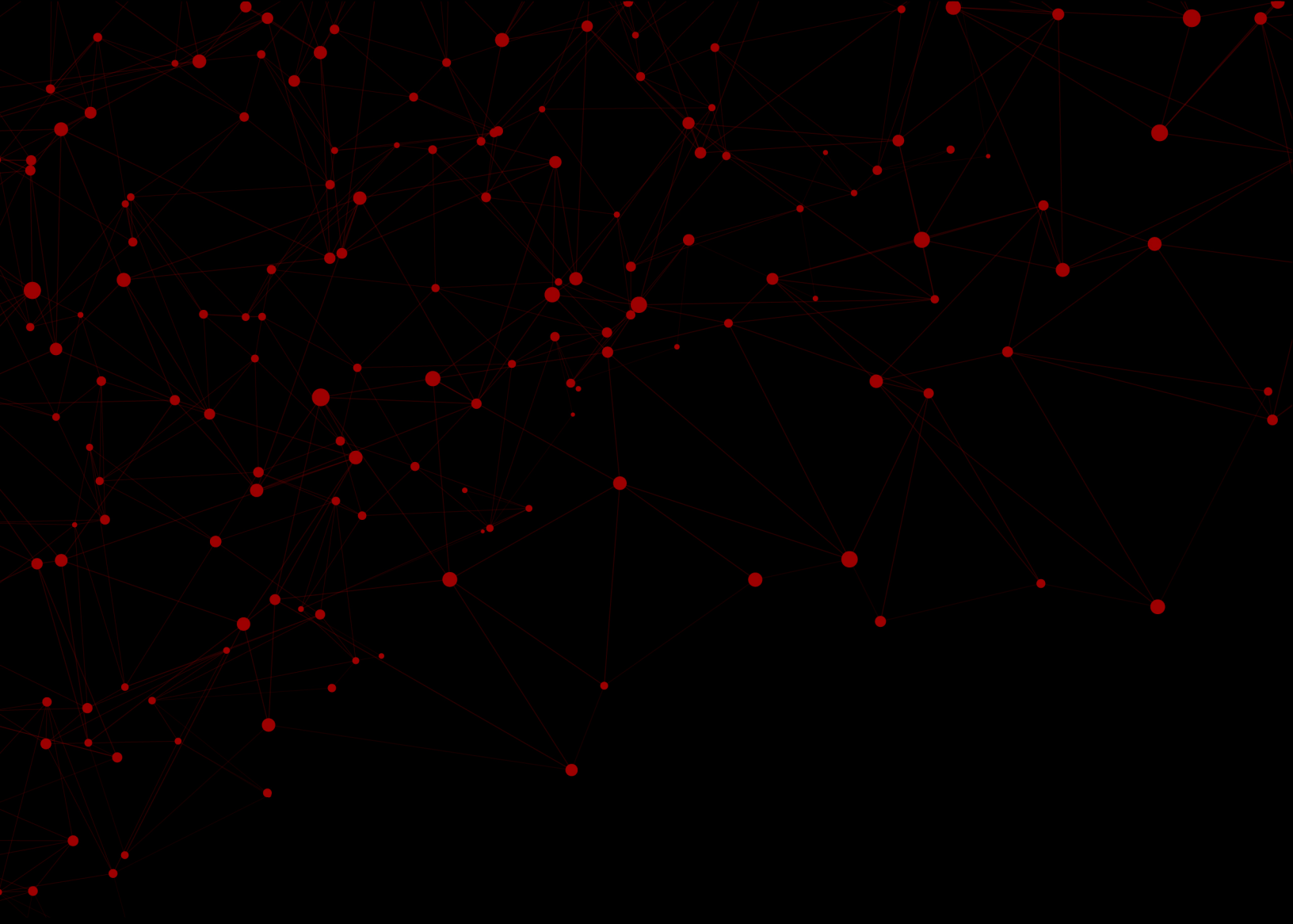


XAYA

LITEPAPER

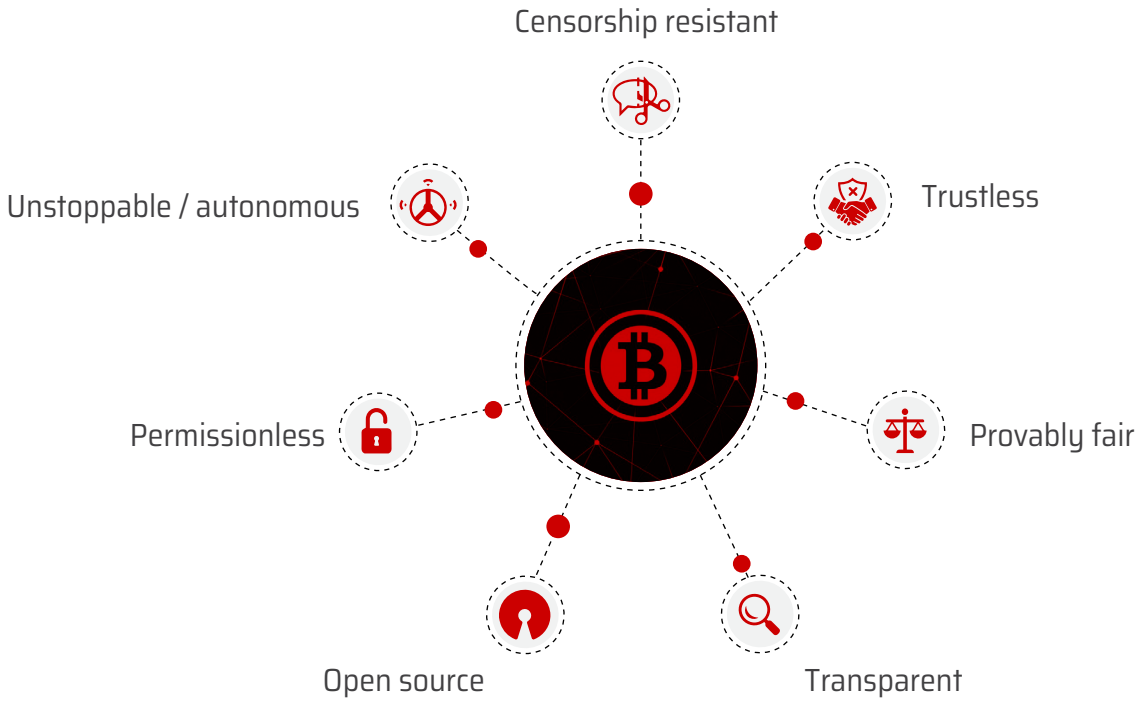
Contents

WHAT IS XAYA?

Xaya is a blockchain and platform designed to solve the scalability issues associated with blockchains and in-particular for Decentralized Blockchain Gaming.

The Xaya architecture strives for elegance through simplicity.

Decentralized Blockchain Games follow the principles and ethos of Bitcoin in that they are:



To keep this paper “lite”, read our high level reasons for “Why Decentralized Gaming” here:

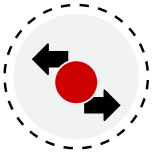
https://xaya.medium.com/why-decentralized-gaming-wip-fc22767e5f4d

How Xaya Games Work on a High Level

More details on how Xaya games work is later in this document. The following is a quick primer to get started in the next sections:

The way that Xaya games work is simple and is as follows:

Players create special “tokens”(accounts) that are human readable and unique within the Xaya blockchain (or other chains using smart contracts)



Players make moves from these accounts by signing special transactions with game move data in them

