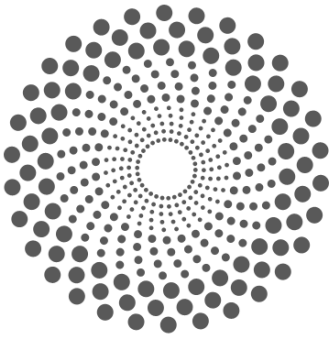


EVA.IO

A decentralized Electric Vehicle Application Platform

The evaio team
www.evaio.info

1. Background

With the rise of Tesla, the automobile industry is going through a paradigm shift. wireless giants have provided mobile internet to the major automakers, Vehicle operating system and user interface become open and friendly, automaker Byton even has a 49 inches giant screen in the concept model, so it is possible for 3rd party developers to design vehicle applications to add quality to the ride experience in the upcoming years, A decentralized system is also approaching its reality.

Last year Elon Mask mentioned in his next 10 years master plan that after true self-driving is approved by regulators, you will be able to summon your car from pretty much anywhere, when you approach your car, the door opens by itself, once it picks you up, you don't have to drive, your car is 10X safer than manual via massive fleet deep learning, on your car you will be able to sleep, read or do anything else enroute to your destination. You will also be able to add your car to a shared fleet and have it generate income for you while you're at work.

After many years of data collecting and improvement on the Level 4 autopilot, to achieve the true Level 5 Self-Driving has been set as Tesla's short-term goal. By that time, commuting by car will be more liberating. We will have more time to get work done, or just relax with some entertainment on the go. An increasing volume of APPs and DAPPs will definitely need to be developed to allow us to fully utilize our time.

2. Opportunities

At present, electric vehicle companies haven't solved all the problems. For example, many EV buyers will install a charger in their parking spaces, but currently they are unable to use other people's home charger, most privately installed chargers are insufficiently shared, a trusted payment solution between the charger and the car is

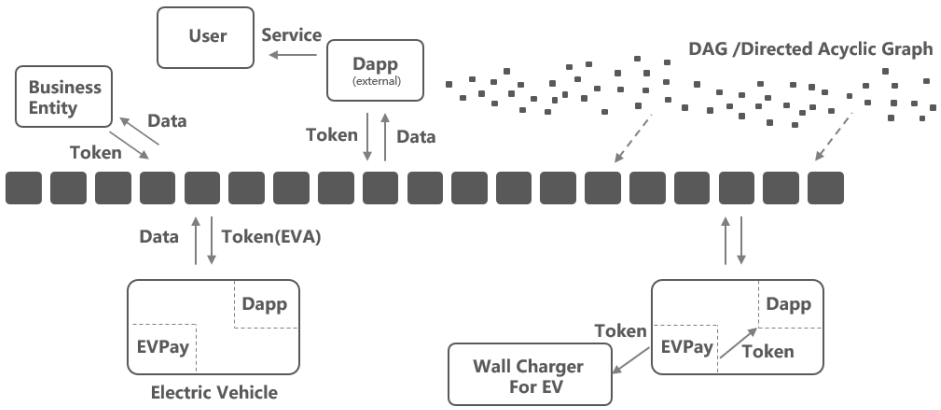
necessary, and most importantly, a system without requiring human supervision.

The connection between the charger and vehicle has already formed a P2P network. Block chain technology makes it possible for this P2P network to achieve efficiency creation and redistribution of profit.

EVA.IO uses block chain technology to create a trusted and secured payment system for vehicles, which can solve the payment issue mentioned above. It also allows unmanned vehicles to be able to trade with the chargers independently. In exchange for contributing data, the vehicle account will receive token rewards. EVA.IO allows vehicles to become self-paid entities with wallets as indicated in above chart, which will stimulate consumption around the vehicles. At the same time, EVA.IO is a public chain to support vehicle decentralized applications including entertainment, communication and shared mobility, all of which brings more value to the riding experience.

3. EVA.IO Highlights

1. Integrated DAG & Blockchain structure
2. Token rewarded when driving
3. Provide Unmanned Self-driving vehicle payment solutions
4. Shadow token and Futures Contract to stabilize price
5. DPOS algorithm to select Block producers, DAG peers and Mainchain Witnesses.
6. Electric Vehicle DAPPs platform that is free to use.

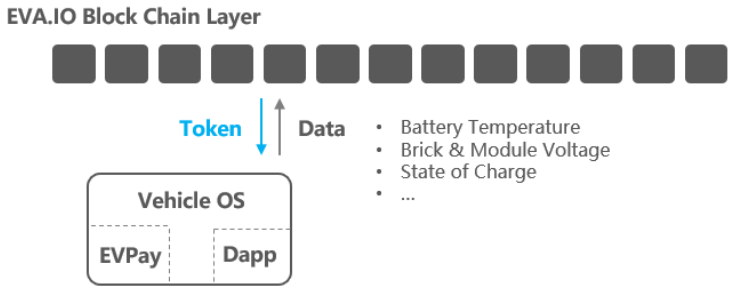


4. Create Currency Based on Data

A decent amount of data can be generated through the usage of the vehicles. The data could be collected from the vehicle, from the drivers' habits, and the usage of APPs and DAPPs. The data can be used to improve services, as well as promote product iteration.

While there is such good volume of valuable data, but the vehicle owners never benefit from it, even though they drove and generated the data.

Before Tesla, there hasn't been a vehicle be upgraded through OTA, and on top of that data was confined to the vehicle. But now we can see a possibility to trade car-generated data and to get paid immediately. The blockchain will create currency and trust, which will make the cycle simpler. Every vehicle owner can directly transmit secured data while get tokens as a reward. These tokens have equity value of the platform as well, actually the token is a combination of currency and equity, and this will enable the early token holders get huge return of investment.



