parsiq

# BLOCKCHAIN MONITORING PLATFORM

WHITEPAPER

FEB 2020

# DISCLAIMER

This document is for informational purposes only and does not constitute an offer to sell shares or participation in the PARSIQ project.

The development described in this document is a conceptual model of a proposed system, rather than a complete specification of a service offering. The future development process and system specification may be subject to change.

# ABSTRACT

PARSIQ is a universal blockchain parsing and monitoring tool. The tool provides an analytical view of blockchain transaction history. It also provides real-time monitoring of accounts, transactions, and related blockchain state.

Advanced parsing and processing of transaction trees using big data analytics techniques and machine learning algorithms provide detailed insights into blockchain asset movements.

The platform will eventually work for multiple blockchains and implements a modular design aimed at providing many applications for different use cases. At its core, PARSIQ is designed as an open platform accessible and extensible for everyone.

PARSIQ solves the problem of insufficient monitoring and analytics tools availability in the cryptocurrency and blockchain ecosystems. The reorganization of asset and financial infrastructures that has taken place in the shift from centralized structures to trust-free decentralized architectures has caused information about digital assets and transactions to be harder to come by. This is partly because of the pseudo-anonymity of most blockchains, but mostly due to the lack of tool support.

PARSIQ addresses this lack of tool support by providing a comprehensive and modular blockchain analytics platform.

The system is implemented as a mix of centralized and distributed components, combining a high-performance transaction backlog processing service with real-time decentralized monitoring of unconfirmed transactions.

At the lowest level, PARSIQ monitors individual blockchains in real time, providing transaction replays through instrumented virtual machines that can classify transactions into event types. The resulting events can be used for forensic and analytical purposes or as triggers for user-specific real-time monitoring services.

TABLE OF CONTENTS

# THE NATURE OF BLOCKCHAIN-BASED ASSETS

Since the launch of Bitcoin in 2009, we have learned a lot about cryptocurrencies and other digital assets. It is now possible to represent almost any asset on a blockchain in one form or another. Tokens are used in business use cases to represent things as diverse as currencies, voting rights, and even non-fungible assets, such as property deeds and shipping containers.

This new way of representing and interacting with assets requires businesses using blockchain solutions to change their procedures in many ways, including how they keep track of assets and asset interactions.

Traditional assets, whether stored in bank accounts or traded in markets, can be monitored, tracked, and safeguarded through a number of mechanisms and tools, —usually through the intervention of a trusted third party, such as a bank, broker, or public authority. Blockchain-backed assets are different in this sense as decentralization usually involves the removal of trusted third parties.

Blockchains return control over assets to the asset owners. This dramatically empowers end-users but also places the responsibility for the asset's security and correct management in their hands. In some cases, control may also be ceded to a new type of trusted third party such as a cryptocurrency exchange, for example.

# THE PROBLEM
# ASSET PROTECTION AND TRACEABILITY

In the early days, Bitcoin was often used for criminal activity because of assumed anonymity and the lack of experience of law enforcement agencies in dealing with it. Admittedly, the pseudo-anonymous nature of public blockchain does in fact complicate asset traceability.

It is often difficult to know who is involved in a transaction. Furthermore, the ease of cross-border transactions also makes it easier for assets to be moved to different jurisdictions. In general, once a digital asset has been compromised it is harder (most of the times - impossible) to recover than a traditional asset.

“Values” represented on a blockchain, whether they are cryptocurrencies or other tokenized assets, are protected by asymmetric cryptography. In theory, private keys are safely kept in the end user's wallet. In practice, however, key management and related off-chain security breaches are the weakest links in cryptocurrency asset management.

Assets are usually controlled by so-called hot wallets for convenience. This type of wallet is connected to the internet and subject to cybersecurity attacks. Even worse, many assets are stored in wallets hosted by a third party's server, such as cryptocurrency exchanges. Private keys stored on an exchange’s server are favorite targets for cybercriminals and, as a consequence, both types of wallets are frequently hacked and drained of their content.

Most assets are compromised through phishing or other scams. As in all cybersecurity, the weakest link in the security chain is the human factor. Even if a user is cautious, assets usually have to be moved and converted through third-party exchanges and could be compromised through phishing attacks on the exchange's staff.  Apart from phishing, there are many other ways that may lead to assets being stolen.

Once an asset has been compromised, it usually ends up on a cryptocurrency exchange or another type of provider working with fiat currencies (e.g. pre-paid debit cards backed by cryptocurrency wallets, p2p marketplaces and other service providers).

Usually, once a user detects a problem, it is too late. Recovering or "freezing" an asset is no longer an option. Furthermore, dealing with such an incident is complicated by the fact that there are seldom any data channels between different providers.

Another issue related to asset management is the lack of transparency in unregulated markets which leads to traders not knowing how their assets are managed. Scams and theft are widespread. Fees and transaction times are often unclear. Insider trading and market manipulation are hard to detect and are fairly commonplace.

# THE SOLUTION
# BLOCKCHAIN ANALYTICS

Bitcoin, and blockchain transactions, in general, are much more traceable then assumed. All data and transactions on a blockchain are publicly readable by everyone. Transactions are interconnected.

The capability of tracking assets used in criminal activities is clearly important to authorities, but it is not the only use case for asset tracking and blockchain monitoring. Business and personal users also need asset tracking. Examples of asset tracking include the monitoring of individual accounts, analyzing transaction histories, keeping track of movements on cryptocurrency exchanges, and many other applications. The primary use cases for asset tracking, whether personal or business-related, focus on protection and market intelligence gathering.

Blockchains are maintained by participating nodes, where each node keeps a copy of the blockchain's transaction and state history. This means that there is a large set of structured data available, but misses interpretation.

This data can be analyzed in many ways. At its simplest, individual accounts can be monitored for movements, providing a history of incoming and outgoing transactions. As a result, real-time alerts of movements can be issued to protect one's assets.

In addition to analyzing individual accounts, it is possible to trace assets through transactions. When tokens, cryptocurrencies, or other assets are transferred, they leave a trace of incoming and outgoing transactions, even if different accounts are used and transfers split up in many transactions in to obscure movements. Bitcoin, for example, uses a so-called "Unspent Transaction Output (UTXO)" model in which unspent outputs of transactions are used as inputs for new transactions (Figure 1). This input/output matching is clearly traceable.

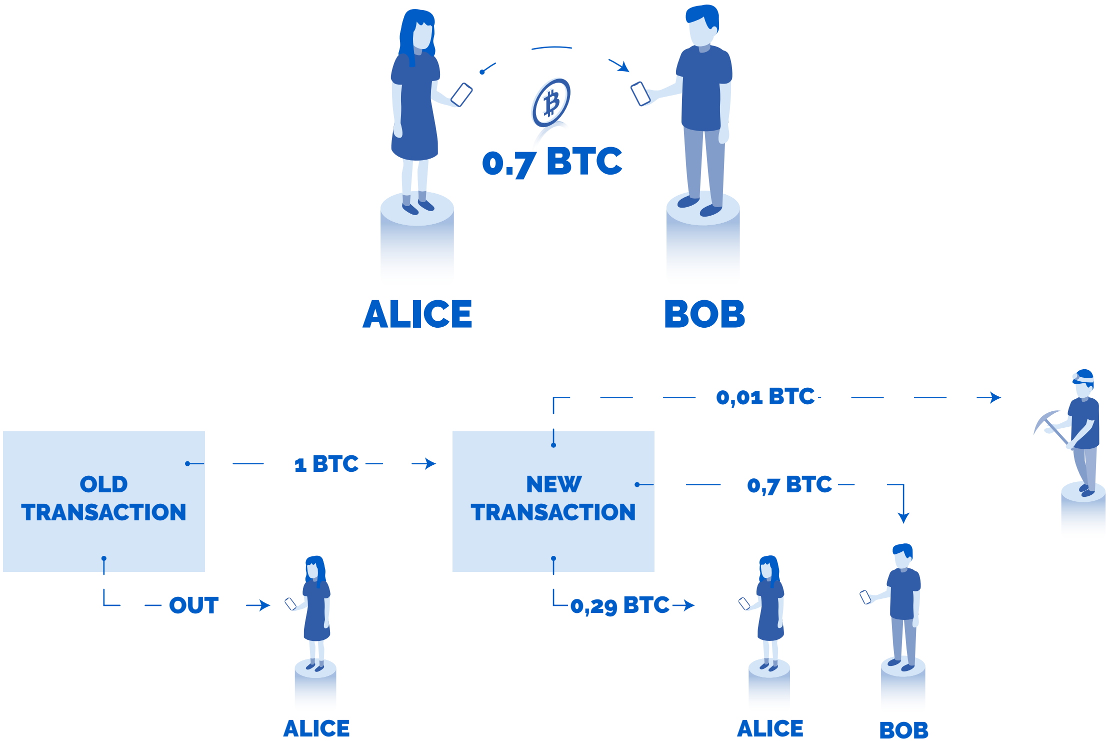


*FIGURE 1 - BITCOIN UTXO MODEL*

Monitoring interactions with smart contracts can also provide valuable information. Cryptocurrency exchanges' smart contracts and cross-chain gateways can be analyzed for asset conversions from one chain to another. This can provide important market insight for traders in certain situations where there are gaps.

