The Gist of GSTN

An engine that can define the Commerce of the Future

SREE IYER

### Why this book?

Having spent about 20 plus years in Startups in Silicon Valley as an Engineer, I accidentally fell into a vat called **Systems Engineering**. This discipline of Engineering is not taught in any school – but as more and more complex devices are being built, the need for a Systems Engineer, who can look at the various aspects of a complex device or system encompassing several branches of science and math (think of Engineering as an application of science) and come up with a commercially viable solution is great. Often times, this breed of fearless warriors are thrown in the deep end of a pool and asked to swim to safety.

As I started reading up on the **Goods and Services Tax (GST)** Bill as it was being readied, I was struck by its noble ambition - being a single tax collector which will allow goods to move seamlessly across states, languages and reach the destination in a time that is a fraction of what it is today. What was even more impressive was the stated goal of the Cyber backbone that was going to run this engine – The **Goods and Services Tax Network (GSTN)**. If designed correctly, GSTN can create thousands of jobs for computer programmers who can build models, analytics, forecasting etc. so that the nation as a whole can weather droughts/ natural calamities etc. easily as all the information will be available on the touch of a button.

### Who should read this?

If you are curious about how GST is going to change your life (believe me when I say this – it will), then this book is for you. You may experience a hiccup or two while the new method takes shape but in the end it will catapult India to that of a developed country.

Many new companies will be formed to take advantage of the Data that GSTN can provide. To every new entrepreneur, this book is for you. These are jobs that are yet to be defined – and therefore you can have a head start.

Sree Iyer

Table of Contents

[Why this book? 2](#_Toc485211793)

[Who should read this? 2](#_Toc485211794)

[Table of Figures 4](#_Toc485211795)

[Introduction 5](#_Toc485211796)

[How should GSTN be designed? 6](#_Toc485211797)

[GSTN Basics 7](#_Toc485211798)

[Type of jobs needed for GSP 8](#_Toc485211799)

[Current GSTN Limitations 9](#_Toc485211800)

[What is the way out? 9](#_Toc485211801)

[Robust GSTN based on Distributed Ledger Technology 10](#_Toc485211802)

[Basics of DLT 10](#_Toc485211803)

[Step 1. Receiver places an order for a shipment 11](#_Toc485211804)

[Step 2. Sender and Receiver are assigned cryptographic keys 12](#_Toc485211805)

[Step 3. The transaction is broadcast and verified by a distributed network 13](#_Toc485211806)

[Step 4. How a Block is created 14](#_Toc485211807)

[Step 5. Now the Block is added to the chain 15](#_Toc485211808)

[Step 6. New Time Stamp added at state crossing 16](#_Toc485211809)

[Step 7. Package reaches the destination 17](#_Toc485211810)

[The complete process 18](#_Toc485211811)

[How DLT addresses the challenges 19](#_Toc485211812)

[How to tie DLT in with GSTN? 20](#_Toc485211813)

[Jobs of tomorrow 22](#_Toc485211814)

[A Site that has the best prices for a farmer 22](#_Toc485211815)

[A Services Site that can carry any amount of baggage 23](#_Toc485211816)

[A Site that can deliver analytics 24](#_Toc485211817)

[A SITE that can forecast demand 25](#_Toc485211818)

[Responding to a crisis 25](#_Toc485211819)

[Conclusion 26](#_Toc485211820)

[INDEX 27](#_Toc485211821)

# Table of Figures

[Figure 1. A simplistic implementation of GSTN 6](#_Toc485211778)

[Figure 2. Typical GSP interface for third party developers 7](#_Toc485211779)

[Figure 3. A Robust GSP application software 8](#_Toc485211780)

[Figure 4. Receiver places an order 11](file:///K:\Performance%20Gurus\2017\Books\The%20Gist%20of%20GSTN\The%20Gist%20of%20GSTN.docx#_Toc485211781)

[Figure 5. Unique cryptographic keys are assigned 12](#_Toc485211782)

[Figure 6. Transaction broadcast and verification by a distributed network 13](#_Toc485211783)

[Figure 7. A Block is created 14](#_Toc485211784)

[Figure 8. Newly created Block becomes part of a chain 15](#_Toc485211785)

[Figure 9. Border crossing time-stamped and added to the chain 16](#_Toc485211786)

[Figure 10. Package reaches the destination 17](#_Toc485211787)

[Figure 11. The Complete Process 18](#_Toc485211788)

[Figure 12. Example of forking and branching 20](#_Toc485211789)

[Figure 13. How BestPrice matches Farmer with Retailer (Buyer) 22](#_Toc485211790)

[Figure 14. Example of how a company helps move temperature sensitive medication 23](#_Toc485211791)

[Figure 15. Using Analytics from GSTN Data 24](#_Toc485211792)

# Introduction

What was the need for GST and GSTN? This excellent video by Spine[[1]](#footnote-1) describes how GST simplifies the life of a Taxpayer (think Goods manufacturer/ Service supplier) etc. by creating a single window to pay taxes. From the Tax Collector side, which typically is the State Governments through which the goods will pass, the collection is made automatic and seamless. This is a win-win situation, as long as it is conceived, designed and developed properly.

As the GST Bill starts taking shape, four different rate slabs have emerged – 5%, 12%, 18% and 28%. Essential items such as food will not be taxed[[2]](#footnote-2). Could this have been simplified to just 3 slabs, such as 5%, 10% and 25%? Because remembering three things is easy – it has been proven in social research. Also, computing tax is easier. Consider a pair of shoes for Rs. 900. If the tax were to be 10%, drop one zero and it is Rs. 90. If it is 5%, halve it after dropping a zero – Rs. 45. And if it is deemed a luxury good and taxed at 25%, it is a quarter of the price – Rs. 225. This will hopefully move pricing to be similar to that of the Metric system, meaning Rs. 10, Rs. 100, Rs. 1000 etc. It will also mean the demise of Eclairs chocolate as a means of currency, which is not a bad side effect – how many diabetics bring home the chocolate?!