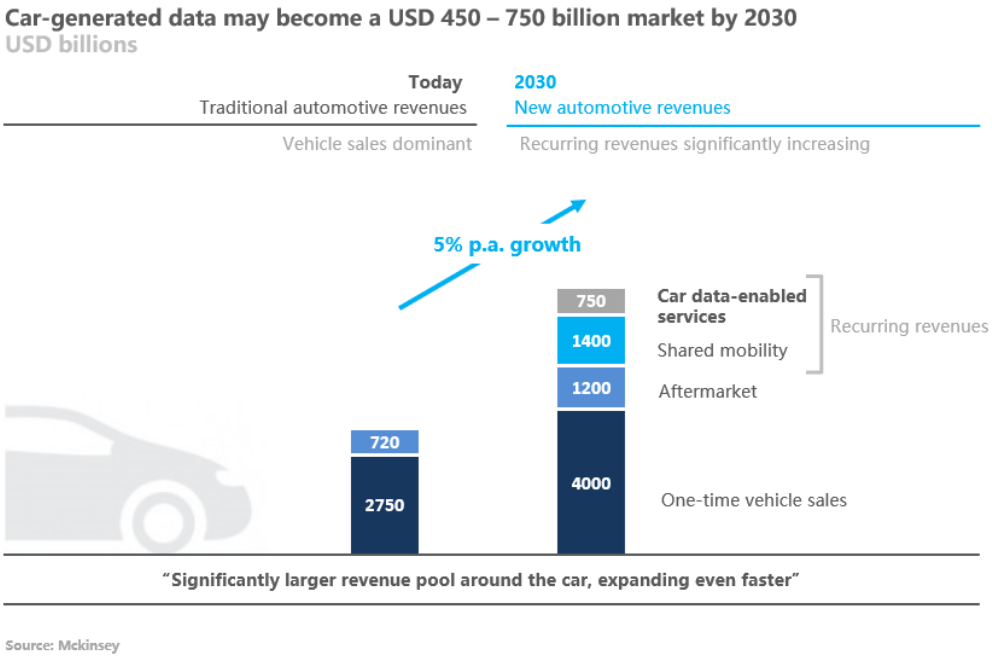
As shown on the chart above, EVA.IO is designed to enable users to activate automatic data transmission to the Blockchain. In return, they get EVA tokens. The information transmitted to the block is all encrypted and stored, and is not open for all to access. EVA.IO allows any third-parties, such as vehicle manufactures, service providers, DAPPs developers and other business entities, to access certain portions of the collected data after paying in form of EVA tokens. The based value of EVA tokens is formed depending on the value and amount of data. Even if no other application on EVA.IO is developed, the potential data trade will still support the basic value of EVA, therefore the price of EVA tokens will never be zero.

EVA.IO will generate multi-cryptocurrency wallets for vehicles with the increasing number of tokens rewarded, meanwhile the token represents EVA.IO system's equity and dividend, with users and money, it will attract more personalized third-party applications to promote consumption and finally activate the EVO.IO ecosystem, when unmanned vehicles become part of daily life, we can imagine that cars become a place to work and for leisure, therefore EVA tokens will represent data value, service value and equity at the same time.

5. Valuable Data

Some of the data we know on Tesla Model S is quite different from traditional vehicles. For instance, battery management system data has provided useful information, there are many sensors deployed on battery modules and bricks that read battery cell voltage and temperature, which then transmits the data to the BMS for analyzing, one important task that BMS do is to control contractors and cut off or restore the power supply to vehicle. Battery temperature is crucial to the life span of the battery. The various temperature gradients will lead to a different performance upon the battery cell during charging and discharging. As a long-term effect, it will result in a discrepancy of cell longevity, which accelerates the degradation of battery pack performance, and finally reduces the quality of the vehicle. This kind of information allows everyone to understand the performance of our vehicle, and it also help improve the product.

Tesla has collected up to 2 billion kilometers of Autopilot data. The feedback has been input to system to train and improve the autopilot. Meanwhile, Google has collected nearly 3.5 million miles of data for their self-driving project. According to McKinsey, Car-generated data may become a USD 450 – 750 billion markets by 2030.



6. Who Owns the Data?

Facebook's data scandal indicates that the questions of data privacy thrown up by the scandal strike at the heart of Facebook's business, which relies on more than 1.4 billion users engaging with the platform each day. Every time they do, they share a bit of information about themselves: what they like, who their friends are, what they want to

watch. That data is the product Facebook sells to advertisers who want to target specific customers. Obviously, a centralized company is unable to protect our data, actually the users generated those data, but we didn't benefit from it.

We know that electric vehicle companies have sent the car-generated data back to their database, it's also confirmed that some governments have required manufacturers to periodically transmit car-generated data to government database. No doubt data is money, we own the vehicle, and our vehicle generates the data while we spend time and drive, then the question is whether those parties using our data is authorized by us.

Vehicle owners have absolute ownership of car-generated data even though the data is not under their control right now. OTA makes the car-generated data transmission a reality, EVA.IO will make vehicle owners benefit from this data selling directly.

7. Data Classification

EVA.IO users will have control over how data is shared. Taking battery for example, to submit below data to blockchain is necessary, Furthermore, besides the car-generated data, other data derived from DAPPs will also be included in EVA.IO. The data in the list will be evaluated before finally confirmed to be input to blockchain, and EVA.IO community will vote and make the final decision.

