PARAM ʎ Blockchain for Commerce

Vaideeswaran Sethuraman
Creator of Param
Divum Labs Pvt. Ltd.
Bangalore, India
vaidee@param.network

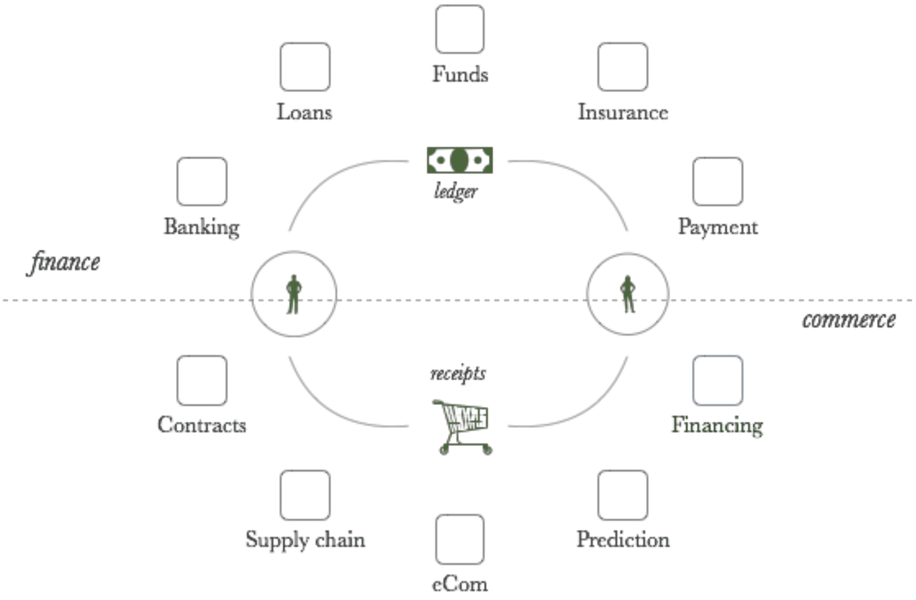
***Abstract*—**The financial system records a series of transactions in a ledger. Each financial institute holds individual's accounts & records their transactions independently. Later Bitcoin, built on Blockchain technology has reinvented the financial system by introducing a distributed ledger mechanism. On the other hand, the commerce system, the enabler of the financial system, records series of transactions against the purchase of goods or service. The proof of sales, receipts are necessary to claim the validity of contracts associated with the purchase. Today, all underlying finance systems are developing in isolation from the commerce system, and they need to be unified. The combined platform with the knowledge of finance & commerce is necessary to drive more intelligence-driven trade. In this paper, we will see how to apply a new blockchain technology to build next-generation commerce system using a new protocol – "PARAM," to record digital receipts on-chain, construct knowledge graphs and build data-driven advanced decentralized applications.

Keywords—Blockchain, commerce, digital receipts, cryptocurrency, decentralized payment gateway, proof-of-purchase, commerce knowledge graph

I. Introduction

Traditionally, the fin-tech related to banking is centered around bringing convenience around payments, and thereby maintaining authentic ledgers of transactions between parties. Even the modern prevailing blockchain protocols seem to be evolve in a similar paradigm. However, in today's digital economy, the systems need to develop beyond maintaining ledgers to digitize the facts and contracts associated with the transactions. While few blockchain protocols are supporting smart-contracts, they constrain themselves by storing the values as ledgers.

Fig1.0: Finance & Commerce ecosystem with sample applications



After the invention of the wheel, the world became a marketplace where goods are transported easily and traded, evolving into modern-day commerce. To facilitate the ease of trade, the exchange of value has developed from a barter system to modern finance. Though business enabled finance systems, it matured at its own pace into an integral part of all the existing systems. As shown in Fig1.0, these systems exist in isolation. To explain, let's consider a transaction that happened through a credit card and later converted to monthly installments. Now If we have to isolate out the products that were bought via EMI, there exists no way to get the details. In case, the product is covered under warranty and if it stops working, should the consumer still pay the installments or get the paid amount as a refund. Since the commerce and finance system is in isolation, the quality of service to the consumer is limited. *We imagine a future where AI based customer service agents will make decisions based on an array of data that the customers purchase history provides, the current infrastructure doesn't allow for it unless you're a behemoth like Amazon.