### How should GSTN be designed?

A simple design of the network would just have a limited interface for Tax payers and Tax collectors. In this simplistic model, the tax payers can login to find out the amount owed on a day-by-day basis. Tax paid will be kept by GSTN until it has determined the apportionment to various collectors. Based on how robust the software is, it can even do the laborious job of filling up the 30 odd forms and send them to the appropriate departments. The collectors could be one or more (typically it will be more than one – State Excise/ Central Excise alone make up two in a transaction that moves within a single state). How long will the money stay in GSTN before it gets disbursed? This has not been answered yet. It would be ideal to have the money be moved within 48 hours or less – this will ensure that the velocity of money supply will be robust.

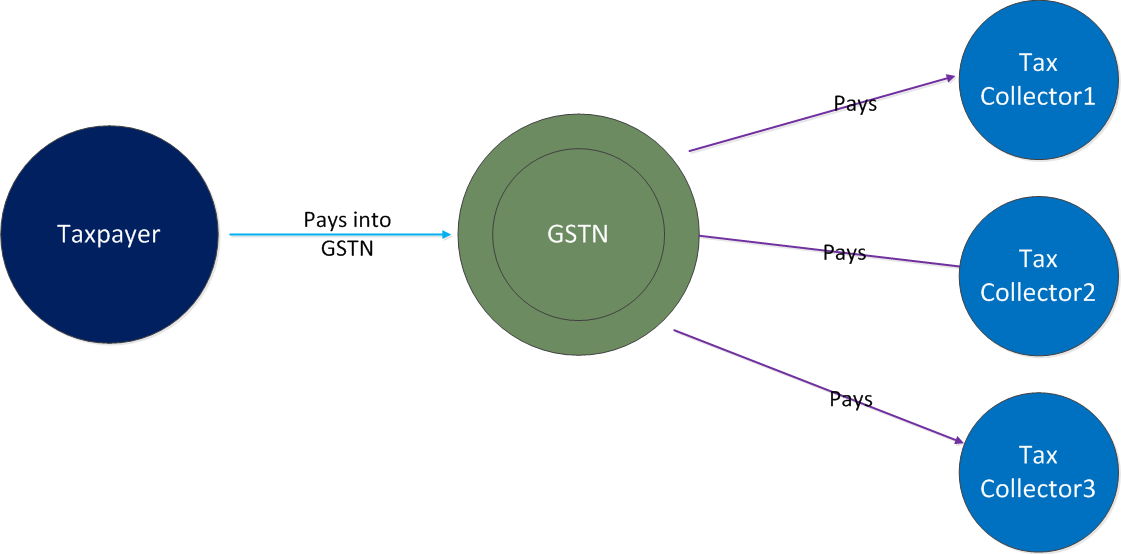


Figure . A simplistic implementation of GSTN

Since all the transactions are computerized, GST with the help of GSTN should bring corruption down significantly – in fact well implemented GSTN software can reduce corruption down to almost zero. Safeguards can be built into the GSTN software that can ensure that goods move freely and quickly across the country with minimum friction (e. g. no stops at state crossings). Some states currently are demanding that the goods be stopped at the state border[[3]](#footnote-3) and their concerns must be addressed and traffic must be allowed to proceed smoothly – otherwise it defeats the very purpose of GST. It is a well known fact that checkpoints are dens of corruption.

But is this all GSTN needs to do? With a few enhancements in the design, built into the plumbing, GSTN can be a treasure trove of data, which with orderly access can help several vertical applications (i. e. new companies) be developed and provide thousands of jobs. By orderly access I mean a method by which data is access controlled with scope defined as needed but at the same time have a robust Application Programming Interface (API) that allows third party software to access the data that GSTN servers will hold for performing various tasks.

# GSTN Basics

In its first avatar, GSTN tries to provide a simple Tax interface to tax professionals and Government departments such as the Income Tax, Central Board of Excise and Customs (CBEC) and State Tax Departments[[4]](#footnote-4).

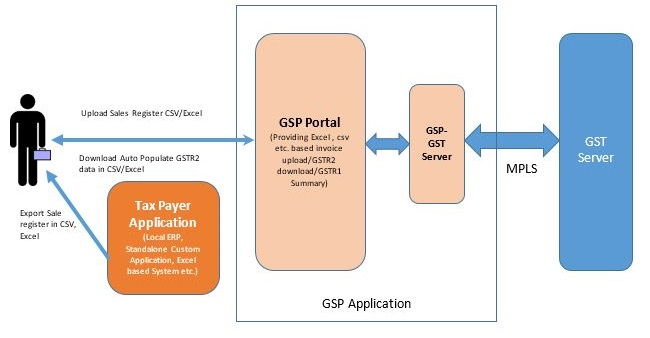


Figure 2. Typical GSP interface for third party developers

Figure 2 is a demonstration of how a third party can design their software/ application to talk to the GST Server. All such applications are grouped into a category that GSTN refers to as GST Suvidha Provider (GSP). In the above figure, the GSP application has its own server, which in turn connects to the GST Server in a Secure manner (think of it like a user buying goods from Amazon online store. An observant user will see a lock at the beginning of the site link to denote that the connection is **secure[[5]](#footnote-5)**). As a rule, I would recommend that you only visit secure sites on the web. These will start with https:// instead of http://. GSTR1, GSTR2 etc. are standard terms used to denote specific functions; GSTR1 refers to data of Outward supplies of goods and services and is a form that needs to be filled electronically. Similarly GSTR2

refers to details of Inward supplies of goods and services, also a form[[6]](#footnote-6).

While it is possible to manually fill in these forms and submit them into the GSTN servers, it is highly recommended that all vendors use computerized GSP applications to automate this process. This will minimize errors, save time and cost and also keep track of records. This alone will create a lot of jobs for programming, data entry and so on. GSTN has provided a useful slide deck[[7]](#footnote-7) that describes how the typical use cases will be for GSPs from Slide 12 onwards.