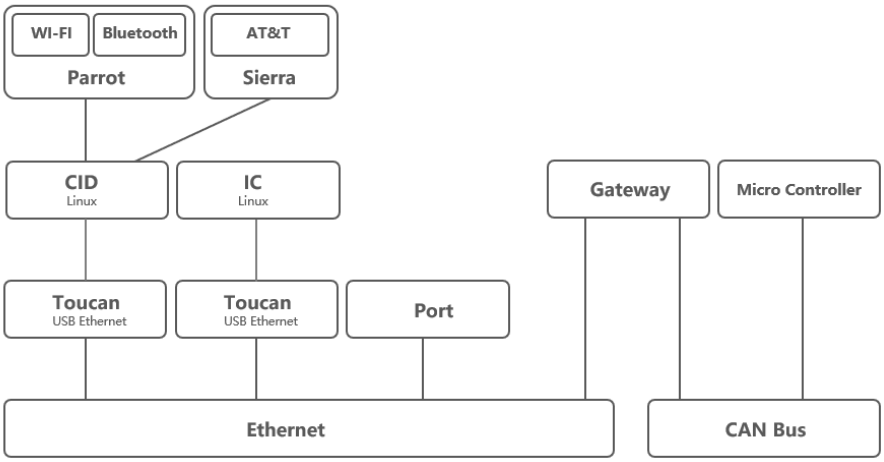
a) *SOC/State of Charge*
b) *Battery Capacity*
c) *Rage (miles)*
d) *Battery module and brick temperature*
e) *Energy consumption*
f) *the consistency of cell/module*

The data will be classified into 3 levels. Sensitive, non-sensitive, and confidential. This gives users the option to choose the level of information disclosure. The data classification also helps sort the value of these data, which makes the token system more accurate and efficient.

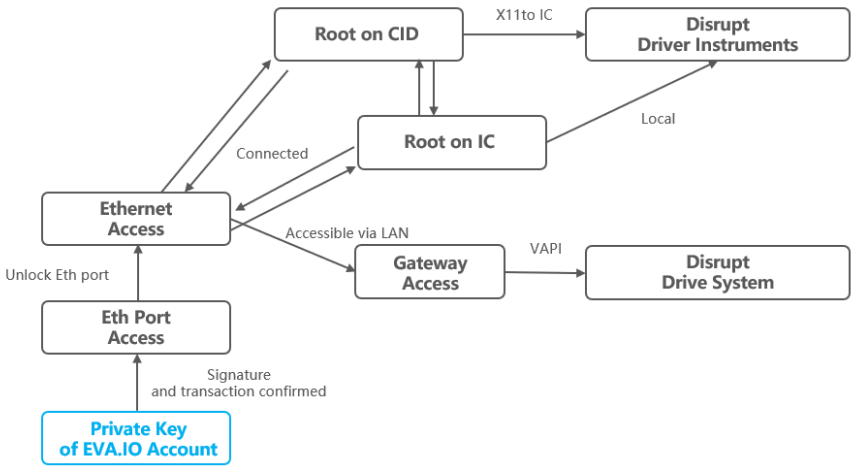
8. Vehicle Safety

When speaking of self-driving, safety issues come to mind. EVA.IO access data through a private key. A hash function will be used to generate the address of vehicle EVA.IO account. This is an extra layer of data protection when the vehicle is connected to internet. Vehicle data can be divided into two general parts, data from Vehicle ethernet and CAN bus. EVA.IO manage those data separately when taking vehicle security into consideration. Taking another look at Tesla Model S as shown in below chart, central information display is connected with in-vehicle Ethernet, while in order to communicate with the micro controllers of CAN bus, all data in DBC format must be transmitted via a

gateway.



When the vehicle is updated via OTA, data transmission becomes more frequent, resulting an increasing pissibility of vehicle being hacked. The below chart shows a special way to access the gateway and finally disrupt the vehicle drive system. To prevent from that, when the car-generated data is transmitted to EVA.IO, it will need the private key first, then the transaction is signed by private key and verified by nodes. Unlesss the transaction is proved to be valid, nobody is allowed to access the data. this structure provided an extra layer of data protection for vehicle. This is how a 3rd pary can trade car-generated data in a trusted environment.



9. Positive Impact to Auto Industry

From the Volkswagen emission scandal to the Subaru mileage cheating, we have to

admit that we cannot 100% trust data provided by automobile manufacturers or any other party of interest. Even with that being said, the data is still mostly relevant to consumers. The block chain can ensure reliability of the data. EVA.IO has defined a boundless data sharing market for every single service provider in the ecosystem. Different manufacturers have different DBC format, but EVA.IO in different vehicles enable the data stored and transmitted in a unique format, the data is shared and encrypted in an order, which will definitely bring the trust to auto industry. Also, DAPP data and CAN BUS data can be also leveraged across companies to create customer value in the vehicle ecosystem.

Even if Volkswagen or Subaru had developed a blockchain with such idea, we would not trust the system since it is highly centralized, while EVA.IO is run by the community. This is also why consortium and public blockchain are not comparable. EVA.IO will positively drive the automobile industry as a trust-establisher within the ecosystem.

10. EVPAY, a Cross-Cryptocurrency Payment Tool

Each electric vehicle will be rewarded with tokens when transferring data to EVA.IO. These tokens will be kept in a wallet of EVA.IO software in vehicle OS. Based on the wallet we create EVPAY, which is something like PayPal but exist specially for vehicle payments. To make sure that EVPAY can achieve the expected performance, we must address three obstacles: speed of transactions, cross-currency payment issues, and acute currency fluctuations. Below are solutions we will offer.

1. Integrated DAG and Blockchain structure to solve the speed.
2. Cross-currency payment and exchange layer to make you use any cryptocurrency in your vehicle without exchange.
3. The shadow token EVAX, and the Futures Contract to solve the issue caused by unstable price.

The above solutions will be described in detail later in the article. Besides another reason that stop cryptocurrency payment from widely accepted is that no decentralized system achieved to engage sellers and customers in one business model. We believe EVA.IO will make it come true after above issues are solved. Below steps are the key strategy that EVA.IO planned to keep users active and bring 3rd party service provider to EVPAY, and finally enable your car to self-pay when self-driving is achieved.