Tracking accounts and transactions this way allows one to gain insights automatically (rather than it would be a process of manual transaction following). However, blockchain data available can be interpreted in many ways. The tools that currently exist do not offer anything specific in terms of analyzing the data and making sense of it.

Different transfers from seemingly unrelated accounts may be co-related to certain events. Assets may also be tracked across non-trivial account movements, and trading patterns may give insights into big investments moves or even market manipulation.

# PLATFORM OVERVIEW

PARSIQ is a universal multi-chain parsing and monitoring platform. The platform provides a transparent way of obtaining an analytical view on the transaction history of different blockchains. It also provides real-time monitoring of accounts, transactions, and related blockchain state.

Blockchains represent a tectonic shift in the way data is transmitted and represented. However, the tool support needed to parse through and analyze that data hasn't caught up with the blockchain realm until now. PARSIQ addresses this lack of tool support by providing a comprehensive and modular blockchain analytics platform that is publicly accessible.

The following stakeholders may benefit from using PARSIQ:

- **Consumers** benefit from PARSIQ by monitoring their cryptocurrency wallets and keeping track of their on-chain assets to provide themselves with an additional layer of security.

- **Businesses** can use PARSIQ in a similar way. They can benefit from additional monitoring of enterprise assets such as multi-signature wallets or track the interactions with certain business-related smart contracts. The transaction history of a business's clients may also be used for customer profiling.

- **Traders** can predict market movements based on factual blockchain data. They can also use real-time tracking to detect movements that may indicate certain market activity and act beforehand.

- **Law enforcement agencies** can use forensic analytics modules to investigate blockchain related cybersecurity incidents or track payments made in relation to illicit activities.

- **Blockchain researchers** from a variety of fields can use PARSIQ to gain insights into blockchain usage relevant for their particular purposes. For example, a computer scientist interested in blockchain scaling may draw conclusions from typical transaction processing patterns to design better blockchain protocols. A social scientist may be more interested in the nature of human interactions across the blockchain.

**PARSIQ supports both B2B and B2C business models, offering services to enterprises and private users.**

# KEY FEATURES

The PARSIQ platform constitutes a **distributed solution for reverse-engineering** of other blockchains. While being up to date with current implementations of major blockchains (mainly Bitcoin, Ethereum, and their forks), PARSIQ listens to all the activity in corresponding blockchains and builds a fully indexed representation of what happens in a particular blockchain. This process generates much more data than can be found in any single blockchain. To manage such a massive amount of data, PARSIQ introduces separate distributed data management layers using **blockchain-specific feature extraction**.

For those blockchains where the notion of the full-node is relevant, PARSIQ introduces instrumented virtual machines for replaying all the existing transactions while capturing crucial data that could not be easily obtained from the artefacts persisting in a blockchain (e.g., “internal” transactions in Ethereum, cross-contract calls, and internal asset movements within transactions). For sharded (future) solutions, PARSIQ maps its functionality over all available shards and reduces collected data to the data management layer. For side-chains and solutions like Plasma, PARSIQ introduces hierarchical data domains that can be attached on demand.

PARSIQ **labels data and builds graphs** to parse blockchain transaction history. All data with asset movement and distribution semantics is indexed using several special indexes that speed up reachability and distance analysis. Source, destination, and the arc itself are subjects for arbitrary labeling. PARSIQ introduces public and private labeling that can be generated by user-provided algorithms.

The platform provides both **pull queries** and **push queries** through the concept of **smart triggers**. While providing traditional query-response interfaces, PARSIQ allows users to upload a definition of a smart trigger with its own private data storage that can be populated by the smart trigger owner (e.g. a list of addresses under investigation).

To effectively monitor multiple addresses generated by hierarchical wallets, the system has an **awareness of common key derivation schemes**. Therefore, instead of uploading and querying a lot of individual addresses, PARSIQ itself can derive the range of keys from provided cryptographic material.

# ASSET TRACKING

One difficulty in designing a platform for blockchain analytics and transaction tracking is that blockchain transactions are relatively low-level. For instance, it is trivial to track individual Ethereum transactions, but higher-level business transactions are often of more interest. As such, low-level transactions may not correspond to asset movements at the business logic layer.

Deriving business logic-level asset traceability from low-level transactions is complex and requires an understanding of off-chain processes. For example, different addresses often correspond to the same user as wallet software may use key derivation standards to generate a hierarchy of keys and accounts from a seed.  Awareness of such standards and additional processing allows PARSIQ to derive information on asset movements at the business layer.

Transactions are classified into different types and translated into higher-level asset movements if appropriate. The resulting events are fed into the PARSIQ analytics components and can be used in user-defined triggers, set up for monitoring purposes.

# CONCEPTUAL MODEL

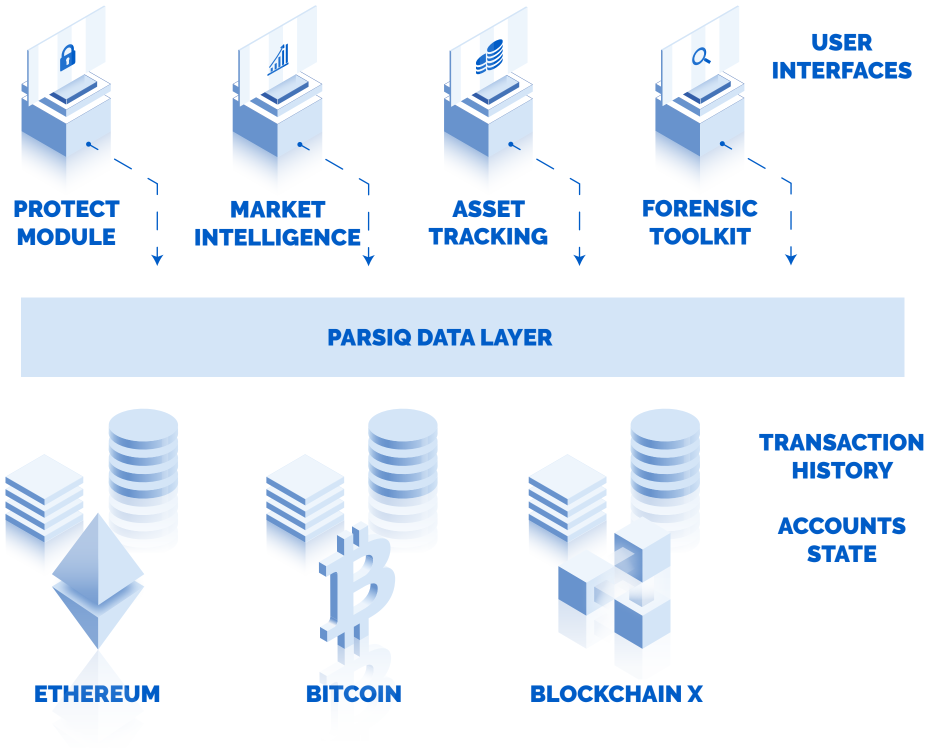


*FIGURE 2 - PARSIQ ARCHITECTURE*

A conceptual view of the high-level PARSIQ system is illustrated in Figure 2. At the lowest level, PARSIQ interfaces with several blockchains through a number of full archive node gateways run by participating nodes to obtain the required data on which to execute the higher-level analytics algorithms.

The first fully supported blockchain will be the Ethereum blockchain. The reason for using Ethereum as the first proof of concept implementation is related to the platform's popularity for tokenized assets and smart contracts. Further blockchains will be added to the system as demand indicates, with Bitcoin scheduled with high priority.

PARSIQ's modular design allows different modules to process this data for specific needs. We will introduce the initial modules in the next section of this paper. The architecture has been designed using this modular approach for several reasons. First of all, using different modules for different types of blockchain tracking and analytics allows to clearly separate concerns between algorithms and use cases. Secondly, a modular structure allows us to release modules as required and in close contact with the community, incorporating feedback through an agile process.