While blockchain reinvented banking, the commerce applications are developed as a workaround within the existing blockchain infrastructure in the form of decentralized apps. Though Dapps can solve a few use cases of commerce, it will not provide a full-fledged system. With many Dapps experimenting with primary use cases, the data is getting fragmented, and it is available in multiple places, accessible to only the owner of the contracts. Imagine how the ethereum evolved from bitcoin by providing access to entire ledger plus storage to unleash the power of smart-contracts, the same way the new commerce systems giving access to particulars of the transactions a.k.a digital receipts legitimately across applications can unleash a new wave of commerce applications on the blockchain.

Original Equipment Manufactures (OEM) could digitize their warranties on smart contracts assuring end-consumers of a high quality of service (QoS) irrespective of the sales channel. The above example is possible if and only if (a) they could establish a universal product catalog which any merchant can link the sale to an authentic product digitally (b) they could validate the proof-of-purchase against legitimate payment (c) the consumer is capable of showing proof-of-ownership. In today's commerce, very established brands ensure the above through authorized dealers & service centers offline. Blockchain technologies can be wisely used to create a trusted ecosystem with the same level of QoS across OEMs, merchants, and channels.

As we are leading to more and more Artificial Intelligence driven world, it is crucial to model consumer behaviors as a whole. Today consumers are shaped independently based on social behavior, digital interactions, commerce interaction on a specific portal. Modeling such complex interactions can be very challenging with the shortage of data and systems that allow churning such data. Commerce data centered around a subject provides an excellent opportunity to model 360-degree view of the matter. Better data points like income, shopping, and savings, shows the purchasing power, preferences towards food, fashion, investments, hobbies, entertainments and more. Using blockchain technology, we can achieve intelligent decision making with ultimate control of data to different parties involved such as buyers, sellers, and OEMs.

All the scenarios narrated in this chapter are legitimately addressed by PARAM protocol, designed for advanced commerce on the blockchain. The protocol aims to democratize finance and commerce systems for all and open up a platform for innovative applications. This paper also introduces decentralized payment gateway solution to drive global adoption of param with minimal friction. As most modern-day commerce functions on global fiat currency, it is necessary for param to be interoperable. The gateway will be a crucial enabler to bridge the fiat- world and the crypto-world of future.

II. Param Protocol

The solution architecture of the PARAM protocol comprises of (a) *store of value as param* (b) *store of particulars of the transaction as digital receipts against payments* (c) *building commerce knowledge graph (comGraph) comprising of catalogue, sellers, buyers & purchase details* (d) *provide a platform to construct advanced decentralized apps using comGraph*.

*A. **Store of Value (param)***

The core of the protocol is a blockchain where the trust, immutability, and anonymity is all built in by default. The blockchain implementation is similar to ethereum, where the data is hashed in a modified Merkle Patricia tree, and the updated state is stored in the levelDb. The basic cryptocurrency is "param". Maintaining the state of the param blockchain will involve mining.

*B. **Store of Digital Receipts as proof-of-purchase***

In real-world, sellers[s] and buyers[b] will be involved to complete a commerce transaction, and a typical transaction can be subdivided into the five-step process as depicted in fig2.0:

- The transaction is initiated by seller sending a *proposal[p]* to buyer
- Buyer agrees to the same by placing a *purchase order[po]* back to seller
- Once the transaction criteria is met, seller sends an *invoice[i]* against the *po* to buyer to initiate the payment process
- Buyer makes a *payment* once the items listed in the invoice has been received
- Finally, sellers shares a *digital receipt* as proof of transaction to the buyer.