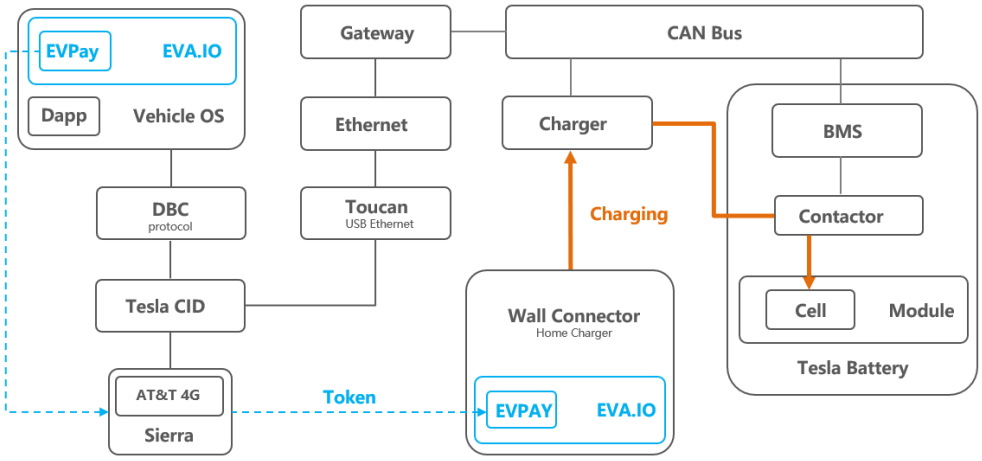
1. EVA.IO creates token based on the value of car-generated data.
2. Launch the Main Net, enable transactions to be confirmed by peers.
3. Test the Drive-Mining function in a small number of vehicles including a Tesla model S.
4. EVA.IO foundation will support to develop the first service application in EVPAY, a decentralized vehicle home charger sharing system, and enable token to circulate in Vehicle EVPAY and Charger EVPAY.

5. EVA and EVAX will work to make the charger owner get fixed-price payment to make business model more attractive.
6. Cooperate with electric vehicle Manufacturers, make more vehicles embedded with EVA.IO software and EVPAY.
7. More vehicles start to get EVA tokens rewarded, which will result in a large number of EVA token holders.
8. EVA.IO will open API for 3rd party vehicle service provider to connect with EVPAY.
9. EVA.IO foundation will support to develop In-Vehicle DAPPs.
10. The self-driving will make a shift from “driving hour” to “consumption hour”. More frequently consumed In-vehicle DAPPs will be developed.
11. The unmanned self-driving scenario will redefine the vehicle as, "sharing, intelligence, self-pay, cryptocurrency owner" which will generate more cryptocurrency consumption outside the vehicle, for example, your car may self-drive to McDonald and self-pay a hamburger and get it back to you.

In next section we’ll introduce first how to deploy a decentralized vehicle home charger sharing system and then EVA.IO architecture.

11. Decentralized Vehicle Home Charger Sharing System

Once cryptocurrency is created based on car-generated data, EVA.IO will be able to support the first EVPAY application, the vehicle private charger-sharing system, this application will be developed by a third-party company with support of EVA.IO foundation. Under this circumstance, the chargers will be upgraded on both hardware and software, and this will enable the charger connected with internet and EVA.IO, the charger will have its own wallet and EVPAY, which will allow the charger to receive the cryptocurrency payment by vehicle users. Meanwhile the chargers become a self-serve “gas-station”, which can provide a solid source of profit, it may generate a new business model, under which people can ran their own “gas-station” network. It will increase the physical density of electric vehicle chargers and finally accelerate the world transition to sustainable energy. The following chart and steps will show how we design the system based on a Tesla Model S and a home charger (Wall connector).



1. A home charger which is called Wall Connector by Tesla, will be upgraded on both hardware and software to enable connection with internet, an EVPAY will also be added to the charger. By doing this the charger will be able to receive the expected cryptocurrency for charging.
2. The vehicle will communicate with vehicle home charger to confirm its availability.
3. BMS (battery management system) determine the State of Charge and transfer the charging request data to CAN bus
4. The Vehicle EVPAY which is connected with Tesla CID get the data from CAN Bus through the gateway.
5. Vehicle EVPAY sends the cryptocurrency that the vehicle home charger owner expected such as EVA or BTC to the charger through 4G module.
6. The Vehicle home charger starts charging once the transaction is confirmed and approved by EVA.IO peers in a second.
7. When the Level 5 vehicle self-driving is achieved, a Snake charger as showed in below images which might be the replacement of current Vehicle home charger, will automatically find the location and plug in for charging.

The snake charger shown above was inspired by a dream of Eon Musk, the CEO of Tesla.

While completing this decentralized vehicle charger sharing system deployment, the payment ability of EVA.IO is also proved. Besides, EVA.IO will open APIs to third parties, allowing more auto-based express services to exist in EVPAY, which can be paying for car washes, parking fees, and so on, just as we used to pay for the utility bill.

As a public chain, EVA.IO can support many future car decentralization applications. One example is decentralized car radio applications. Since the car radio system is user-based content, decentralized car radios will bring more variety of content. The token system is great in redistributing the profit generated by content, the token will be a natural incentive for content creator to encourage them keep providing content and get money, it will make the business model more attractive, and at the same time, it will shorten the process of content trade.

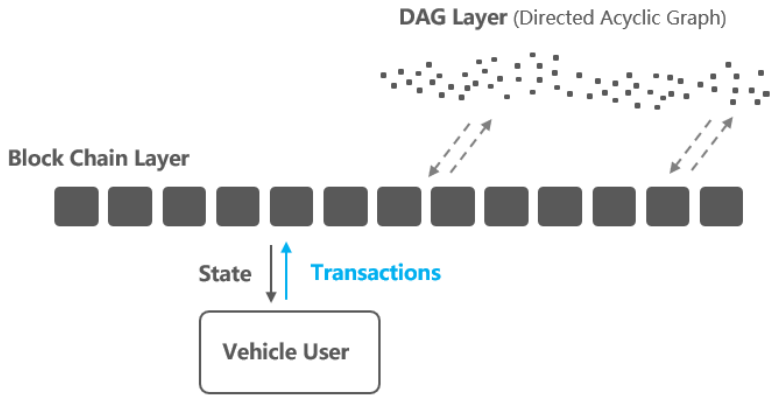
Another interesting application may be the car to car driving social tools based on distance and driving habits, a driver can chat while driving as long as they are

hands-free. For this application, when driving, users can form a group chat. Users will be rewarded tokens based on their input. This activity has advertisement value. Such a business model is more viable than traditional, because there will be no centralized operating company and the profit will be sent to content contributor as the form of token reward.