Depending on specific use cases, a module may expose different user interfaces for different user categories. For example, a business user may be presented with a slightly different user interface than a private consumer.

# USE CASES

Algorithms can be used to achieve a variety of blockchain analytics functionalities, suitable for many use cases. These use cases can be classified into three categories along a timeline:

- Forensics → Analyzing past occurrences and transaction history

- Real-Time Monitoring → Analyzing the present by generating alerts on specific occurrences

- Forecasting → Predicting future occurrences

The following table summarizes some example use cases for each of these categories:

| Real-Time Monitoring | Forecasting | Forensics |
|---|---|---|
| Tokenized asset tracing | Tracing wallets of “whales” and big players that have certain market influence | Token circulation analysis (post-ICO behaviour) |
| Wallet protection | Pattern recognition | Cryptocurrency laundering & theft tracking |
| Real-time accumulation or sell-off alerts based on movement of a screened asset | Market intelligence | Forensic intelligence for parties that are interested in such services (authorities, government, KYC/AML procedures etc.) |
| Real-time exploit execution tracking (smart-contract monitoring) | Gas price and congestion prediction | Wallet classification |

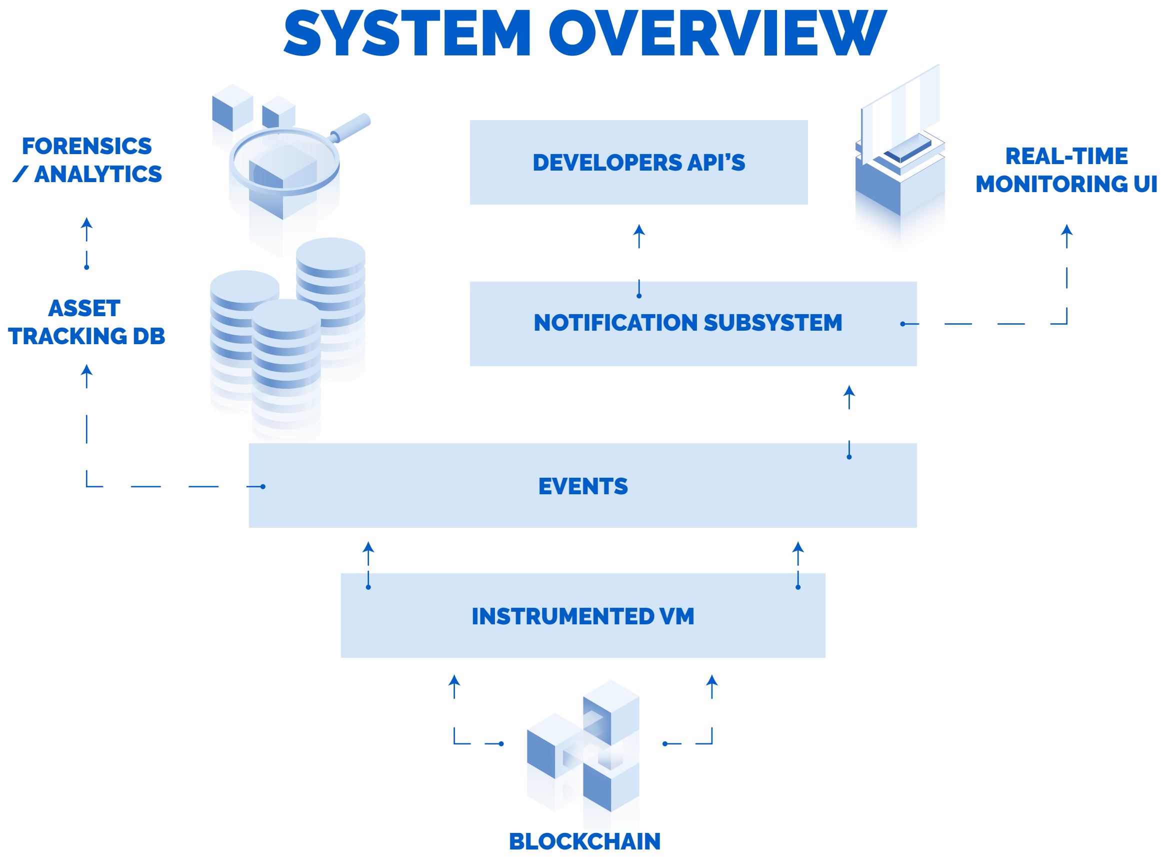


*FIGURE 3 - PARSIQ ARCHITECTURE*

The PARSIQ platform for each blockchain is fed by nodes that participate in the blockchain's core network. Figure 3 illustrates such a setup.

Each feeder node runs an instrumented version of the blockchain-specific virtual machine, allowing low-level transactions to  be classified into higher level events. These events serve as an input to two user-level components. A generic asset tracking database and analytics front-end allow end-users to track assets, visualize results and perform forensic asset tracking tasks.

In parallel, a notification subsystem, aimed at developers, allows the implementation of use-case specific applications using real-time alerts.

In what remains of this section, we will discuss these components in further detail.

# DATA ACQUISITION LAYER

Data is read directly from the blockchain. To this end, full nodes participate in the blockchain monitoring transactions at the protocol level.

The PARSIQ platform divides relevant transaction history into three parts: cold, warm, and hot. This classification is illustrated in Figure 4.

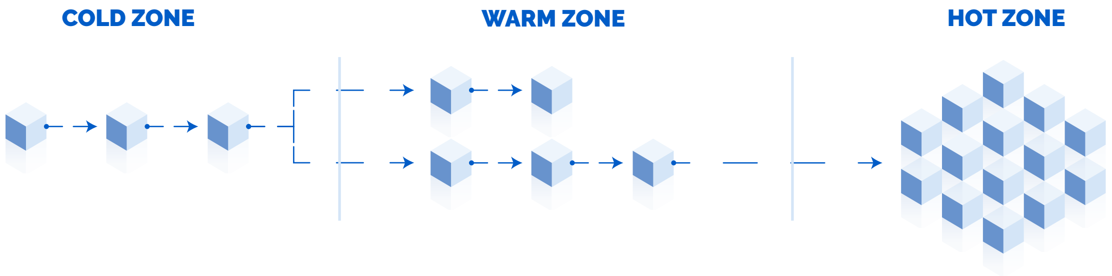


FIGURE 4 - BLOCKCHAIN LAYER

The need for this classification stems from how most blockchains deal with transaction confirmation. Transactions embedded into the blockchain for a certain amount of time are linked in a single chain of immutable blocks. We call this area of the blockchain the **cold zone**, which constitutes the majority of the transaction history.

However, as most blockchains use a probabilistic finality model, the chain frequently forks for short period of times. In a healthy blockchain, this occurs when different miners or block producers generate blocks more or less at the same time. Forking can also be the result of malicious activity, but conventional forking is very common, in particular in blockchains with a relatively low block time. In Ethereum, the rate of forking is measured by the so-called “uncle rate” meaning that, because forks are so likely, producing stale blocks (uncles) is considered a contribution worth incentivizing with a portion of the block reward. The probabilistic finality model is the reason why transactions are only considered confirmed until after a certain number of blocks have been added onto the chain. In PARSIQ, we call this sliding window of potential forking the **warm zone**.

Finally, before transactions are included in a block, they are kept in a data structure called the mempool. Nodes submit transactions to the mempool in the hope they'll be picked up by a miner or block producer. Depending on the load of the network and the transaction fee a sender is willing to spend, transactions may remain in the mempool for a long time before being included in a block. Transactions in the mempool may also be "overtaken" by paying a higher transaction fee. This is called front running and is sometimes exploited in trading. However, front running may also prevent a fraudulent transaction from doing too much damage by removing funds that are about to be stolen. We call the mempool the **hot zone** as it is here where most changes can occur and transaction history is most recent and volatile.