A robust GSP would be able to display on many devices such as a Laptop, a Desktop, a Tablet or a Smartphone. All of these come with their own display sizes and features and therefore the GSP software that renders data on these devices needs to be smart about the device it is talking to. Many Open Source frameworks exist today that can form a starting point for simple GSP applications. Different departments will need to look at different aspects of a company’s operations. A well designed GSP app would show the tax information to the accountant, the status of various carriers to the logistics person, information on where the raw materials are for the production person and so on. Figure 3 shows a typical GSP that would perform such functions. In this instance, all the devices use Wi-Fi network to connect to their server (GSP Server). The GSP Server itself uses any of the connectivity interfaces such as Ethernet/ Fiber/ Satellite etc. to connect securely to the GST Server. This is the preferred configuration for most GSP apps.



Figure . A Robust GSP application software

### Type of jobs needed for GSP

As mentioned before, all companies big and small will need to automate their processes. Many Open Source Software tools (meaning free) are available for this. What is required is basic programming skills in these tools. Skills that are required for these jobs are experience in client standards such as Hyper Text Markup Language V5 (HTML5)/ Cascading Style Sheets V3 (CSS3), server side tools such as Javascript, Javascript Object Notation (JSON) and a Content Management System such as WordPress, using which GSPs can be quickly put together. While text based forms will work, it is recommended that Comma Separated Variables form of Excel spreadsheets be used for storing/ submitting data. While this is a good first step, the servers should allow encrypted binary data for speed.

These skills are not just for those with a Computer Science background - anyone can get proficient within a small period of time to be able to start such apps. Note that the programming skills are in addition to core competencies such as scientists/ chemists/ physicians/ researchers/ mathematicians etc. Welcome to the jobs of the future where one has to combine a multitude of disciplines to get the best out of your career. And we are just getting started!

# Current GSTN Limitations

There is a proverb that says that one has to learn to walk first before one can run. The way GSTN API is designed today, it can at best automate the reporting of taxes by various entities to a central source for processing and disbursement of tax revenues. What are the ways this back end software can be enhanced for the e-Trade of the future? Let us look at some of the challenges:

1. Certification of the content in the goods - for instance, how will the tax calculation be made? What prevents a vendor from declaring a lower tax for transporting the goods and then tacking on a higher tax slab at the retail point? There has to be an independent checking and validation of the goods with proof of shipment from the point of origin to the destination so the two can match.
2. How does one ensure that the goods are not tampered along the way?
3. If shipment needs to be split from one source to several destinations, how does each of the split shipment get accounted?
4. The entire tax amount paid now currently gets accumulated at GSTN’s own banks, giving them a big float of money that they can arguably use for other purposes. The time money spends at GSTN waiting to be distributed should be minimized. At this time, there is no provision to ensure this.
5. New services such as Refrigerated Trucks[[8]](#footnote-8) (reefers) will become popular as the need arises to transport Pharmaceutical products, fine antiques, personal care products in addition to perishable commodities such as Vegetables, Fish etc. When these become fully operational under GSTN, a lot of logistics have to be addressed - what happens when the product being carried goes bad because of truck breakdown or road mishap or flood and other natural disasters?
6. Broadly failures due to pilferage, fraud, theft and ownership disputes are not addressed in this version.

### What is the way out?

Can these shortcomings be overcome? The above list is just a snapshot of things that can possibly go wrong and as GST is deployed, there will be many more. Looking into the future, can the GSTN Server design be enhanced such that it will last and scale for the next decade? Fortunately, there is an answer and that is to have GSTN embrace Distributed Ledger Technology into its architecture.

# Robust GSTN based on Distributed Ledger Technology

When you read the heading, you may be tempted to ask, “What exactly is Distributed Ledger Technology (DLT)? How does it make a difference in payment, clearing and settlement processes? Why should GSTN be based on DLT?” These are good questions and this chapter explains how this revolutionary technology which is a combination of components including peer-to-peer networking, distributed data storage, and cryptography that can potentially change the way in which the storage, recordkeeping, and transfer of an asset is done[[9]](#footnote-9). DLT is sometimes interchangeably used with the word **blockchain**. Keep in mind that there is website called **Blockchain.info** that keeps creating new blocks and hence we will stick with DLT to make it clear that we are talking about Distributed Ledger Technology and not about a specific company. DLT can be used in a variety of day-to-day applications such as:

* Make bank-to-bank transactions easier, faster and more efficient,
* Make land records and ownership bulletproof and wring out benami ownerships
* Track movement of goods from source to destination with no fear of pilfering / doctoring
* Using an application called Smart Contracts[[10]](#footnote-10) it is possible to completely **eliminate** the bank in a peer-to-peer transaction.
* Smart Contracts will also enable payments that can be automated upon delivery of specific, measurable targets

### Basics of DLT

The reason for the name Distributed Ledger Technology is that any transaction record that uses this technology ends up being copied and stored (or in other words *distributed*) in multiple locations. From the time a record starts till it is ended, every detail of which points it touched gets appended like a chain in block form. Every block relies on data from its previous block so tampering any data will be caught immediately. Since multiple copies exist, even successfully tampering one chain of one block will not match with its copies distributed over a wide area and hence will be caught.

Let us say there is a supplier, who we shall call **Sender**, who makes a lot of household appliances such as grinders, mixers etc. Sender is headquartered in Mumbai with their manufacturing in Thane, a suburb of Mumbai. This example is being kept simple in order to understand DLT. A distributor for Sender works out of Kolkata and caters to the requirements of West Bengal. Let us give a name for the distributor and call it Receiver. We are about to explain how a transaction, such as shipment of appliances works between these two.