Once the level 5 Self-driving is achieved, decentralized applications in the car can thrive. Also, there will be great opportunities for decentralized gaming, information and social media applications. Due to the natural advantages in business models. The development of Vehicle DAPPs will beyond our imagination, which makes EVA.IO, the decentralized electric vehicle application platform, more valuable.

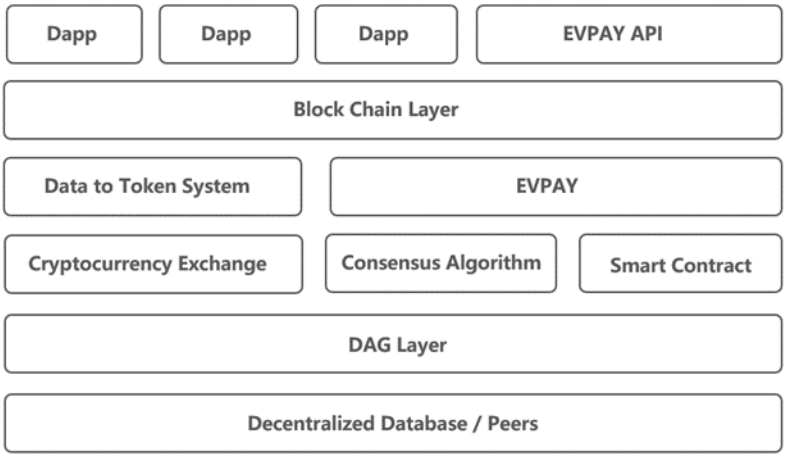
12. Integrated DAG and Blockchain

Bitcoin showed the blockchain architecture and cryptocurrency to the world. A large proportion of projects are blocks and chain mode. Later, IOTA pushed the DAG structure to us. It is undeniable that Both blockchain and DAG structure have their own unique advantages. The discussion around crypto world right now is focused on the three key points, free to users, scalability, and decentralization. In solving the problem of transaction speed, EOS and Ethereum will try on sharing. IOTA and Byte ball are progressing in the direction of DAG. These are actually the expressions of parallel ideas, but we do not know the solution through the concept to practice test. What kind of solution will be the ultimate winner in scalability, especially when the transaction speed is solved without affecting the decentralization; and in different industries, the best solution is different. EVA.IO brings you a new idea, the integrated DAG and Blockchain Structure as shown in below chart.



In this structure, we will use the DAG layer to perform high-speed transactions on the

vehicle IoT payment part. The Blockchain Layer is used to process smart contract transactions, summary of DAG transactions, and issuance of EVA tokens, cross currency conversion tasks. When a transaction is submitted to EVA.IO, the blockchain does not verify or confirm transaction, instead it will analyze the type of transaction, submits high-frequency transactions to the DAG layer, and deals with smart contracts. Then code of smart contract will be transferred to the operating environment. And finally, it is executed by the blockchain nodes.



This kind of integrated structure gives full play to the high scalability of DAG and can better solve the high-speed payment part of EVPAY. In addition, Blockchain Layer and smart contracts ensure that EVA.IO can be used as a distributed database platform for DAPP development, therefore to carry more extensive applications for vehicle services to increase the use of EVPAY.

We see that the DPOS algorithm of EOS allows the token owner to better participate in the management of the blockchain, it reduces the number of nodes to increase the transaction speed, block producers are elected by token holders to reduce the possibility of doing bad things. But it's getting less decentralized; just as there may be many inside stories in national elections, the EOS approach still leaves too much power to few people.

Another important reason for our use of the integrated DAG and blockchain architecture is to make the system more decentralized. Therefore, the block producers and the DAG Peers don't physically need to be the same node. We use the following imprecise analogy to describe the design of EVA.IO in the direction of making power more decentralized.

The block producer is like a finance director, DAG Peers is more like accounting, DAG witnesses like a third-party auditing agency; all three roles have their own responsibilities. Of course, the selection of these roles in EVA.IO is determined by the

DPOS algorithm. We will bring you details later in this article.

EVA.IO Block chain layer allows anyone to write smart contracts and decentralized applications where they can create their own rules for ownership, transaction formats and state transition functions. Our cross-cryptocurrency exchange module and shadow token EVAX using are implemented by using this smart contract as well. EVA.IO virtual machine is Turing-complete. It can encode any computation that can be conceivably carried out.

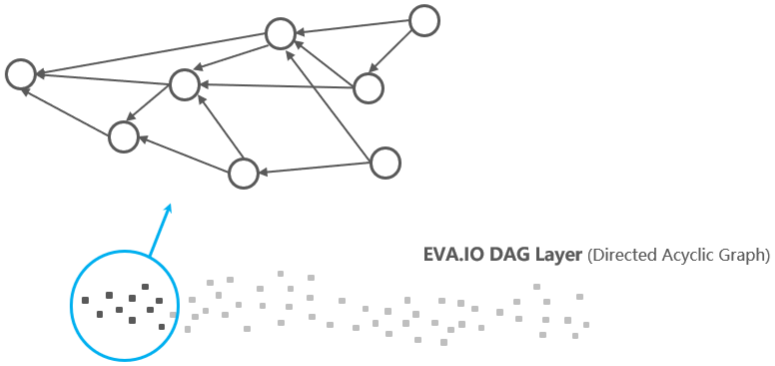
All transactions that are verified by the DAG layer, are eventually summarized sent to the blockchain layer and then verified by the block producer, added into the current block, thus forming the EVA.IO blockchain.