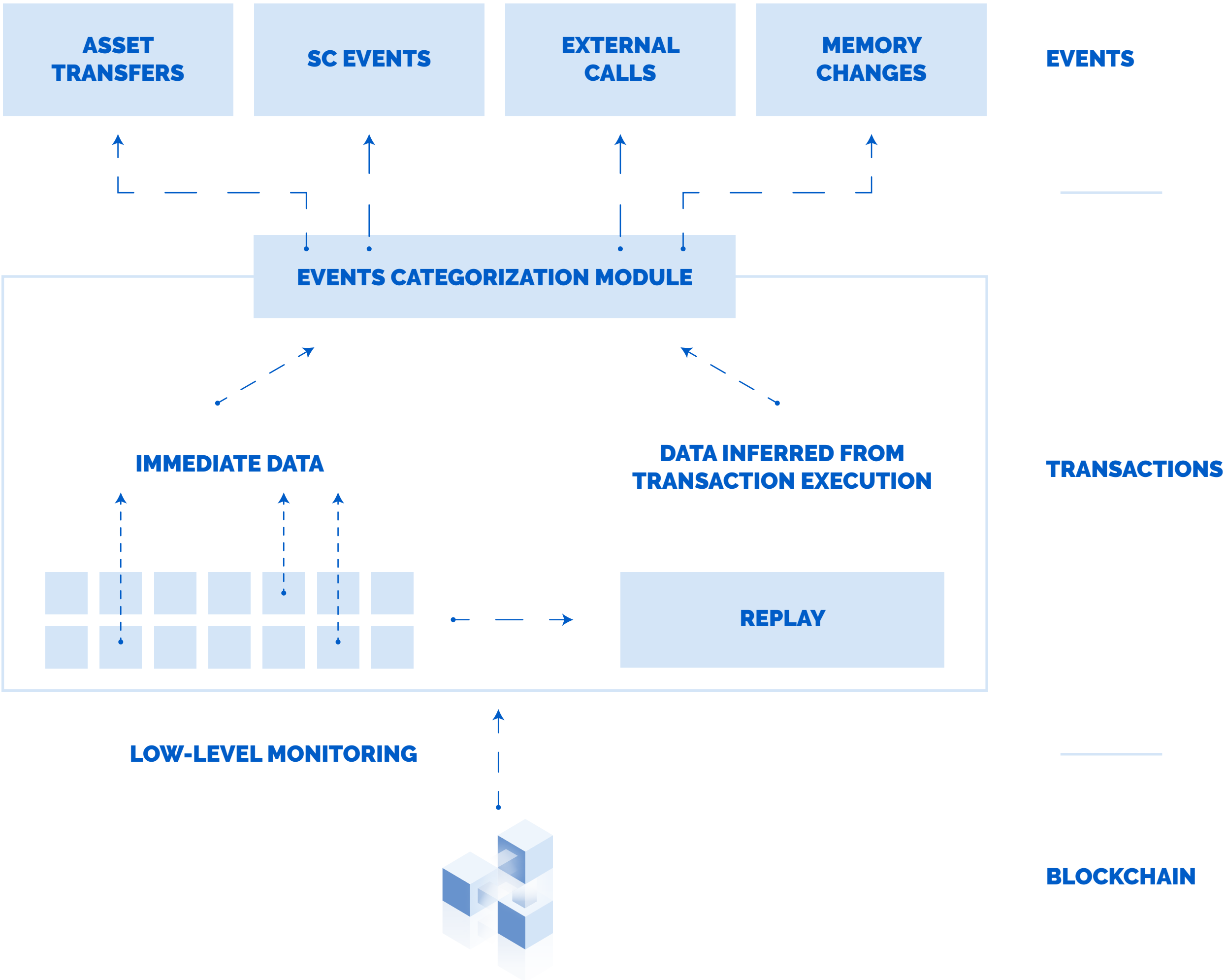


*FIGURE 5 - TRANSACTION MONITORING*

Monitoring of the larger cold zone can be performed by a centralized system as correctness can always be verified by comparison to the actual blockchain. However, the warm and hot zones must be covered by a distributed set of nodes as each node may actually have a different view of current state (different transactions may have propagated to one node but not yet to another).

Figure 5 shows how transactions read from the blockchain are used for two purposes.

Firstly, raw transactions provide a certain amount of valuable immediate information. Other information can be gathered from replaying the transaction through the PARSIQ instrumented virtual machine.

The resulting information is fed into an event categorization module, which classifies events that are then sent up to the application layer.

# ANALYTICS AND FORENSIC UI

The analytics and forensic UIs are web interfaces aimed at visualizing the transaction history of blockchains, thus allowing users to trace digital assets, label wallets, view risk scores, and perform analytic studies that aid in cryptocurrency management and trading.

Our interfaces consist of two separate UIs: a **forensic module** and a **market intelligence module**. The former includes the following functionalities:

- Transaction visualization: Funds can be tracked and visualized in trees of transactions, allowing the origin of assets to be traced.

- Connections between wallets: A graph view will enable visualization of connections between different accounts and transactions between them.

- Labeling of accounts: This functionality allows accounts to be classified according to their usage. Labels may include large traders, ICO accounts, exchange accounts, multisig smart contracts, token contracts, etc.

- Address statistics. Statics, such as the origin and use of funds can be retrieved from a database and visualized in different views, including pie charts, bar charts, and histograms.

- Wallet scoring: Trust scores are kept for wallets and can be displayed in the analytics UI.

The Market Intelligence module focuses on trading analytics. Apart from the simple trading intelligence parameters such as ticker prices and news feed, the “MI” module allows the user to follow specifically labeled wallets and trading accounts of interest.

There are two views of the UIs. B2C clients are presented with an end-user specific view. Enterprise clients have slightly different UI, due to the nature of the services required by B2C customers.

# NOTIFICATION SUBSYSTEM

The notification subsystem is the second application-level component provided by PARSIQ. It provides both a graphical user interface for end-users and APIs for developers, allowing the programming of real-time monitoring applications.
Figure 6 illustrates the structure of the notification subsystem.

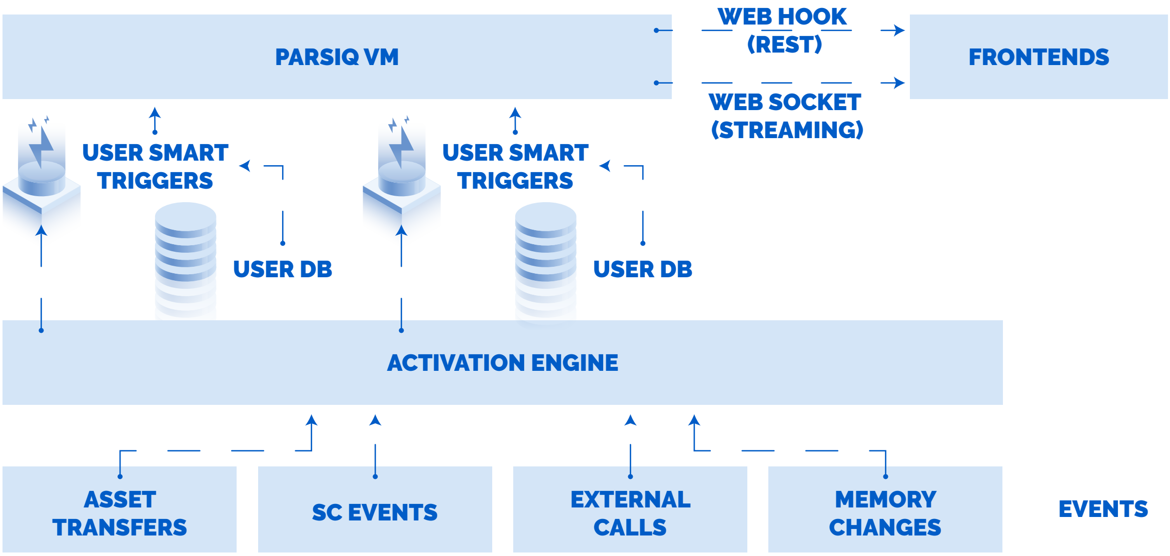


*FIGURE 6 - NOTIFICATION SUBSYSTEM*

Events received from the data acquisition layer are fed to an **activation engine**, which causes user-defined **smart triggers** to be executed. Triggers are stored together with relevant data in PARSIQ-hosted user databases.

Triggers execute pre-defined functionalities of the PARSIQ VM, which are accessed by two means:
• A REST API for standard HTTP messaging
• A Web Socket interface, allowing synchronous streaming services with push functionality

On top of these APIs, different user interfaces can be implemented. PARSIQ itself provides a user-friendly web interface which allows non-expert users with a means of deploying simple smart triggers.

Further interfaces can be implemented by developers using the APIs directly.

# SMART TRIGGERS

## IMPLEMENTATION

Smart triggers are one of the most powerful features of the PARSIQ platform. They allow a wide range of monitoring and notification applications to be built on top of our infrastructure.

While the PARSIQ VM provides the primitives to implement very sophisticated triggers through a domain specific language, it is entirely possible to expose a significant subset of these primitives at the graphical user interface layer. Doing so allows non-expert users to set up their own triggers easily.

At the logic level, smart triggers consist of the following three parts:

- An **expression** defined by the user in a **domain specific language**. These expressions follow a reactive paradigm and are usually in the "if-this-then-that" format.

- **Atomic inputs** from the **event layer**. These inputs are in the form of classified events being passed up to the activation engine from the data acquisition layer.