##### Step 1. Receiver places an order for a shipment

Sender has to send a package when the order is placed. The terms of shipment are agreed to, such as the date of shipment and the date it will be delivered etc. For more, see Figure 4 below:

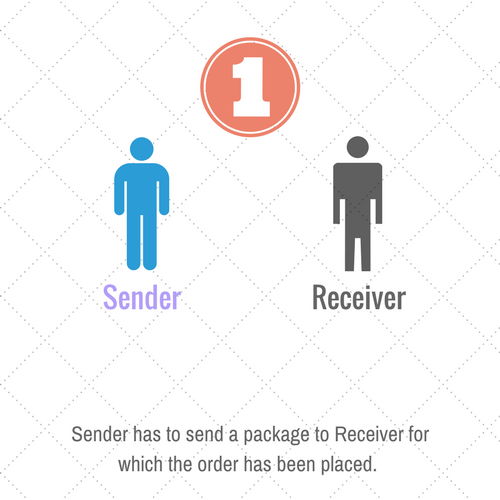


Figure . Receiver places an order

##### Step 2. Sender and Receiver are assigned cryptographic keys

In order to track the shipment, both the Sender and the Receiver must be assigned unique Cryptographic keys. Let us assume that there is a central body (such as the GSTN) that will assign this. These crypto keys are very large numbers, typically 32-bytes or more and are prime numbers with some unique properties. They come in pairs - one that you can share with everyone called as the **Public Key** and the other that you keep it with yourself called as the **Private Key**. Prime numbers are numbers that can be divisible only by the number 1 or the number itself. Examples of Prime numbers are 1, 2, 3, 5, 7, 11, 13, 17, 19 and so on. These key pairs, i. e. Public and Private have a unique property. You can use them to code and decode messages. For those interested in knowing the details, you can read up on Public Key Infrastructure (PKI). When the Sender and Receiver are assigned Key pairs, each gets a set of Public and Private keys. Both the Sender and the Receiver broadcast their Public keys to all the recipients in the transaction[[11]](#footnote-11).

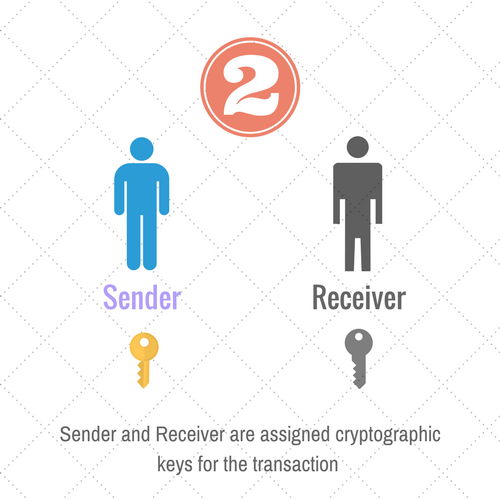


Figure . Unique cryptographic keys are assigned

In the above Figure 5, the Sender would **code** any message that he wants to convey to the Receiver using the **Receiver’s Public Key**. The Receiver will then use his **Private Key to decode** the message. No one else can decode the message except the Receiver. This concept is used every time someone buys something from an online store, such as Amazon or Flipkart. Everything happens automatically so you are spared of the details.

##### Step 3. The transaction is broadcast and verified by a distributed network

In the message that the Sender prepares, he would perhaps put the invoice of the shipment, the weight, its dimensions, value for insurance purposes etc. and time stamp it and then code the message using the Receiver’s Public Key. The message (also referred to in the Figure 6 below as a transaction) is broadcast and gets copied on to 12 different nodes on the network, who then validate it. The number of network nodes is just an illustration - it varies from application to application.

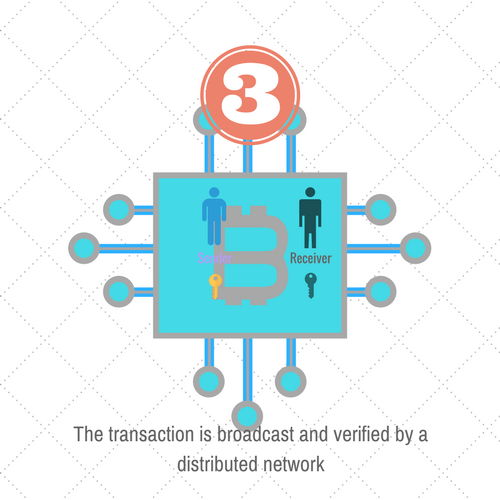


Figure . Transaction broadcast and verification by a distributed network

##### Step 4. How a Block is created

Once the data is validated / verified by the distributed network nodes, it is ready for transmission. A date and time stamp (and optionally other details) are added to create the block before it is sent to the Receiver, as shown in Figure 7 below:



Figure . A Block is created

##### Step 5. Now the Block is added to the chain

The newly created block is now added to the chain, creating a permanent **mark** in the block chain. The chain thus formed looks as follows in Figure 8.

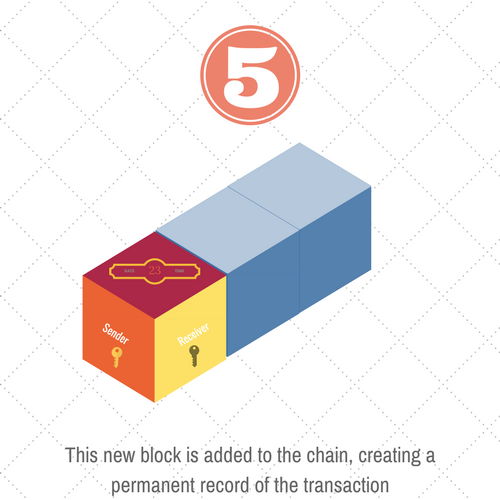


Figure . Newly created Block becomes part of a chain

As the package moves by road from Mumbai and goes towards its destination (Kolkata), it will pass through several states (one route would be through Madhya Pradesh, Jharkhand and West Bengal). Assume for the moment that at every state border, a time stamp is added to the package.