EVA.IO blockchain layer has access control protocols for the car-generated data. These data should not be viewed by anyone unless it was paid.

In EVA.IO Blockchain layer, the block producers do not have to use the computing power based mining to reach a consensus. 15 block producers are chosen by votes cast by token holders, and the selected producers are scheduled in an order agreed by 11 or more producers to produce blocks.

13. EVA.IO DAG Layer

In a DAG structure, there is no Block or Chain, instead if you want to issue a transaction, you must work to approve other transactions. Therefore, users who issue a transaction are contributing to the network's security. It is assumed that the nodes check if the approved transactions are not conflicting. If a node finds that a transaction is in conflict with the transaction history, the node will not approve the conflicting transaction. So, the DAG does not consist of transactions grouped into blocks and stored in sequential chains, but as a stream of individual transactions linked together as showed in below chart.



This DAG structure also enables high scalability of transactions. The more activity in the DAG, the faster transactions can be confirmed. That's why EVA.IO added a DAG layer to solve the speed, EVA.IO DAG layer is designed to process transactions only, and the primary target of TPS is above 5000.

We also have witnesses in our design, witnesses are some of the participants of the network, they are non-anonymous reputable people or companies who might have a long-established reputation and finally chosen by votes of token holders, 9 witnesses will finally participate in the system to verify transaction in DAG layer and post the main chain. While it is reasonable to expect them to behave honestly, it is also unreasonable to totally trust any single witness.

The way to build a main chain is to develop an algorithm that, given all parents of a unit, selects one of them as the "best parent". The selection algorithm should be based only on knowledge available to the unit in question, on data contained in the unit itself and all its ancestors. Starting from any tip (a childless unit) of the DAG, we then travel backwards in history along the best parent links. Traveling this way, we build a main chain and eventually arrive at the genesis unit.

Markov Chain Monte Carlo (MCMC) algorithms is used to randomly select the peers on our DAG layer for a transaction that has just arrived. This is to make sure that peers don't know which transaction they will verify, and finally to decrease the possibility that the DAG peers attack the system.

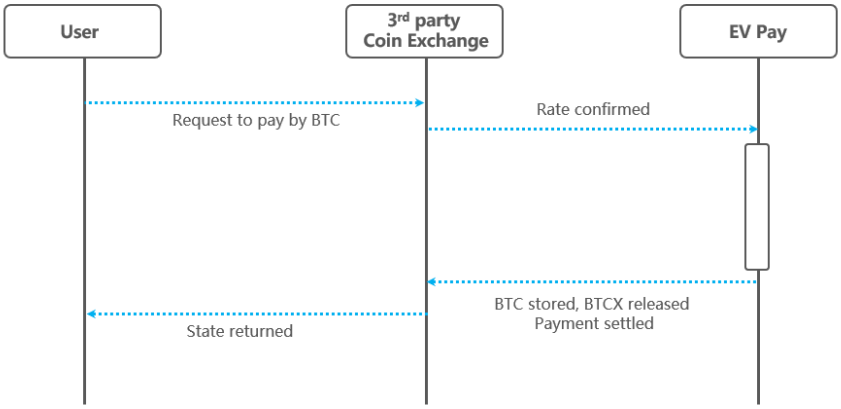
In EVA.IO DAG layer, the number of peers will increase when it receives more transactions.
confirmation time is the time from a unit entering the database to reaching stability. It depends on how often the witnesses post, since to reach stability we need to accumulate enough witness-authored units on our DAG Main Chain after the newly added unit. To minimize the confirmation period, the witnesses should post frequently enough. Our witnesses will be elected by the community and they need to meet the above expectation to ensure the performance.

14. Cross-Cryptocurrency Payment

The types of cryptocurrency are increasing. If we design a system that cannot accept multi-cryptocurrency, it would be disappointed for users. Say like if you are a buyer, you have 10 ETHs, and service or commodity sellers want to receive BTC. We need to allow EVPAY to perform quick redemption in system without the user's perception and successfully complete the payment; while the BTC and ETH's blockchain architecture determines that they are slow and it might be hard to complete the transactions on BTC and ETH mainchains; therefore, we use the following process to solve the problem of rapid payment across cryptocurrencies.

1. EVPAY Wallet generates shadow tokens for each cryptocurrency, such as BTCX, ETHX.
2. EVPAY and the external Exchange set up an exchange rate inquiry and settlement agreement.
3. EVPAY will use the exchange rate to achieve the desired currency quote in the trading system. If you wish to pay with BTC, you will see the BTC quote.
4. Taking BTC payment as an example, the buyer triggers the smart contract to lock the BTC under the multi-signature mechanism, and the system 1:1 converts the shadow token BTCX.
5. BTCX will be exchanged to ETHX according to the exchange rate and pays the seller to complete the payment. All this transaction is verified and approved by the DAG layer and was finally summarized in the blockchain layer.
6. The external Exchange will simultaneously complete the real transactions of BTC to ETH on their main net.
7. After the exchange is completed, ETHX is replaced with ETH.

In order to ensure a 1:1 exchange rate between shadow token and mortgaged token, a smart contract is used. there is to ensure nobody can unlock the mortgaged tokens. All transaction will be verified and approved by peers as showed in blow chart.



15. Shadow Token EVAX and EVA Futures Contract Module

The high degree of instability in the value of digital assets is an important reason that hinders the widespread use of digital assets in actual transaction payments. We recognize that to apply digital assets to day-to-day transaction payment scenarios requires the currency value of digital currencies to be stable; while most current digital assets may be related to the equity, dividends, and usage rights of blockchain products, which makes the current value of token fluctuates all the time, and the seller doesn't need an equity, but pays more attention to how much money he has actually earned

today. This restricts the seller from accepting cryptocurrency payments on a large scale; for this we create the shadow token EVAX makes the payment related to USD. EVAX has the following characteristics.