- A **user database** containing descriptions of assets being monitored — for example, a list of addresses to be monitored for transfers. The user database is hosted by PARSIQ but allows users to upload their relevant data.

Smart triggers can be used for multiple functions and allow creative combinations of the classified events that can be detected by replaying transactions through  PARSIQ's instrumented virtual machines.

## USE CASES

The following are very simple examples of the type of actions that can be monitored using smart triggers:

- Asset being withdrawn from a certain address
- Asset being sent to a certain address
- Asset transfers above a certain threshold
- High-frequency of transactions to and from exchanges
- Interactions with a certain smart contract
- Invocation of a specific smart contract function
- High gas transactions that may indicate front-running attempts
- Failed transactions

It should be evident from the range of possibilities that the applications of smart triggers are far-reaching. Users and businesses are able to use our infrastructure instead of building their own.

One basic but handy feature is allowing users of cryptocurrency exchanges to detect unauthorized asset movements. Exchanges themselves can use this feature to implement automatic blocking/instant tracing of stolen assets. They may use smart triggers themselves and integrate them with blacklists of fraudulent addresses.

The prevention and tracing of unauthorized asset transfers extend, of course, beyond the scope of exchanges and is of equal benefit to enterprise users and end-users. Smart triggers allow exchanges and other businesses to react immediately and automatically to questionable transactions.

Smart trigger functionality doesn't end there, however. Let's consider a deployed smart contract that is missing a particular feature. Imagine that an action has been left out or needs to be added to a smart contract event. Due to the immutability of the blockchain, upgrading the smart contract is not possible, and re-deployment is not always an option. This could easily be the case for a token sale contract to which a bonus feature needs to be added. Smart triggers can react to these types of smart contract events by being set up to detect relevant investments, thus invoking a separate bonus payment and **correcting the mistake** in the original smart contract.

Naturally, the monitoring of smart contracts can be used for many things beyond the above example, ranging from the detection of attacks to unintended smart contract interaction patterns that may indicate attempts to exploit a vulnerability. Spam pattern detection can be employed at the mempool level, detecting attempts to cause transaction congestion by launching large numbers of transactions that will never succeed.

# CENTRALIZED VS. DISTRIBUTED MODEL

When designing a blockchain-related architecture, an inevitable consideration that arises is whether the system itself should be implemented as a decentralized application. As supporters of decentralization and valuable members of the crypto community, the PARSIQ team has performed extensive research into the different ways of designing the platform's architecture.

At first, it might seem that beyond ideological considerations, a fully decentralized architecture is the best solution due to its independence and geo-distribution. However, performance considerations and the hardware requirements for a fully decentralized PARSIQ platform may negate the benefits obtained through decentralization.

In general, there are three possible models for implementing the PARSIQ platform:

- Centralized. The service is provided by PARSIQ operated backends.

- Distributed. The service is provided through a P2P network of independent PARSIQ nodes and synchronized through a special-purpose PARSIQ blockchain.

- Decentralized. Independent user nodes operate the PARSIQ software and provide different results to the system but without the need for strict consensus and full data replication.

The table below highlights the implications of three possible models at each point in the implementation.

| | Centralized | Distributed /duplicated | Decentralized |
|---|---|---|---|
| Data processing and analysis. | Most efficient solution. No need for network transfers, no latency problems. Data is always at the place where calculations are performed. | Replication problems and inefficiencies. High requirements for data traffic, decreased latency for the client, increased availability for the client. More attack resistant. | The problem of trust in data, data availability at the processing node for analysis and computational overhead, redundancy consensus problems. Increased requirements for networking, nodes have to perform identical calculations to prove the legitimacy of results. |
| Data storage of transactional data | High requirements for storage and its performance, fastest access to data, the highest data processing rate and capabilities. | Processing speed is still high, duplicated structure leads to an increase in costs, replication issues. | High requirements for storage space and performance, data availability at a processing node, lack of trust to stored data. |

| | Centralized | Distributed /duplicated | Decentralized |
|---|---|---|---|
| Data storage for smart triggers (user storage) | Can guarantee security and access restrictions. | Can guarantee security and access restrictions, increased costs for servicing infrastructure, higher availability compared to the purely centralized option. | Problem with keeping sensitive data in a decentralized manner (access to sensitive data), problems with the integrity of data (large chunks of data have to be kept at different nodes and compiled together each time this data is used in processing algorithm). The cumulative cost of data storage (nodes storing data have to be incentivized to do so, which leads to the drastic increase in excessive growth in overspending). |

| | Centralized | Distributed /duplicated | Decentralized |
|---|---|---|---|
| Blockchain cold analysis | Done once and its results are kept in trusted storage, creates a temporary spike in hardware load. | Leads to duplicated calculations and excessive overhead. | Networking overheads, duplicated calculations, high requirements for motivating nodes (fees) that perform calculations. Cumulative costs for motivation rapidly go through the roof. |
| Blockchain warm analysis | Reasonable access to p2p broadcasted blocks. | Slightly faster access to p2p broadcasted blocks, duplicated costs. | Lack of trust, consensus problems, slow and expensive. |
| Blockchain hot analysis | Slower detection of mempool transactions, spam problem, most cost-efficient. | Faster detection of mempool transactions. Spam problem. No trust issue, but has the problem of costs. | Faster detection of mempool transactions. Spam problem which is partially solved via reputation tracking. Trust problem. |

PARSIQ IMPLEMENTATION

| | Centralized | Distributed /duplicated | Decentralized |
|---|---|---|---|
| Mem pool spam problem | Machine learning | Machine learning | Excessive costs, inefficient process of machine learning, a mismatch between hardware and requirements for successful system training and test. |
| Activation engine | Cost-effective, but slower solution compared to distributed. | Increased costs, but win in speed and parallel processing. | Lack of trust, problems with access to sensitive data. |
| Parsiq VM | Most cost-effective. | Slightly faster due to network latency, but increased costs. | Lack of trust, problems with access to sensitive data, speed inefficient. |
| Execution Parsiq VM (Delivery) | Most cost-effective, but single point of failure. | Faster, but possibly increased costs. | Unacceptable for clients due to potential security issues. |
| Back-end of front-end | Straightforward implementation. | No obvious advantages compared to centralized. | Relevant only for a fully decentralized system and entails network, trust, and sensitive data problems. |

|  | Centralized | Distributed /duplicated | Decentralized |
|---|---|---|---|
| Back-end of data processing | Straightforward implementation. | Makes sense only in case of distributed data storage. | Security issues, vulnerable to DDOS attacks in relation to the data keeping node. |
| Front-end | Straightforward implementation. | Same as centralized. | Possible implementation as a thick client, more flexibility in terms of white labeling and modifications of the front-end. |

# CHOSEN IMPLEMENTATION

From the above table, it should be surmised that a detailed analysis for each of the platform's functionalities has been performed with the chosen outcome highlighted.

The PARSIQ team has concluded that a decentralized architecture model is not the most efficient solution. PARSIQ will operate on tens of terabytes of data meaning the speed of calculations and RAM requirements would easily make min. requirements for a full operational node too high to run on general-purpose hardware.

Making nodes less resource demanding would come at the cost of speed, which is a cost the PARSIQ platform can't afford. Speed and quality of calculations are the reasons clients value PARSIQ over other solutions, so compromises can't be made to that end.

Thus, the PARSIQ implementation is a mixed centralized/distributed model. In this way, a centralized back-end operated by PARSIQ receives inputs submitted from decentralized monitoring nodes and offers services to end-users and application providers.

Decentralizing blockchain monitoring and event classification in a distributed PARSIQ P2P network is particularly advantageous for the monitoring of the warm and hot parts of the blockchain, as it is there where different nodes may receive different views and report different observations. The PARSIQ team will continuously review this situation with the intention of striving toward greater platform decentralization in the future, whenever advances in technology permit us to do so.

# PROTECT

## THE PROBLEM

Blockchain-based digital assets are stored in wallets that, in theory, are kept safe by user-maintained private keys. In a perfect world, private keys would suffice to keep those assets safe and untouched. In reality, however, user-errors, off-chain security breaches, and private-key mismanagement result in tragic losses that mount on a daily basis.

Adding to the ongoing crisis of cryptocurrency assets lost to user-errors, the persistence of cybersecurity threats looms large over assets held in so-called hot wallets. Hot wallets are wallets hosted on third party servers — this practice is common amongst cryptocurrency exchanges. Such wallets are popular targets for attackers and have proven fruitful for them as billions of dollars worth of digital assets have been stolen.