##### Step 6. New Time Stamp added at state crossing

After the package crosses into Madhya Pradesh, a time-stamp of the package is added to the existing block. The process used is called as creating a hash (or digest) from the existing block along with the newly added time stamp (shown as **Node 1**) upon crossing Madhya Pradesh. This new block is appended on to the chain as shown in Figure 9.



Figure . Border crossing time-stamped and added to the chain

The **twelve network nodes** (see Step 3 above) - not to be confused with **Node 1** above will all get updated with the newly created block and will also reflect the same exact chain. Since everything is happening digitally, there is no way to tinker/ alter the data.

##### Step 7. Package reaches the destination

Assuming that the package got time-stamped every time it crossed a border, one can expect three Nodes to be added to the original block that started out in Mumbai. Using these blocks, the Receiver can find out the exact time taken to travel between each state. Additional blocks can be created in the event of a breakdown of container or a diversion or a road block. Time stamps can be introduced into the Block as and when felt necessary. As part of the data, using GPS, the co-ordinates of the Node can also be included.



Figure . Package reaches the destination

# The complete process

From start to finish the complete process is shown in Figure 11 below:

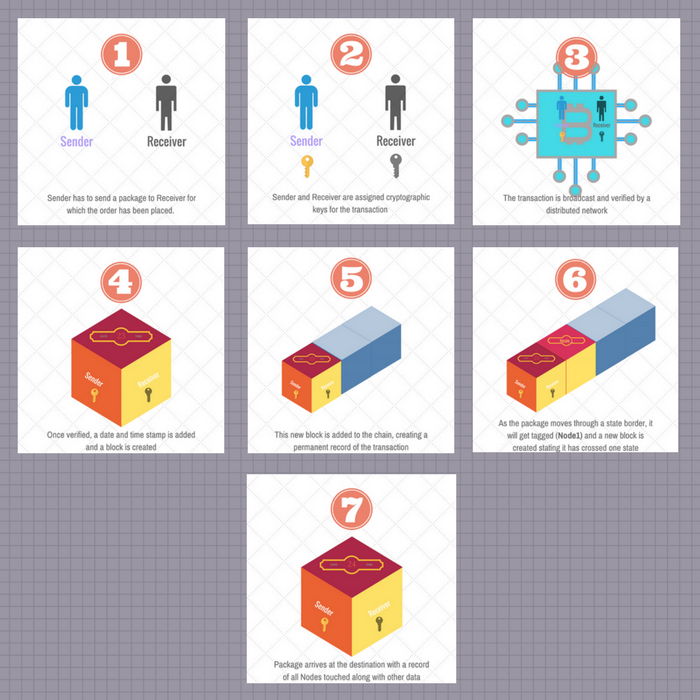


Figure . The Complete Process

Now let us re-visit the challenges in the current implementation of GSTN and how embedding DLT into it addresses these.

# How DLT addresses the challenges

The same questions are listed here along with how DLT addresses these in *italics*.

1. Certification of the content in the goods - for instance, how will the tax calculation be made? What prevents a vendor from declaring a lower tax for transporting the goods and then tacking on a higher tax slab at the retail point? There has to be an independent checking and validation of the goods with proof of shipment from the point of origin to the destination so the two can match. *Because the Sender and Receiver have an* ***exact*** *copy of what is being sent, it cannot be tampered. Since the copy of the transaction exists with several independent network nodes that are distributed throughout the county, any tampering at any stage will be caught immediately.*
2. How does one ensure that the goods are not tampered along the way?
3. If shipment needs to be split from one source to several destinations, how does each of the split shipment get accounted? *With a bit of planning, this can be achieved easily. Think of a trunk-branch situation. At the beginning of each branch, new data will be appended to the trunk and this branch data becomes a block that will traverse till the destination(s).* For more seeFigure 12. *At every fork, the original block is appended with the new destination information and date and time stamped. This way every destination knows the source from which it was hived off.*
4. The entire tax amount paid now currently gets accumulated at GSTN’s own banks, giving them a big float of money that they can arguably use for other purposes. The time money spends at GSTN waiting to be distributed should be minimized. At this time, there is no provision to ensure this. *This is where DLT really shines. On top of the DLT platform, Smart Contracts can be written such that payment will be made* ***automatically*** *to the Sender’s account upon successful delivery. There need be no intermediate holding place for money. Commissions for services rendered by say GSTN can be automatically written into the Smart Contract.*
5. New services such as Refrigerated Trucks[[12]](#footnote-12) (reefers) will become popular as the need arises to transport Pharmaceutical products, fine antiques, personal care products in addition to perishable commodities such as Vegetables, Fish etc. When these become fully operational under GSTN, a lot of logistics have to be addressed - what happens when the product being carried goes bad because of truck breakdown or road mishap or flood and other natural disasters? *Logistics industry is still finding its feet in India and DLT has built in features to allow tracking and transport of time-sensitive products.*
6. Broadly failures due to pilferage, fraud, theft and ownership disputes are not addressed in this version. *With DLT one can guarantee with 100% accuracy where discrepancies occurred and further, these* ***cannot*** *be covered up!*

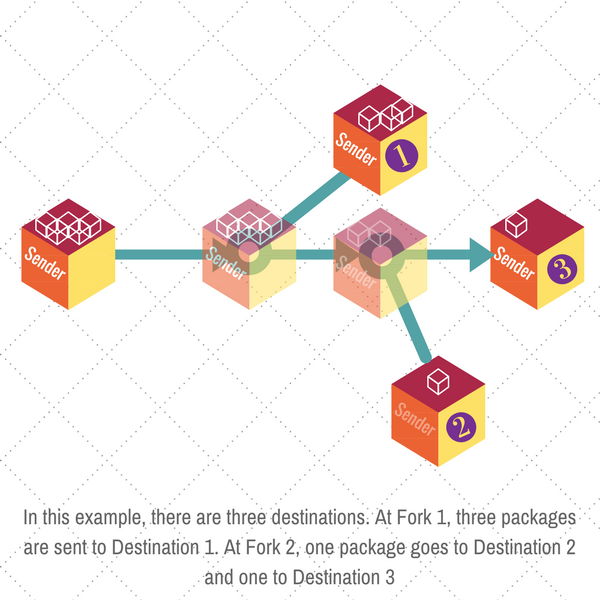


Figure . Example of forking and branching

In summary, DLT provides accurate, un-changeable record of events as they transpire. Since multiple copies exist, it is impossible to tamper with them.