The proof-of-purchase can be visualised as digital receipt against legitimate payments.

proposal[p] -> purchase order[po] -> invoice[i] -> payment[p] -> digital receipt[dr]

Fig2.0: Five step process involved in commerce transaction

A dictionary definition says receipts are the fact of its being received. They are the universal truth of the commerce transaction, which contains the list of items or service headers that were part of the transaction between buyer and seller. To store digital receipts authentically and to increase fault tolerance (explained later) on the blockchain, the param allows recording of all the five states of the transactions as and when they occur. In a typical B2C transaction, this may appear to be instantaneous five-step processes; however in B2B scenarios, these steps may be well spread across time, which opens up many interesting use-cases for financing and invoice discounting applications to build over the protocol states. As every transaction state change would result in an update to the blockchain, this would once again involve mining. Therefore, one commerce transaction would result in five state updates (mining) in the blockchain.

```
<script type='application/ld+json'>
{
  "@context": "http://www.schema.org",
  "@type": "product",
  "brand": "aliceNbob",
  "name": "paramPhone",
  "image": "http://aliceNbob.com/images/
paramPhone.png",
  "description": "Imaginary product used for illustration",
 "catalogue": {
    "@type": "catalogue",
    "sku": “s00001",
    “mfgId”: “PP007”,
    "reviewCount": "100000"
  }
}
 </script>
```

```
<script type='application/ld+json'>
[
{
  "@context": "http://www.schema.org",
  "@type": "receiptItem",
  "sku": "s00001",
  "quantity": “5",
  "maxPrice": “1121",
  "salesPrice": "1020"
 }
},
…,
{
  "@context": "http://www.schema.org",
  "@type": "receiptItem",
  "sku": "s00009",
  "quantity": "1",
  "maxPrice": “33",
  "salesPrice": "33"
 }
}
]
 </script>
```

Fig3.0: Extended JSON-LD structure for storing catalogue information

To store digital receipts on the chain, the merkle-patricia tree is modified to include a new value type (*ldtype*), which allow storing the receipts in JSON-LD [3] format. The consensus to store receipts on the blockchain involve validating JSON-LD schema and structure. Fig3.0 depicts the JSON-LD schema for a typical product definition and digital receipt structure, as extended by PARAM. This structure excludes typical transaction information that stored on the blockchain, such as the buyer & seller id, total value & currency involved in the transaction. This ensures utmost security by providing pseudo anonymity features to the parties involved in the transaction.

C. ***Construction of Knowledge Graph***

As discussed in the previous chapter, today’s commerce heavily relies on the data for improving both its day-to-day operations and to enhance the customer experience with the aid of artificial intelligence. Therefore, PARAM being commerce centered protocol, needs to provide an excellent framework to do data-mining of all the data present on the blockchain to support advanced applications.

Though levelDB is highly scalable, it has its limitations when the context switches to its querying capabilities. levelDB stores all the keys and values as byte arrays making it difficult to query. Even though we store the json, it will be saved as a byte array, making it harder for queries. The difficulty of the processing is directly proportional to the complexity of the query.

We can overcome the above challenge by constructing one of the largest commerce based knowledge graph (comGraph) derived from every single transaction that is executed on the PARAM network. The comGraph exists in parallel to blockchain data that stored in levelDB. The unique network topology of the protocol, allows miners to optimally store the relevant data depending on the role chosen, as explained in the next chapter.

Google's work on this space inspires our work on knowledge graph, PARAM aims to contribute a new receipt type schema to schema.org. The purpose of constructing this graph has got a two-fold agenda

- The transactions are well structured on-chain for querying for any kind of data
- To aid building advanced applications in the space of machine learning and artificial intelligence over the decentralised network.