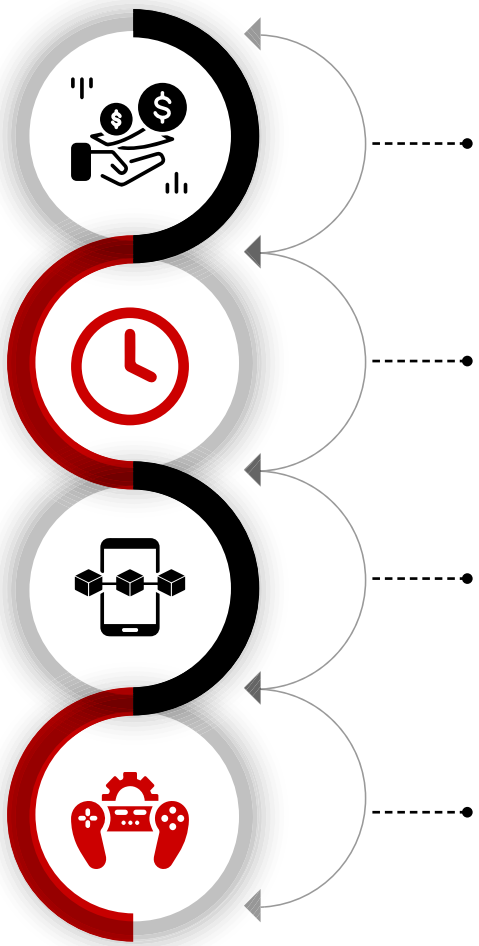
A separate application called a Game State Processor (GSP) processes all moves from each block for a particular game and computes the state of the game (gamestate)

The frontend receives the state from the GSP

More details are in the *Architecture* section.

PROBLEMS WITH CURRENT TECHNOLOGIES

Blockchain technology suffers from 4 main issues with regards to blockchain gaming:



Expensive transaction fees

- Gas related to computational complexity and state storage limits the complexity of the games

Slow block times

- Unsuited for most modern games
- Each move must be mined before the action takes place
- Limited TPS or very large blockchains

Blockchain UX

- Hard to on-board non-blockchain gamers
- Wallets, etc.

Difficult to develop complex games

- Learn new programming languages

SOLUTIONS AND ARCHITECTURE

Decoupling State Computation from Blockchain Consensus

Many Highly Complex Games

Transaction fees with blockchains are an unfortunate necessity. Their primary goal is to prevent spam. As data is stored by all nodes, this requires a cost.

Ethereum, for example, also adds a cost for computation based on the complexity of smart contracts, as computing the states of millions of smart contracts is expensive. This is something every full node does. Similarly, writing state updates of a smart contract to network storage is one of the most expensive operations in terms of gas.

This limits how complex a game can be. As for one example, computing checkmate in a game of chess through a smart contract could be extremely expensive and thus unplayable.

Xaya's solution to this is simple:

Gamestate computation is detached from the blockchain nodes.

This means that miners and those running a full node do not need to compute the state of a game or application, and only verify that the blockchain transaction itself is valid, not if it is a valid action in the game.

This keeps transaction fees low as the fee is only related to the size of the transaction as opposed to the complexity of the game or application.

Who Computes the State?

Any one who is interested in the game can compute the state locally based on the actions in the blockchain.

This is performed by a completely separate application called the **Game State Processor (GSP)**.

Any one who is using the same GSP will have the same state of the game.

This allows for games and applications to have the computational complexity of "normal" games whilst being decentralized and having the security of blockchain technology.

This method also allows for any number of decentralized applications to run simultaneously without requiring Xaya nodes to compute the state of every game or application.

It's also worth noting that even if no one is running the GSP and no one is computing the state of the game, that does not affect the security or compromise the state of the game.

Any one at any time can run a GSP and sync from the blockchain to the current state. There is no P2P network between the GSPs, only between the blockchain nodes that do not need to know about the states of any games or applications.

This method also allows GSPs to be **blockchain agnostic** - see more information later in this document.

Blockchain Performance

Although the above method of decoupling state computation allows for complex applications, it does not help improve the issue of raw blockchain throughput.

Only a limited number of transactions (or moves) on-chain are possible.