### How to tie DLT in with GSTN?

DLT framework should be incorporated into GSTN Server code. The sooner this is done, the better. GST Council has already specified the different rates for Goods/ Services etc.[[13]](#footnote-13) As the API versions get rolled out, a DLT enabled framework along with sample code on how to tap into the information contained in the DLT ledgers can be enabled. It would have been nice to have this from the get go but DLT is compute resource intensive and therefore it needs to be eased in.

Distributed Ledgers can be designed in many ways – it can be Permissionless (meaning open for all) such as the **Bitcoin**. This would mean that a copy of the block can rest with anyone. However in this case, the data is of a sensitive nature and therefore it needs to be Permissioned. This means the network nodes that will hold copies are carefully chosen and this information is kept Private. This makes the data safe, even if it **physically resides** outside the country! Why? Because it is encrypted and can only be opened by the rightful owner.

The most significant advantage from using DLT is that most of the infrastructure is **Open Source** (meaning free), which can be downloaded and enhanced. Chain.com is one company which has specialized components[[14]](#footnote-14), all of which are Open Source and can be downloaded and tested before deployment.

This Whitepaper titled **Chain Protocol Whitepaper[[15]](#footnote-15)** is an excellent starting point on how to build the Chain core that can form the foundation of DLT in an organization. A quick look at how Chain has designed the ledger shows that they have addressed all the requirements that an entity such as the GSTN would need. For those who are new to DLT, it is suggested that they go through the Whitepaper and understand the underlying concepts. Here is an excerpt from this Whitepaper:

*In this paper, we present the Chain Protocol: a design for a shared, multi-asset, cryptographic ledger. It supports the coexistence and interoperability of multiple independent networks, with different operators, sharing a common format and capabilities. Using the principle of least authority, control over assets is separated from control over ledger synchronization.*

*The Chain Protocol allows any network participant to define and issue assets by writing custom “issuance programs.” Once issued, units of an asset are controlled by “control programs.” These programs are expressed in a flexible and Turing-complete programming language that can be used to build sophisticated smart contracts.*

*Each network is secured by a federation of “block signers.” The system is secure against forks as long as a quorum of block signers follows the protocol. For efficiency, block creation is delegated to a single “block generator.” Any node on the network can validate blocks and submit transactions to the network.*

*Chain Core is an enterprise software product that implements the Chain Protocol. An open-source*[*developer edition*](https://github.com/chain/chain)*is freely available, and Chain operates a Chain blockchain network as a freely accessible testnet.*

*In the*[*second section*](https://chain.com/docs/1.2/protocol/papers/whitepaper#2-motivation)*, we explain the background and motivation for the design of the protocol.*

*Next, we describe key concepts —*[*assets, transactions, blocks*](https://chain.com/docs/1.2/protocol/papers/whitepaper#3-data-model)*,*[*programs*](https://chain.com/docs/1.2/protocol/papers/whitepaper#4-programs)*and*[*consensus protocol*](https://chain.com/docs/1.2/protocol/papers/whitepaper#5-consensus)*— and discuss how programmed rules are enforced and double-spending is prevented.*

*Finally, we discuss various aspects of the blockchain network:*[*security*](https://chain.com/docs/1.2/protocol/papers/whitepaper#6-security)*,*[*scalability*](https://chain.com/docs/1.2/protocol/papers/whitepaper#7-scalability)*,*[*extensibility*](https://chain.com/docs/1.2/protocol/papers/whitepaper#8-extensibility)*and*[*interoperability with other networks*](https://chain.com/docs/1.2/protocol/papers/whitepaper#9-interoperability)*.*

Chain.com’s Chain Core is an excellent starting point to understand the capabilities and power of DLT. By the way the word **blockchain** used in this site and DLT are one and the same.

So what kind of jobs will a GSTN fortified with DLT have?

# Jobs of tomorrow

### A Site that has the best prices for a farmer

**BestPrice.com** is a one stop shop for Farmers looking for the best price for their produce – be it grains or pulses. BestPrice is designed to be able to locate the best price for a producer by matching the buyer who needs it the most with the farmer. How does BestPrice do it? It tracks the movement of various commodities across the country by monitoring the data from GSTN Server and knows the places that need the commodity the most. Demand varies for a variety of reasons and the software uses data from GSTN server to find out the best fit.

The Smartphone app which is a service that a farmer can sign up for will ping the farmer once it has found a match. The farmer now needs to know where to drop off the product. BestPrice will work with a fleet of Container companies to figure out the best route for the goods. Think of Google Map telling the farmer the route his product is going to take to reach the destination.