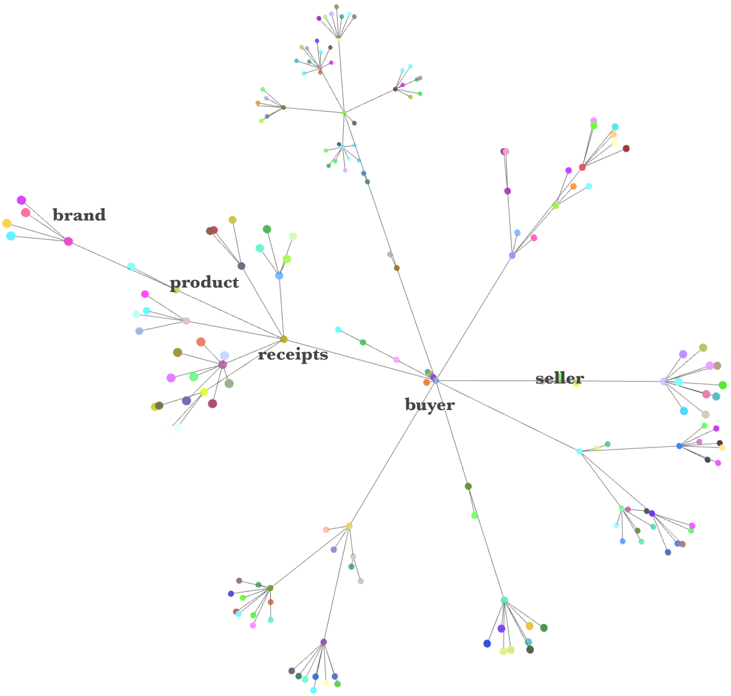


Fig4.0: Knowledge Graph on the PARAM blockchain

The underlying graph database is stored using the triples or the Subject-Object Predicate (SOP) model. Triple-stores can offer generic schema definitions and also the capabilities for semantic querying. Linked-Data (JSON-LD) combined with SOP graph model leads to excellent knowledge graph on the PARAM chain.

D. ***Graph based decentralised applications (gDAPP)***

gDAPPs are advanced smart-contracts that can be built on PARAM. The major contribution over the other popular blockchain EVMs is (a) Support for multiple languages such as Solidity, JavaScript, python (b) support for GraphQL like querying data on-chain (c) Support for building & exercising ML models by allowing apps to leverage the anonymous data present on the network. All of the above features gives an unparalleled opportunity to build powerful commerce applications using gDAPPs.

digitalReceipts (dr) = JsonLinkedData (Products x global catalogue)

comGraph = knowledgeGraph (digitalReceipts x sellers x buyers)

gDAPPs = function (comGraph x contract-storage x accounts-storage)

More examples of the kind of applications that can be built using gDAPP is in the next chapter.

III. Param Network Architecture

The high-level architecture of the network consists of three layers with the blockchain at the core to build trust and immutability on the data. The chain holds all the truth on all types of transactions. Once the miners confirm the transaction, the details are written to the data pool, from where the applications can access the data. The fig5.0 depicts the same.