1. EVAX is designed to be equal to the dollar price at any time, 1EVAX=1USD
2. EVAX does not trade alone, nor does it appear alone. It only serves as the shadow value of EVA.
3. The generation of EVAX is triggered by smart contracts based on the value of the mortgaged EVA.
4. The number of EVAX generated = the number EVA mortgaged * the current price of EVA token.
5. EVA.IO will Collaborate with a third-party company to short EVA in the secondary market while EVA is mortgaged, this ensures the value of digital asset is always equal to the dollar price represented in the exchange during the delivery period.

Here we take the unmanned charger-sharing system as an example to describe the EVAX generation and transaction process, and how to use the futures contract to stabilize the value of tokens seller received.

Alice drives a Tesla Model S looking for a shared private charger; she finds Bob's home charger is available nearby, so she approaches the charger and the vehicle identifies Bob's charging pile and inquires through the BMS. She pays with EVA tokens through the EVPAY. At this time, the EVA price is US$10, and Alice pays 1 EVA token for this charging. The payment triggers a smart contract on the EVA.IO and transfers the EVA to a smart contract account. This process is multi-signed to ensure the security of the mortgaged EVA.

The price of Shadow token EVAX is always a constant price of US$1. While the system mortgages EVA, 10 EVAX is paid to Bob from the smart contract account. Then EVA.IO verifies the transaction, and then triggers an event to inform Bob's charging pile. The charging pile then performs charging. If only this is the case, the rise and fall in the price of EVA will actually have a serious impact on exchanges.

If EVA drops to $9 after 1 minute, then an EVA is still mortgaged in the system. With a current worth of $9, Bob receives a constant EVAX value of $10. When Bob proposes to convert 10 EVAX in EVPAY. the system can only redeem one EVA worth a total of $9, and no one can afford another $1 loss.

The smart contract will short EVA futures contract in an exchange, When the system mortgages the EVA to EVAX, this EVA futures will guarantee the total value of the EVA collateral + futures contract is at the constant value of 10 US dollars. This approach ensures that Bob can convert 10 EVAXs into EVA worth $10 at any time, regardless of whether the EVA price goes up or down.

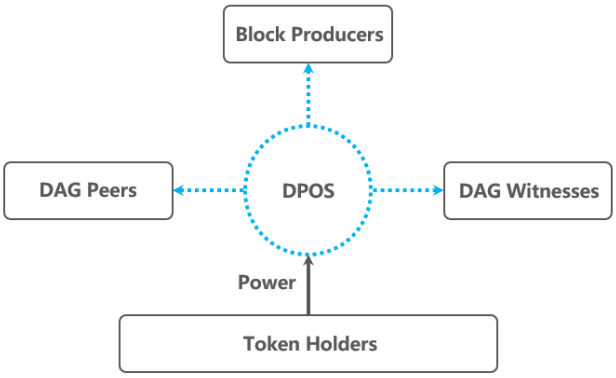
16. Consensus Algorithm（DPOS + MCMC + PBFT）

EVA.IO utilizes Delegated Proof of Stake (DPOS) to select 3 roles, Markov Chain Monte Carlo (MCMC) to randomly send the transactions to DAG peers, Practical Byzantine Fault Tolerance (PBFT) for state communication. Under DPOS algorithm, those who hold EVA tokens may select Block Producers, DAG peers and DAG witnesses through a voting system.

Anyone may choose to participate in the system, will be given an opportunity to produce blocks, issue transactions and post main chain, they can persuade token holders to vote for them.

In general, the DAG peers will issue transactions for end users, the DAG witnesses will post the best child-parent link transactions as the DAG main chain, the witnesses can be understood as regulatory agencies in the DAG layer to ensure double-spend will not happen, the Block producers will verify the summarized transactions and produce blocks in Blockchain layer.

The 3 roles in this model have their separate responsibilities but serve the same purpose to make the ledger safe, we imprecisely call this Separation of powers, the powers come from token holders as showed in below chart.



The algorithms in EVA.IO work in the following procedures:

1. DPOS Algorithm is used to elect 3 roles, Block producers, DAG peers and Witnesses.
2. The DAG peers are randomly selected by MCMC algorithm.
3. The selected DAG peers start to do a light POW algorithm to find the nonce and then issue a transaction in DAG network.
4. The DAG witnesses post the best Parent-child link as the Main Chain.

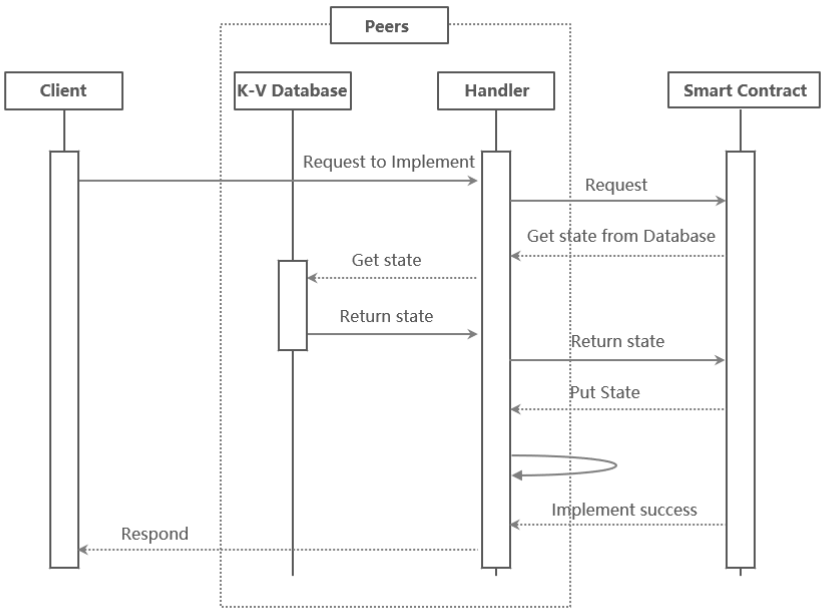
5. PBFT algorithm is used when block producers send messages to each other.
6. The block producers summarize the transactions and create the EVA.IO blocks.