Fully on-chain games will be limited due to:

- Slow block times
- Blockchain bloat
  - Blocks are limited in size so eventually fully on-chain games will become slower and/or fees will become higher
  - The entire blockchain can become large and potentially not practical for those who wish to go "fully decentralized" and trustless to run and store locally

For some game genres this may be acceptable.

To solve this, Xaya uses a technology we call **Game Channels**

Game Channels

Game Channels allow for real-time decentralized and trustless gaming while reducing blockchain bloat.

Game channels are similar to payment channels on Bitcoin and state channels on Ethereum, only designed specifically for decentralised gameplay.

On a high level they work like this:

1. Alice creates a special transaction to open a channel
2. Bob creates a special transaction to join the channel

They both then connect P2P (or through some relay) and sign their moves and state to each other in a turn

based fashion (this can happen in real time and be done automatically by their computers, without explicit user interactions).

A game can then be played with an unlimited number of actions between them in real time.

Once the game is over, the winner can prove this and receive the reward or points depending on how the game is designed.

This method opens up the possibility for game genres that require fast real-time gameplay to be trustless and secured by blockchain technology, while also reducing blockchain bloat and transaction fees (there are no transaction fees in the channel).

As channels do not interact with other channels, there is no limit on the number of parallel channels and this allows for almost unlimited throughput for game channel games.

It is also worth noting that game channel games can be part of an overall fully on-chain game, e.g. a strategy game with game channel mini games, and the outcome can affect the state of the overall game, e.g. you attack another player on a strategic map, a real-time game follows, if you win, you take over the land. Similar to “Total War”-style games.

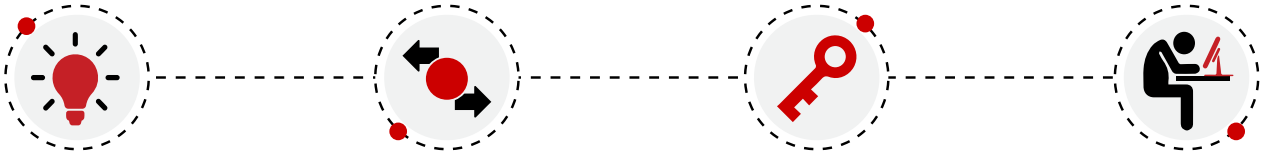
The game channels paper was published in 2015 and peer reviewed in 2016.

You can read the original paper here:

https://www.ledgerjournal.org/ojs/index.php/ledger/article/view/15/64

Onboarding Mainstream Gamers

The majority of the billions of players in the gaming market have had little to no exposure to crypto currencies. This makes on-boarding difficult due to the added complexity:



Learning wallets

Signing transactions each time you make a move

Backing up your private keys (seed phrases / private keys)

Uninteresting or trivial games

Xaya technology allows for deep, complex and fun games that resemble games players are used to, not only browser based games but standalone games using game engines such as Unreal and Unity. And of course still keeping the principles of decentralization intact.

Xaya games are playable with 2 options:

1. Fully decentralized mode
   - Requires a fully synced Xaya (or other base chain) node
2. Lite mode
   - Electrum (or Metamask) wallet (user keeps private keys)
   - Receives state from centralised servers
     - These can be run by the user to stay decentralised or by a trusted friend

Fully decentralized mode is for those who wish to play 100% trustless. This adds some complication to the initial setup although it is still relatively simple to set up.

Lite mode is designed to bring the feel of any “normal” game to the user.

Lite mode should allow users to only need to learn one thing: “write down their seed phrase”. Although this is not perfect, this is just something that must be done, and something that is not just a hurdle for blockchain gaming, but for cryptocurrency adoption in general.

Xaya games have a “1-click to run” UX and allow actions in the game to happen without requiring any blockchain knowledge, or even knowing that it is a blockchain game.

More details are in the *Architecture* section.

Building Games in Any Programming Language

Developers wanting to build blockchain games are generally limited to the programming languages that they already know and can use.