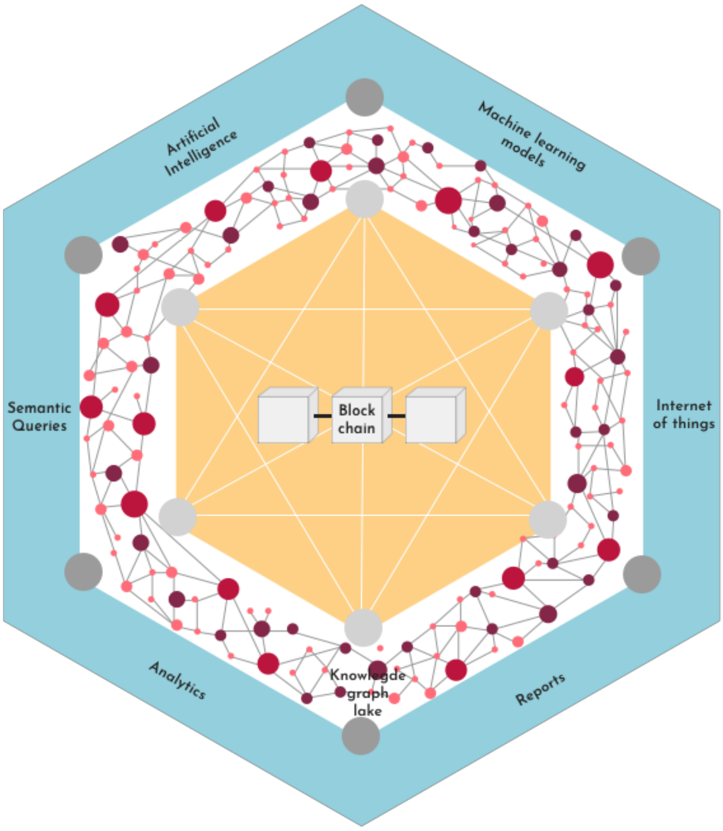


Fig5.0: PARAM Network Architecture

A. ***Blockchain Layer***

The core of the protocol is a blockchain where the trust, immutability, and anonymity is all built in by default. The levelDB holds the state changes and the chain data along with the transaction data. The Merkle Patricia tree is modified to keep the details of the transaction (digital receipts), in addition to the value.

The layer is responsible for (a) accepting transactions on the network (b) providing infrastructure for running smart contracts (c) predictive GAS computation (d) establishing a consensus of the five-step transaction process (e) defining mining algorithms and (e) finally, providing storage layer for all chain data.

The protocol imposes stringent requirements to provide maximum fault tolerance such as, the seller need to pay a GAS fee in advance for five-step transactions. In a pseudo-anonymous network, this ensures buyers aren't burdened with both GAS fees and malicious proposals thrown against their account. There is also a contemplation on supporting Proof-of-Stake protocol to waive off the seller's GAS fee based on seller's stake on the network. This layer is fully compatible with supporting token based gDAPPs to be built on top.

B. ***Knowledge layer***

Once the transaction is confirmed by the miners and written to the block, the transaction details are picked up by special nodes, responsible for converting the details into the semantic graph. The semantics is written to the data pool when the selected nodes vote in favor of the transaction. A strict consistency will is imposed before the next set of data is uploaded into the pool.

This layer is responsible for providing graph compute engine to (a) validate semantics of the JSON-LD embedded into the transaction (b) construct the graph using SPO (subject-predicate-object) format upon validation (c) reliable graph data pool storage (d) indexing and querying infrastructure

To validate JSON-LD semantics, the network randomly selects nodes to dissect the JSON-LD embedded into the transaction and vote on the validity of the semantic data. This voting process is an integral part of the protocol's core consensus algorithm. Upon validating, the data is converted it into an SPO (subject predicate object) format, which when verified by the selected nodes is added into the graph data pool.

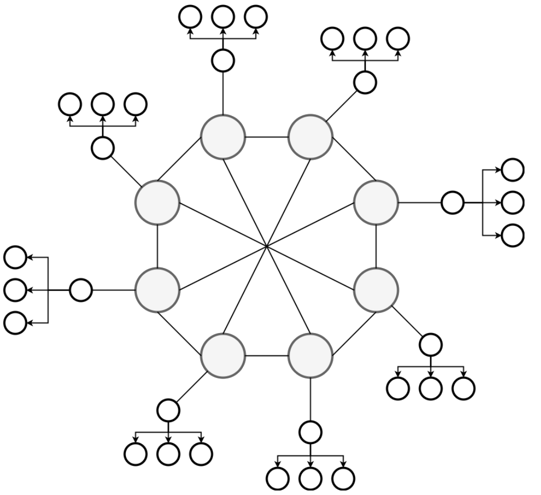


Fig6.0: Knowledge layer rearranged for Graph Query

To provide decentralized graph storage, The knowledge graph is sharded across different nodes, closely following the principles of Hadoop. The comGraph will be partitioned keeping a set of sellers as the centre of the graph, and multiple sub-graphs are formed in parallel and distributed onto the network with enough redundancy. The seller is chosen as the core to shard the graph as they hold the maximum nodes connected to them. The size of the seller set per node is directly proportional to the number of edges connected to them.