Just like getting a UPS/ FedEx Tracking number, BestPrice will provide both the Buyer and the Seller (farmer) a tracking number on the status of the shipment. BestPrice has the access rights to get data from GSTN for a limited set of items, say commodities. As soon as it is delivered, the farmer as well as the Container companies will be paid using a Smart Contract, with no money wasted on bribes/ commissions or kickbacks. Additionally BestPrice can provide advice for the farmer on the fastest return on investment – for instance, a shipment that takes 2 days versus a shipment that takes 2 weeks. Based on what the farmer’s liquidity requirements are, the farmer can make an informed decision. For more, see Figure 13 below:

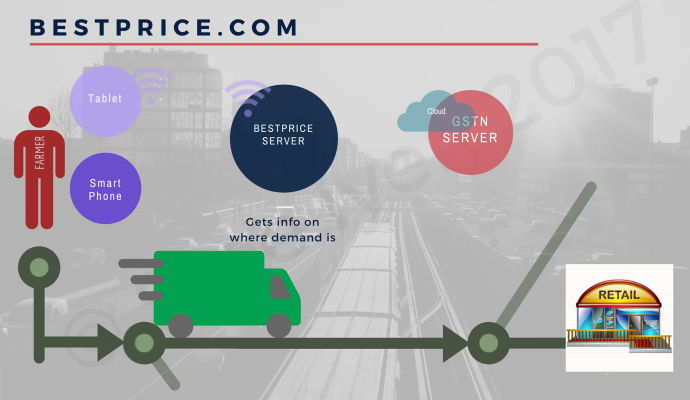


Figure . How BestPrice matches Farmer with Retailer (Buyer)

This is not limited to commodities – the concept can be extended for other goods and services too. The onset of a heat wave in a certain region would spur sales of fans and air-conditioners which can result in a great selling opportunity for a company with inventory.

### A Services Site that can carry any amount of baggage

**WeMoveIt.com** has a focused mission – to help transport frozen and climatically sensitive goods all over India. To this end, this company constantly gets updates from GSTN server on travel times to various destinations and combines that with data obtained from various Reefer (Refrigerated Mover) companies on their load capacity usage. When a shipment of temperature sensitive medication has to be moved from a Pharma company to a Hospital, the Pharma company can look up WeMoveIt site for the fastest and cheapest way to get it to the destination. Data is located in at least two places in this instance, the latest commute times from the GSTN and the availability of Reefer trucks by looking at various Reefer truck sites. Figure 14 below helps explain this process:

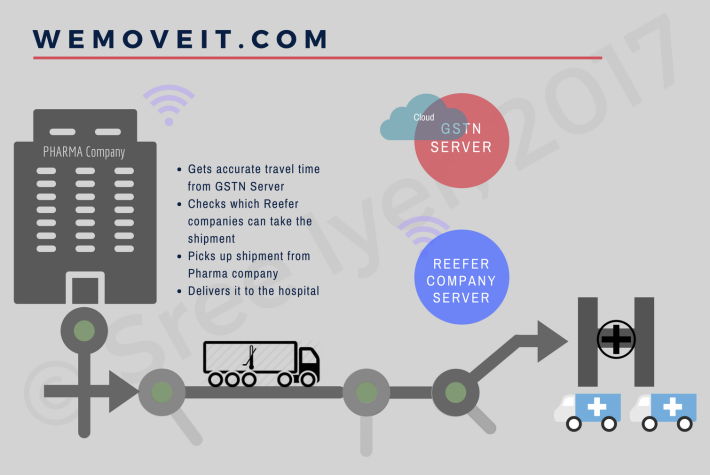


Figure . Example of how a company helps move temperature sensitive medication

In this instance, the Reefer company has to go to the Pharma Company’s delivery area in order to transfer the medication. If say a combination of Reefer trucks is needed to reach it to the destination, the Service company (in this case WeMoveIt) would stamp via DLT details of the Goods (medication in this case) as well as a service (of transporting it from the Pharma company to the Hospital). WeMoveIt will handle all the paperwork of reporting these to the GSTN. WeMoveIt could work as either a subscriber model or a pay as you go model for generating revenue.

### A Site that can deliver analytics

GSTN running with DLT will generate a gold mine of data which can then be parsed and monetized. Consider this for an example – **TyxRUs.com** (short for Analytics Are Us) helps its subscribers enter a category (Consumables) or a specific item (Rice, Car) and will publish all there is to know about it.

Analytics may turn up interesting information – Most number of Used cars in Delhi are bought during the month of July. The reasoning could be that most families relocate to Delhi around the month of June-July to synchronize with the school year and therefore maybe looking for used cars. But if Dharwar has peak demand for shoes in the month of May, that might not be obvious. A little deeper digging might reveal that Dharwar is a College Town and also receives a lot of rains from the end of May for about 3 months and so the population maybe preparing itself for the twin events.

A Bank looking to find out where to open its next branch maybe able to do so by using TyxRUs software. All it has to do is locate places which do a lot of business but are underserved in terms of banks.

In not too distant a future, Elastic Employment may become common. Consider this example – Banks do the most business on Mondays, Wednesdays and Fridays and also around the middle of a month and the end of a month. Banks can then add more tellers/ cashiers on the busy days only. Many stay at home Moms/ Dads may find this kind of a flexible arrangement useful.

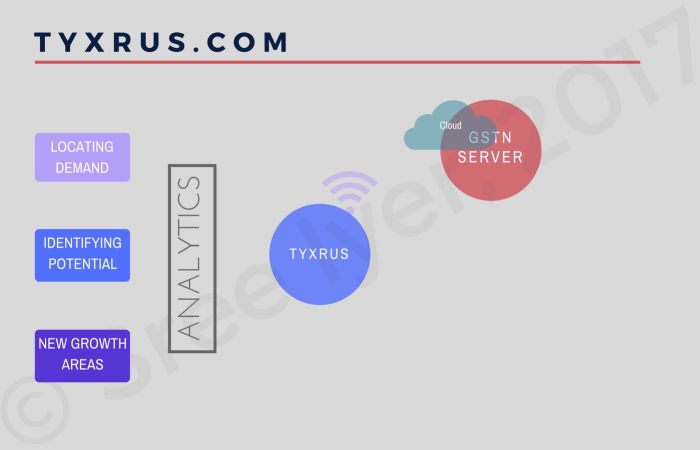


Figure . Using Analytics from GSTN Data

### A SITE that can forecast demand

This is a more long term scenario. As GSTN accumulates data over the years, sites can start analyzing the data patterns for being able to accurately forecast where the demand lies. For long gestation products such as town planning, infrastructure development (roads, railways, airports) it is important to be able to spot the trends and plan accordingly.