When performing a security audit of an entire chain of actors, it is important to recognize the weakest part in that chain. As is the case in cybersecurity at large, the weakest link in blockchain security is the human factor. Phishing attacks and other scams are rife in the cryptocurrency world and provide ample opportunity for users to be tricked into revealing their private keys.

For users who have mistakenly revealed information leading to the theft of their assets, there is generally nothing that can be done to recover or otherwise suspend the stolen assets.

## THE SOLUTION

PARSIQ Protect is an alarm and notification system for wallets, contracts, and service providers. It offers a real-time monitoring solution for end-user wallets and accounts, exchange accounts and smart contracts. Unauthorized movements can be detected instantly and the user will be notified in several ways. In addition, funds can be frozen straight away once a breach is detected on the receiving party's end. PARSIQ Protect is the **next layer security feature**, which is still active after the theft incident already occured (e.g. it is "above" two factor authentication in a situation where the mobile device was compromised).

Use cases of the PARSIQ Protect can be categorized into **consumer and business use cases.**

A **consumer** can set-up an alert on a wallet he/she owns. The user can prove ownership of the wallet by signing a message with Metamask's browser plugin[1] or in an any other convenient way.

If the wallet is not in use, an alarm system may be set to ON mode the same way a burglar alarm is switched on in a real-world property. If an unauthorized transaction occurs, the user is notified in several ways:

- PARSIQ Protect smartphone app
- E-mail notification
- SMS notification

Notifications will provide the user with information on the incident and ask for quick confirmation and feedback.

Independent of the user's actions the system will start tracing the unauthorized transaction, and even if the assets are funneled through a mixer, all the wallets that are involved will be "flagged" to ensure full traceability. Once these assets reach any exchange participating in the PARSIQ or the ECA network, the exchange will be instantly notified.

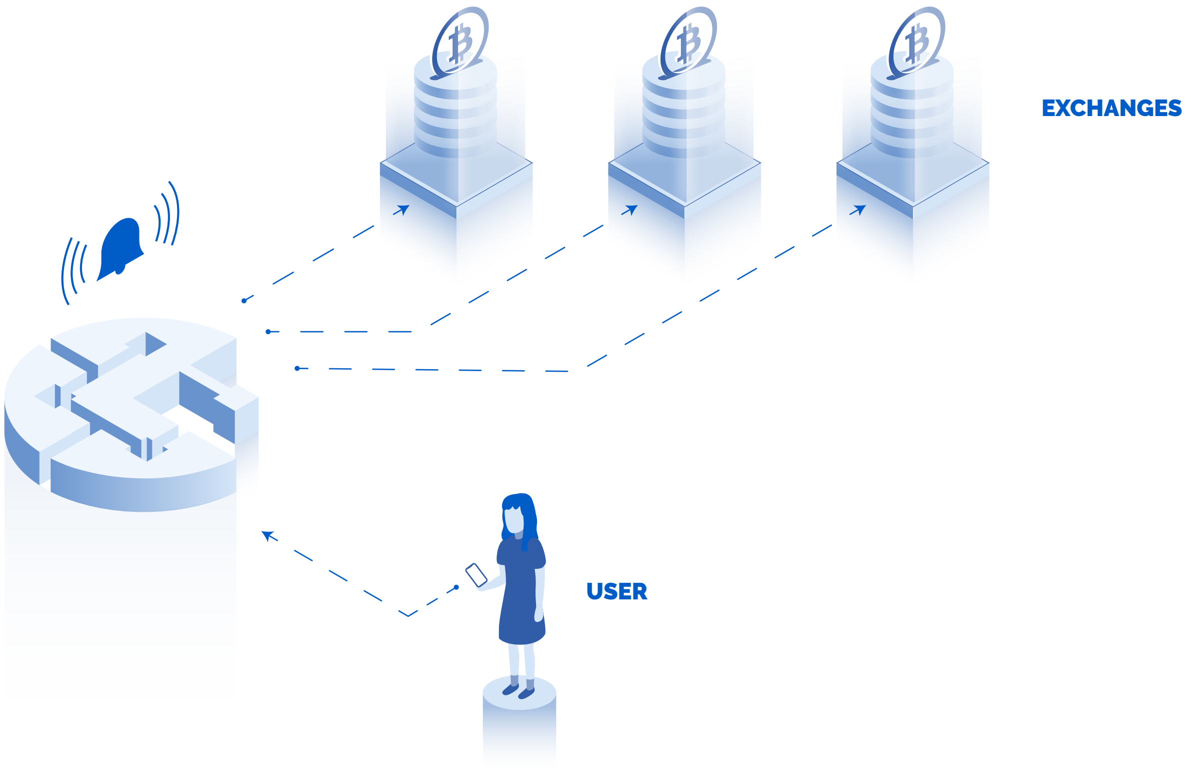
*[1] Metamask Browser Plugin. https://metamask.io/*

Participating providers receive these notifications even before the number of confirmations needed to process a deposit is reached.

As the PARSIQ Protect user-base grows along with the participating provider network, it will become harder to launder stolen assets.

PARSIQ Protect's **business use case** is similar in nature. Businesses can set up monitoring for their smart contracts (such as ICO contracts). PARSIQ, for example, can monitor ICO contracts for changes in token supply that are due to an exploit in the smart contract's code. PARSIQ then alerts the contract owners, follows the newly generated tokens, and automatically alerts the service provider to whom the tokens were transferred to.

Businesses such as exchanges can monitor their hot wallets for large volume of outgoing transactions. Exchanges using PARSIQ can set a percentage change in wallet balance as a threshold for triggering an alert and tracing the assets involved.



# MARKET INTELLIGENCE

## THE PROBLEM

Crypto-asset markets are a recent phenomenon and are still mostly unregulated. Several issues make it very difficult for investors to read the market, interact with assets and time their investments:

- Lack of transparency. The unregulated nature of the crypto markets means that certain activities do not have to be registered and are not overseen by any authority. This leads to situations in which traders are unaware of certain caveats.

- Market manipulation. Big players and other so-called “whales” have the resources to manipulate the market. “Pump and dump” schemes are widespread. Individual traders have no ability to react fast.

- Insider trading. People close to project teams have access to privileged information that allows them to time their investments to maximize profits and gain unfair advantage.

All the above issues are made worse by the fact that crypto markets are small in comparison with traditional markets — small manipulations can have a tremendous impact.

We have to keep in mind that cryptocurrency trading is different in comparison to traditional markets. There is an ability to **cross-correlate** public blockchain data with news and exchange prices immediately.

## THE SOLUTION

PARSIQ Market Intelligence (MI) provides "behind the curtains" analytics for crypto asset investments. The MI module is based on the fact that there is a short but noticeable lag before blockchain transactions are confirmed and reflected in an asset's price. For example, it may take at least 20 minutes for large incoming transfers that may cause a selldown to be confirmed.

PARSIQ MI can detect such movements in real time before movements affect a user's trading account. Notice that in this case, PARSIQ provides functionality not present in traditional trading — this is made possible by the open and transparent nature of the blockchains.

The following are examples of PARSIQ MI's capabilities:

- Track specific movements and transactions before they reach cryptocurrency exchanges

- Track trends based on actual transactions instead of unreliable or misleading information "outside" of blockchains.

- Provide deeper statistical data on crypto asset markets

- Cross-correlate news and prices with blockchain transactions

- Set-up monitoring and alerts to know when certain events happen and take action beforehand

# BLOCKCHAIN FORENSICS

The Blockchain Forensics module exposes the functionality underlying asset tracking functionality to users to allow lower level forensic analytics.

Whereas the previous two modules described a focus on real-time alerts, the forensics module allows looking at historical data and individual assets and transactions in detail.

The module provides facilities for consumers and business users to trace transactions and wallets of their choice. To do so, the user provides the wallet address of the txid and uses a powerful user interface to analyze movements. Transactions can be traced according to a user-defined depth expressed in hops. The hop terminology is borrowed from computer networking[2]: A hop represents a single step in crypto asset movements.

Users can create forensic cases in the forensic module, label transactions and wallets, and create detailed analytics studies. To this end, the forensic module provides a large variety of filters allowing detailed customization based on search criteria.

*[2] Hop terminology. https://en.wikipedia.org/wiki/Hop_(networking)*

# PARSIQ TOKEN (PRQ)

## INTRODUCTION

The PARSIQ token (PRQ) is an essential piece of the PARSIQ platform that co-exists with FIAT payments for using it's services. PRQ tokens will be issued during the crowdfunding period and unlock various possibilities for their usage in the PARSIQ ecosystem.

## USE CASES

Payments within our platform that are made in PRQ tokens guarantee a discounted rate. PRQ token utility is mainly tied to computational costs.