To provide decentralized graph network, different nodes are selected to play different roles by the system based on the network need & their capacity. Different node types are (a) graph storage/worker nodes (b) graph indexing node (c) graph query master node. Typical query flow would be the chosen master node on the network will intercept the queries submitted by EVM, index node identifies the worker nodes for master, and worker nodes executes the 'map' query, and finally, master node reduces the map response to derive a final answer for the smart-contract under execution.

C. ***Application layer***

The application layer helps in maintaining the engines responsible for manipulating and deriving meaning from the knowledge data pool. Any third-party developers can connect their application to the data pool and perform analytical or deep learning algorithms on the available data.

D. **PARAM potential applications at a glance**

- On this platform, building applications such as digitizing the contracts related to a transaction such as warranties & insurances become trivial. The brand owner can now publish a public commitment that any buyer purchasing its product (as identified by JSON- LD) through any seller, would ensure the eligible contracts are automatically executed. Thus, the buyer is organized when a purchase is made using this infrastructure.
- Building solutions like C2C marketplace application need solutions for (a) claiming ownership of a product under resale and (b) executing a change of ownership. The store of digital receipts allows the authentic buyer to prove on the platform and by merely initiating the five-step transaction process, the owner can quickly issue a digital receipt on the platform to transfer ownership.
- The Commerce data centered around a subject provides an excellent opportunity to model a 360-degree view of the matter. Right from the income, shopping to savings, shows the purchasing power, preferences towards food, fashion, investments, hobbies, entertainments and more. Over and above, what makes this an unparalleled protocol is the power of intelligence layer that can be built on top. The gDAPPs can contribute new dimensions to knowledge graph outside the commerce eco-system and exploit the network infrastructure to look for hidden patterns in the knowledge base that can help business serve their customers better and operate at a better efficiency.
- Imagine what if you can throw an accurate recommendation to an anonymous user logged into the application. The above is possible using this protocol because the user on the network is not unknown to the network, though he opted in the application context. The system now bridges the gap by merely providing the model to the application instead of data associated with it.
- Building micro-insurance application for cab travel. The app can allow a cab booking engine to issue a proposal at the start of the booking process, which triggers the issue of micro-insurance valid until the journey ends. The end of trip can be identified by the successful payment.

  To conclude, PARAM protocol aims to bring a fair marketplace by providing an opportunity to small-size to large-size commerce business owners/application developers to ensure the same level of Quality of Service (QoS) through gDAPPs as behemoths like Apple. Thus, eliminating the need for every seller to be an established/identified player in the market. Instead, their digitized contracts will speak for their assured promises. The protocol also democratizes the AI by providing access to big-data & much-needed infrastructure affordable to all equally.

IV. Decentralised Payment Gateway

To drive adoption of the blockchain , the inbuilt decentralized payment gateway acts as a bridge to support multiple payment currencies natively on the PARAM blockchain, enabling payments to be made for any commerce transactions on the chain with the fiat and other cryptocurrencies outside the chain. Also, external commerce applications (non-blockchain) can also post their receipts onto PARAM blockchain, without compromising the proof-of-purchase requirements.

To support fiat transactions complete key-store management, wallets and thin gateway client is created for commerce merchants and financial institutes to adopt and integrate quickly. This deeper integration is needed as the core protocol cares for recording digital receipts against legitimate payments strictly.

Similar infrastructure can be set up to support popular cryptocurrencies with decentralized exchange approach, which is discussed later.

Acknowledgment

We like to acknowledge the param development community for validation.

References

Some of the references are highlighted and few might have been ignored unintentionally.

(1) S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.
(2) Ethereum.org
(3) JSON-LD
(4) Knowledge Graph
(5) Commerce Systems
(6) dGraph