One of the biggest shortcomings in India is that by the time a four lane access road is completed (delays typically due to approvals/ congestion etc.) the traffic needs have already caught up with the **additional lanes** provided. In other words, planning is always coming up short and is out of sync with the needs. Perhaps lack of actionable data is also a reason. However this is not a problem limited to India – Japan had an even bigger challenge as it *just does not have enough*space for roads! How did Japan solve the problem?

By creating a multi-level subway system, especially in the urban and suburban areas. Tunneling is considered an expensive proposition but by creating private entities to manage its rail systems and allowing these entities to own/ manage real estate from which they could fund the railway development, Japan has created a viable transportation system that crisscrosses the length and breadth of the country.

### Responding to a crisis

How can GSTN help in the event of a National calamity? Relief organizations/ local governments can look at GSTN data to find out how to quickly transport the essential supplies such as food, clothing, shelter and essential appliances. They can also locate and assign specific drop off points all over the country so the donations do not overwhelm the system.

# Conclusion

GST presents a very big step in the evolution of India. For the first time, the data about the entire country is going to be housed under one umbrella. The back end of GST, GSTN has a great responsibility on this shoulders – it has to ensure that critical data is protected while still be able to provide an API that allows software companies to mine its data for useful applications.

DLT is proven to be robust and has survived threat attacks (Bitcoin is an application based on DLT). Building in DLT into the GSTN is not only a good move, it is a must if India is to rid itself of Corruption that plagues it at every level.

As more details on GSTN emerge, this document will be updated. For any comments/ questions, please write to me at **sree@sreeiyer.com**.

# INDEX

A

API · 6, 9, 20, 26

C

CBEC · 7

D

Distributed Ledger Technology · 3, 9, 10

DLT · 3, 10, 18, 19, 20, 21, 23, 24, 26

E

entrepreneur · 2

G

Goods · 2, 5, 20, 23

**Goods and Services Tax** · 2

**Goods and Services Tax Network** · 2

GSP · 3, 4, 7, 8

**GST** · 2, 5, 6, 7, 8, 9, 20, 26

GST Suvidha Provider · 7

**GSTN** · 1, 2, 3, 4, 5, 6, 7, 9, 10, 12, 18, 19, 20, 21, 22, 23, 24, 25, 26

I

India · 2, 5, 19, 20, 23, 25, 26

M

manufacturer · 5

N

**Node** · 16, 17

O

Open Source · 7, 8, 21

R

Receiver · 3, 4, 10, 11, 12, 13, 14, 17, 19

Refrigerated Trucks · 9, 19

S

**Sender** · 3, 10, 11, 12, 13, 19

Service · 5, 23

State Governments · 5

supplier · 5, 10

**Systems Engineering** · 2

T

Tax Collector · 5

Tax collectors · 6

Tax payers · 6

Taxpayer · 5

1. [*What is GSTN*](https://youtu.be/g6l0pLjV4lE) A five minute **YouTube** video by **Spine** that explains the big picture of what GSTN does [↑](#footnote-ref-1)
2. [*Final GST rates out*](http://timesofindia.indiatimes.com/business/india-business/Final-GST-rates-out-slabs-fixed-at-5-12-18-28/articleshow/55225847.cms)**Times of India** post dated Nov 3, 2016 [↑](#footnote-ref-2)
3. [*E-Permits: A Bottleneck in GST Implementation*](https://www.pgurus.com/e-permits-a-bottleneck-in-gst-implementation/) – **PGurus.com** dated February 26, 2017 [↑](#footnote-ref-3)
4. [*GST Suvidha Provider (GSP)*](http://www.gstn.org/ecosystem/index.php) **GSTN Website** - Front end design [↑](#footnote-ref-4)
5. [*Amazon.com*](https://www.amazon.com/) Link and sign of a **secure connection** [↑](#footnote-ref-5)
6. [*GSTR1, GSTR2*](https://gst.caknowledge.in/key-components-gstr-1-gstr-2-gstr-3/)Explanation of various forms used in **reporting GST data** [↑](#footnote-ref-6)
7. [*GSTN Use cases*](http://www.gstn.org/ecosystem/pdf/GSP_Implementation_Framework_V_3.0.pdf) **GSP Implementation Framework** [↑](#footnote-ref-7)
8. [*Refrigerated trucks haul more than just perishables*](http://www.inboundlogistics.com/cms/article/refrigerated-trucks-haul-more-than-just-perishables/) by Tom Kampf [↑](#footnote-ref-8)
9. [*Distributed Ledger Technology*](https://www.federalreserve.gov/econresdata/feds/2016/files/2016095pap.pdf) – **Distributed Ledger Technology** by the US Federal Reserve [↑](#footnote-ref-9)
10. [*Distributed Ledger Technology beyond Blockchain*](https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/492972/gs-16-1-distributed-ledger-technology.pdf) - A **White Paper** by UK’s Government Office for Science [↑](#footnote-ref-10)
11. [*Public Key cryptography for block chain*](http://www.cfbtranslations.com/bitcoin-part-3-hashes-public-key-cryptography-for-dummies-and-the-block-chain/) - **Hashes and PKI for block chain** [↑](#footnote-ref-11)
12. [*Refrigerated trucks haul more than just perishables*](http://www.inboundlogistics.com/cms/article/refrigerated-trucks-haul-more-than-just-perishables/) by Tom Kampf [↑](#footnote-ref-12)
13. [*GST India Download Page of Rates*](http://www.gstindia.com/category/download/) *–* **GST Free Tax Site** [↑](#footnote-ref-13)
14. [*Chain.com video*](https://youtu.be/bK6wHW1K9jM) on how Block chain **is faster, scalable and cheaper** [↑](#footnote-ref-14)
15. [*Chain Protocol Whitepaper*](https://chain.com/docs/1.2/protocol/papers/whitepaper) by Chain. com [↑](#footnote-ref-15)