The tokens are usable for:

- Computing resources costs (storage, network, computation)
    - Storing User Data
    - Notification type and rate (up to some amount of notifications per a certain period)
    - Smart-trigger deployment and execution (complexity, sensitivity)

- Forensics product queries

- Market Intelligence product queries

- PARSIQ Protect subscriptions

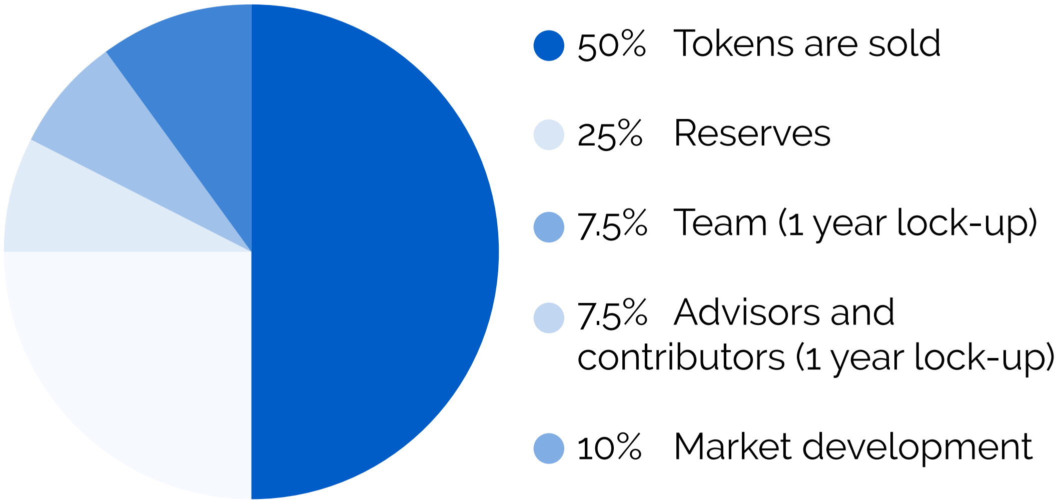
- Other services related to the platform

## PARSIQ TOKEN (PRQ) - 0XFE2786D7D1CCAB8B015F6EF7392F67D778F8D8D7

### TOKEN DISTRIBUTION

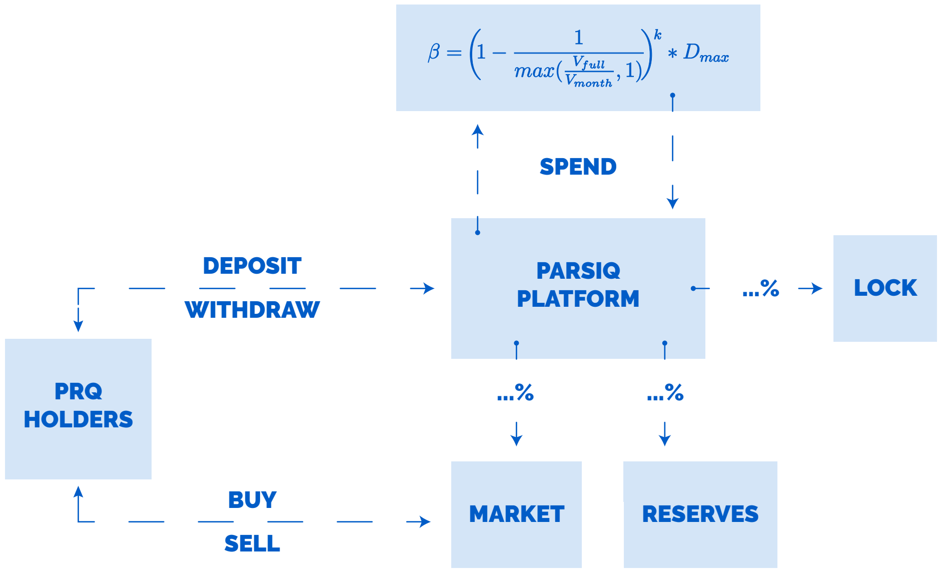
PARSIQ tokens are issued on the Ethereum blockchain and are of ERC-20 format.

Ticker: PRQ
Token type: ERC-20
Total Tokens: 500,000,000 PRQ
Available for Token Sale: 250,000,000 PRQ
Amount sold (IEO+Private sale): 1,695,000 EUR (Goal was: 5,250,000 EUR)
Unsold tokens were sent to the retention wallet and will be locked for up to 5 years
Max bonus (private sale/pre-sale/rounds): up to 30%



## PARSIQ TOKEN (PRQ) - 0XFE2786D7D1CCAB8B015F6EF7392F67D778F8D8D7

### TOKEN CIRCULATION SCHEME

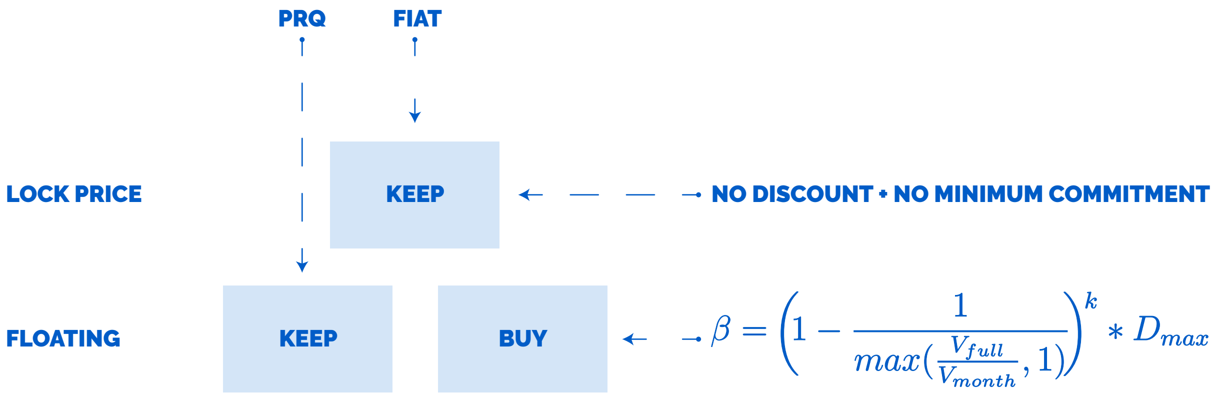

$\beta = \left(1 - \frac{1}{max(\frac{V_{full}}{V_{month}}, 1)}\right)^k * D_{max}$
SPEND
DEPOSIT
WITHDRAW
PARSIQ PLATFORM
...%
LOCK
PRQ HOLDERS
...%
...%
BUY
SELL
MARKET
RESERVES

A fraction of PARSIQ's profit margin from each payment that PARSIQ receives (in fiat or other cryptocurrencies) for providing its services will be converted to PRQ (if not received in PRQ tokens). Every time a user consumes the platform's services this fraction of tokens will being sent to the retention wallet where it will be locked for up to 10 years

$$\beta = \left(1 - \frac{1}{max(\frac{V_{full}}{V_{month}}, 1)}\right)^k * D_{max}$$

$$V_{full} \geq V_{month}$$

Where:
$\beta$ = discount
$V_{full}$ = assets deposited to the account
$V_{month}$ = planned token usage in service per month
$max$ = extremum, function is 1 if max
$k$ = smoothing parameter
$D_{max}$ = maximum possible discount (defined by PARSIQ)



PRQ

FIAT

LOCK PRICE

KEEP

NO DISCOUNT + NO MINIMUM COMMITMENT

FLOATING

KEEP

BUY

$$\beta = \left(1 - \frac{1}{max(\frac{V_{full}}{V_{month}}, 1)}\right)^k * D_{max}$$

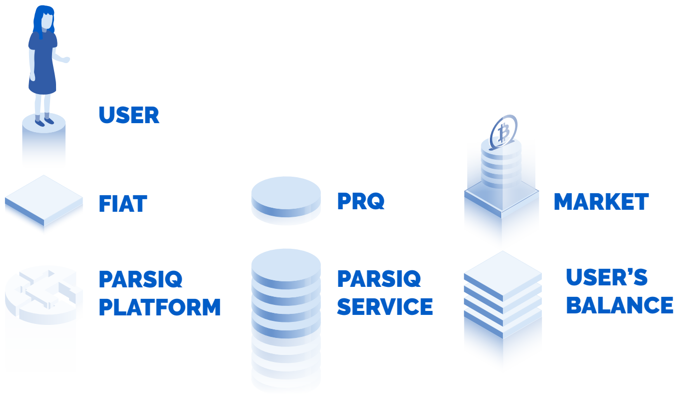
Discount is calculated at the beginning of a deposit period, if the deposit will be withdrawn before the end of the period, then a fine is applied. Fine is calculated fairly, as a difference between the initially claimed discount and the discount based deposit amounts vs usage.

