by a constant, regardless of the number of transactions processed and the size of the network. For virtually all financial systems, the per-transaction cost had been O*(1)*.

But the Bitcoin network or a network based on Block Chain had a per-transaction cost that was exploding without any upper bound.

First, the per-transaction cost related to real accounting was multiplied by the number of miner nodes. If there were 1000 miners, the associated per-transaction cost would increase by 1000 times. If there were a million miners, the associated per-transaction cost would increase by a million times.

Second, the per-transaction cost attributed to the overall proof-of-work increased with the increase in the number of miners—both for the tremendous duplications of processing and for the increased difficulty level of gaming and gambling. Furthermore, this part of the per-transaction cost also increased with the increase in the computing power of the miner machines. Any innovation of miner machines could not possibly reduce the per-transaction cost. It will rather serve to intensify the level of competition to push the per-transaction cost even higher, and to waste even more energy.

The academics in algorithm analysis and computational complexity will catch up with it sooner or later. The thought sent a chill down Tom's spinal cord. It prompted him to rethink if he could get the NSA/CIA to resurrect his original design.

Tom had handed over his authority over Bitcoin software development to the other developers in the open source development forum. He had been told by the NSA/CIA at the end of August 2010 to move on and to think about new opportunities. He had not been told then that the CIA had believed at that time it had located the number 1 most wanted terrorist by following Bitcoin money trails.

He had completed the handover process by the end of 2010.

The Bitcoins mined in the name of Satoshi Nakamoto were not his. He didn't do it for the money anyway. He doubted the government would ever take the money into its coffers. *There will be money trails*.

Tom had come up with a new idea for another peer-to-peer networking based intelligence scheme. The project had just started recently—Magnet Snare 3.0. He vowed to do it better this time, not to commit the same atrocious mistake as with Magnet Snare 2.0, even if that meant fighting harder with the NSA/CIA, possibly failing to get it fully funded and done. *No more shoddy architecture and algorithms regardless.*

ABOUT THE AUTHOR

The author received his Ph.D. from McGill University, Montreal, Canada. He held various architecture leadership positions at two of the world's largest telecommunication equipment makers, some startup companies, and a mid-sized international corporation. He played a pivotal role in the design and development of numerous products in HPC, server virtualization, and telecommunication. His extensive work in telecommunication spanned the areas of electrical and optical, as well as wireless and wireline.

The author has traveled frequently around the globe, partly for pleasure but mostly for providing technical guidance to product development at his companies' R&D sites in North America, Europe, and Asia.

The author has retired from the hi-tech industry and now lives in Canada.