This design is more decentralized and has higher scalability compared with a blockchain, also this model allows us to deploy decentralized apps much easier.

17. Smart Contract

EVA.IO's smart contracts are divided into system contracts and user contracts. System contracts are used to implement system-level functions, and user contracts implement user applications. The contract is compiled into a separate application and runs in isolated Docker. Compared with Ethereum, the EVA.IO smart contract is separated from the bottom account. When you upgrade smart contracts, you do not need to migrate the ledger data to the new contract. while the contract is written in Go, Java, NodeJS.

Below procedure shows how a smart contract transaction is processed between Block chain layer and DAG layer.



1. When the DAG peers receives the input (propsal) and request from client, it will send a message object (with input information, caller information) to the corresponding smart contract.
2. The contract invokes in the ContractBase. By sending getState and putState messages, the contract obtains the account status information from the DAG peer and sends the precommit status.

3. The contract sends the final output to the DAG peer. The peer signs the propsal and output (propsal response) and completes the first paragraph of the signature submission.
4. The client then collects the first piece of submitted information, forms a transaction and signs it, sends the transaction to the Blockchain layer, and finally produce the block and send feedback to DAG peer. The output falls on the ledge and the second submission process is completed.

With the support of smart contracts, cross-currency transactions can be completed. The process is as follows:

1. User initiated asset deposit request and entered digital asset into system
2. system returns the token to the user's wallet and triggers a joint billing request in EVPay and the digital asset distribution network.
3. The seller is pricing the digital asset as expected.
4. Buyers complete real-time payments in tokens of their own digital assets.
5. Update the mapping relationship between digital assets and tokens in EVPAY.
6. Initiation of redemption acceptance requests for tokens and digital assets triggers cross-chain and cross-network transactions.

18. Scalability

The DAG allows the child transactions to approve parent transactions when the number of transactions increases. Obviously, we need more DAG peers elected to issue transactions for users. The number of DAG peers is confirmed by the following method.

user think time is *T_think*,
number of users is *U_concurrent*
transaction response time is *T_response*
system throughput is: *TPS=U_concurrent / (T_response+T_think)*

EVPAY guarantees that the time for each transaction is *C_time = Confirmation time*, assuming *T_response = a*C_time*, then: *TPS=U_concurrent / (a*C_time +T_think)*
So, *C_time = (U_concurrent/TPS - T_think)/a*

In the production environment, the system adjusts the C_time to the set value according to the number of concurrent users U_concurrent, T_think and TPS design indicators affecting the user experience, and horizontally expands the DAG bandwidth, storage, memory, and other computing resources to ensure the TPS index, and user experience.

19. Token Model and Incentive

We believe it's important for application developers to offer users free services; users should not have to pay in order to use a decentralized application, which means there

will be no transaction fee in EVA.IO platform. That' why a constant incentive plan is needed for Block producers, DAG peers and DAG witnesses. The EVA.IO may be configured to enforce a cap on this incentive plan that the total annual increase in token supply does not exceed 0.5%, the community will participate in the final decision making.

In our design, the total number of EVA in system is 20billion, car generated data is the basis of token value, therefore 40% of total supply will be locked and rewarded to Car owners and manufacturers in a long period (more than 50 years), after that the community will discuss and decide whether to increase the token supply. Besides, every time when a driver get 10 EVA rewarded, the manufacturer will receive 1 EVA.

In addition, we have set up a decrementing mining mechanism similar to bitcoin as follows

1. An individual owner could be rewarded no more than 100 tokens per day when less than 0.5 billion tokens in total are mined.
2. An individual owner could be rewarded no more than 50 tokens per day when less than 1 billion tokens in total are mined.
3. An individual owner could be rewarded no more than 25 tokens per day when less than 1.5 billion tokens in total are mined.
4. An individual owner could be rewarded no more than 15 tokens per day when less than 2 billion tokens in total are mined.
5. An individual owner could be rewarded no more than 10 tokens per day after 2 billion tokens in total are mined, this pattern ends until all 8 billion tokens are rewarded.

The owner may not be able to receive the above-mentioned 100% mining incentive token limit because mining is related to the mileage and data level of the day. We can describe this model of EVA.IO as mileage mining.

EVA.IO allows each account to consume a percentage of the available capacity proportional to the amount of tokens held by developers. If an account holds 1% of the total tokens, then that account has the potential to utilize 1% of the storage capacity.

20. Conclusion

We have proposed a new structure in EVA.IO, the integrated DAG and Blockchain, a high scalability can be achieved without suffering the less decentralizing. tokens are created based on data, no transaction fee is needed for users to consume the resources, payment can be settled by any cryptocurrency you expect, a fixed-price shadow token helps seller to accept cryptocurrency, and finally it will enable your car to self-pay when self-driving.

Reference

1. Satoshi Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System, https://bitcoin.org/bitcoin.pdf, 2008.
2. EOS.IO https://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper
3. Anton Churyumov, Byteball, https://byteball.org/Byteball.pdf
4. Ethereum, A Next-Generation Smart Contract and Decentralized Application Platform, https://github.com/ethereum/wiki/wiki/White-Paper
5. Sergio Demian Lerner. Dag Coin, https://bitslog.files.wordpress.com/2015/09/dagcoin-v41.pdf, 2015.
6. Serguei Popov. The Tangle, http://iotatoken.com/IOTA_Whitepaper.pdf, 2016.
7. Tom Holden. Transaction-Directed Acyclic Graphs, https://bitcointalk.org/index.php?topic=1504649.0, 2016.
8. https://github.com/bitcoin/bitcoin