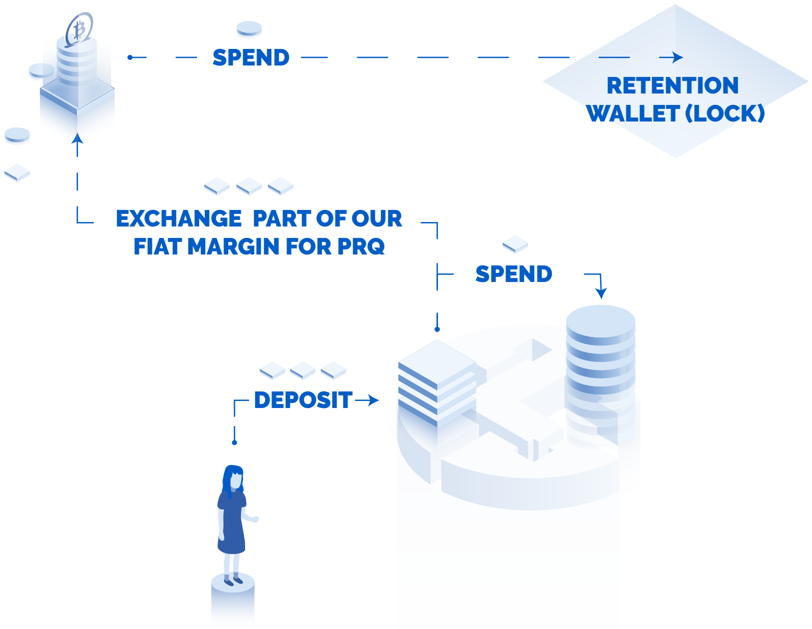
In Case 1 (*see below*), while the user consumes our services using fiat, we deduct a part of the fiat payment (from our margin) and use it to buy tokens from the market and lock them on the retention wallet. While when the user deposits PRQ tokens and uses them to pay for our services, we lock a certain amount of tokens directly.

# CASE 1

## USER DEPOSITS FIAT AND LOCKS THE RATE (NO CONVERSION TO PRQ)

There are no discounts for using the platform’s services

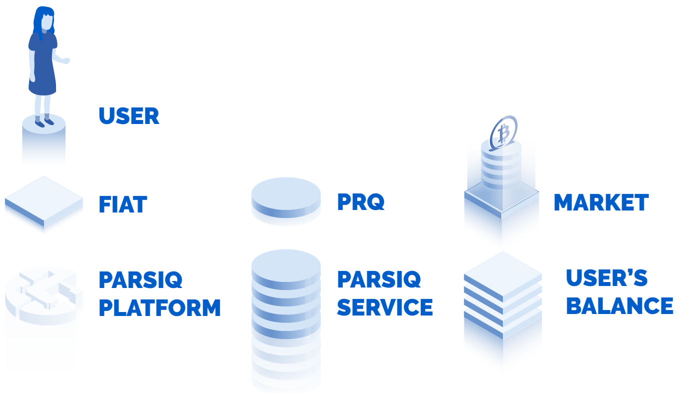


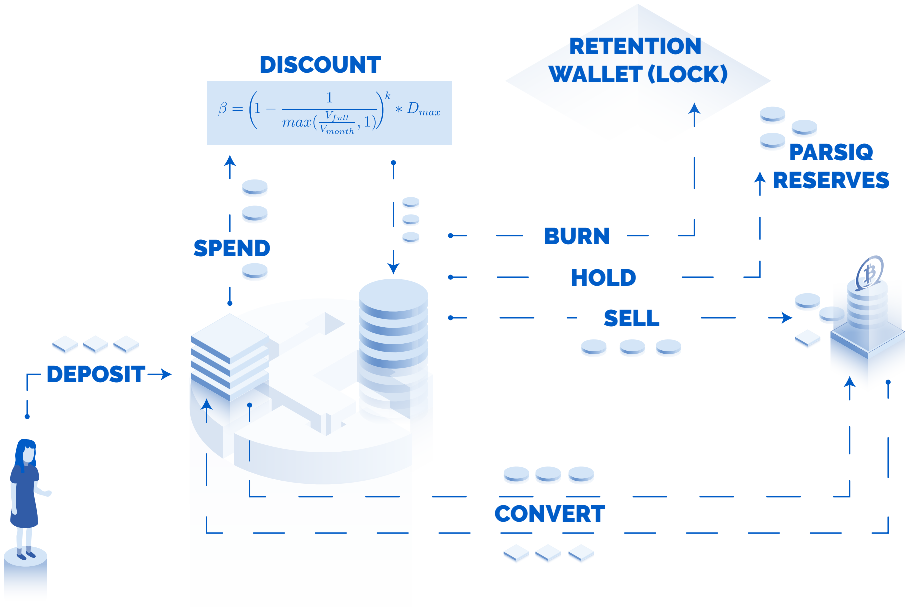


# CASE 2

## USER DEPOSITS FIAT AND TICKS THE BOX “CONVERT TO PRQ”

In this case the user is eligible for the discounts that our token utility provides. So when the fiat deposit is processed, the user is credited with PRQ tokens. And the discount he receives is processed according to the formula.



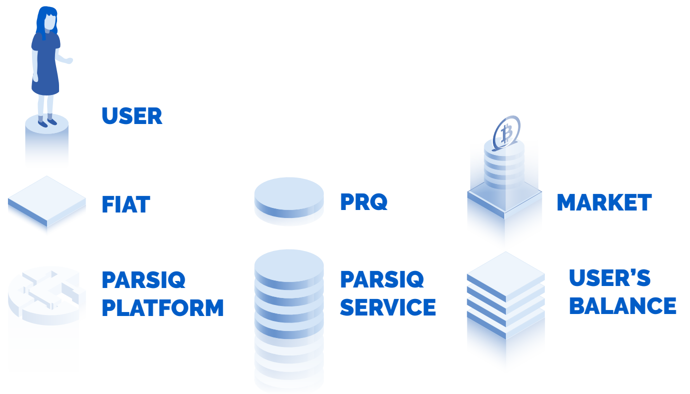


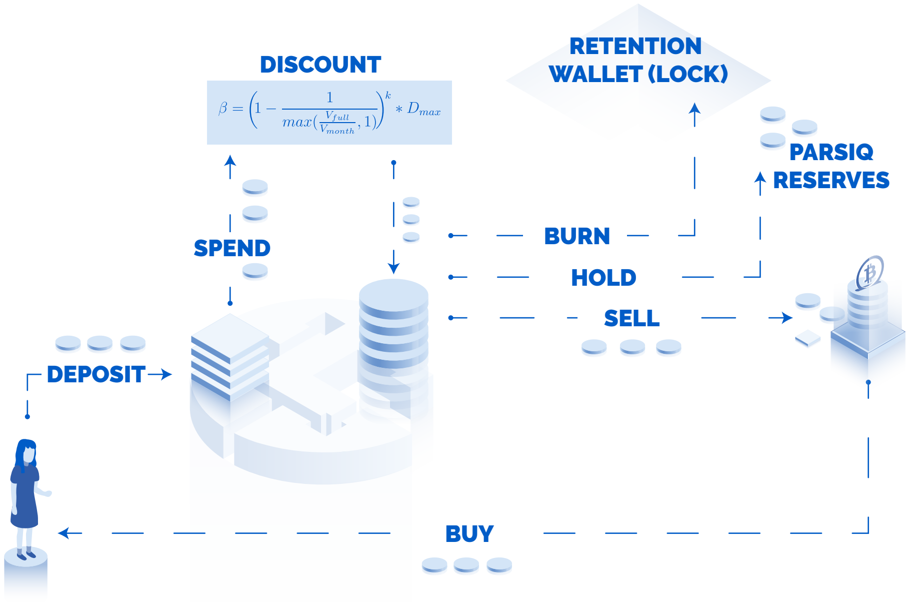
$$\beta = \left(1 - \frac{1}{max(\frac{V_{full}}{V_{month}}, 1)}\right)^k * D_{max}$$

# CASE 3

## USERS DEPOSITS PRQ AND USES PRQ

This is similar to Case 2, only that user deposits PRQ tokens directly.





$$\beta = \left(1 - \frac{1}{max(\frac{V_{full}}{V_{month}}, 1)}\right)^k * D_{max}$$