Using the Xaya SDK, it is possible to build games in almost any language the developer wishes. Although primarily written in C++, it is possible to use wrappers for many programming languages.

The Xaya platform provides the libxayagame library as part of its SDK.

https://github.com/xaya/libxayagame

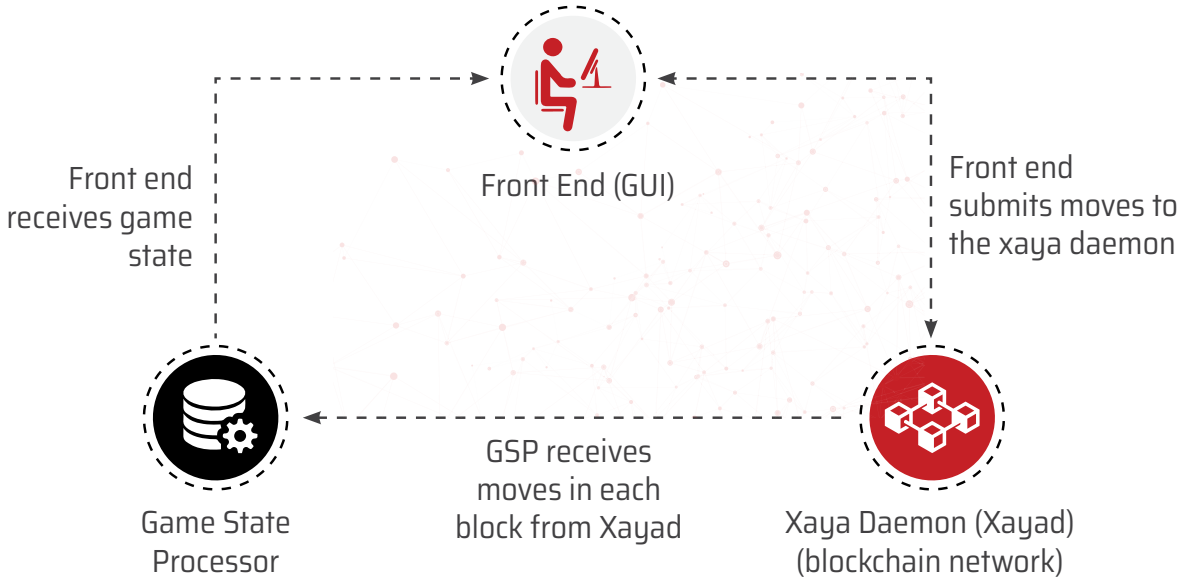
libxayagame includes and takes care of all the blockchain communication, reorganizations, and other blockchain complexity so that the game (or dapp) developer may focus solely on the game logic.

ARCHITECTURE

There are 3 primary components in the Xaya architecture:

1. Xaya Core Blockchain (or other base layer chain such as Ethereum or Polygon)
2. Game State Processor (GSP)
3. Front end (GUI)

This is most simply represented as illustrated below.



The way that games work is simple and is as follows:

1. Players create special names (accounts) which are human readable and unique within the Xaya blockchain (or other chains using smart contracts)
2. Players make moves from these accounts by signing special transactions with the move data in them using their wallet. This is stored in the base chain.
3. The GSP processes all moves from each block for a particular game and computes the state of the game
4. The frontend receives the state from the GSP

This approach is elegant and simple. There are no validators or complex systems in place, yet games and applications can run fully decentralized and are secured by the base layer blockchain.

As the base layer is also simple, this can be replaced with almost any blockchain and thus the Xaya technology and platform is truly **blockchain agnostic.**

You can read more in this blog here:

https://xaya.medium.com/xaya-blockchain-agnostic-polygon-b0b3d29cccb3

Full Nodes

As illustrated above - there is of course a drawback to this configuration. Running a full node can be difficult or off-putting for most people due to the time to sync the chain and the size of it, and also the added complexity of having to run a separate wallet as well as the game.

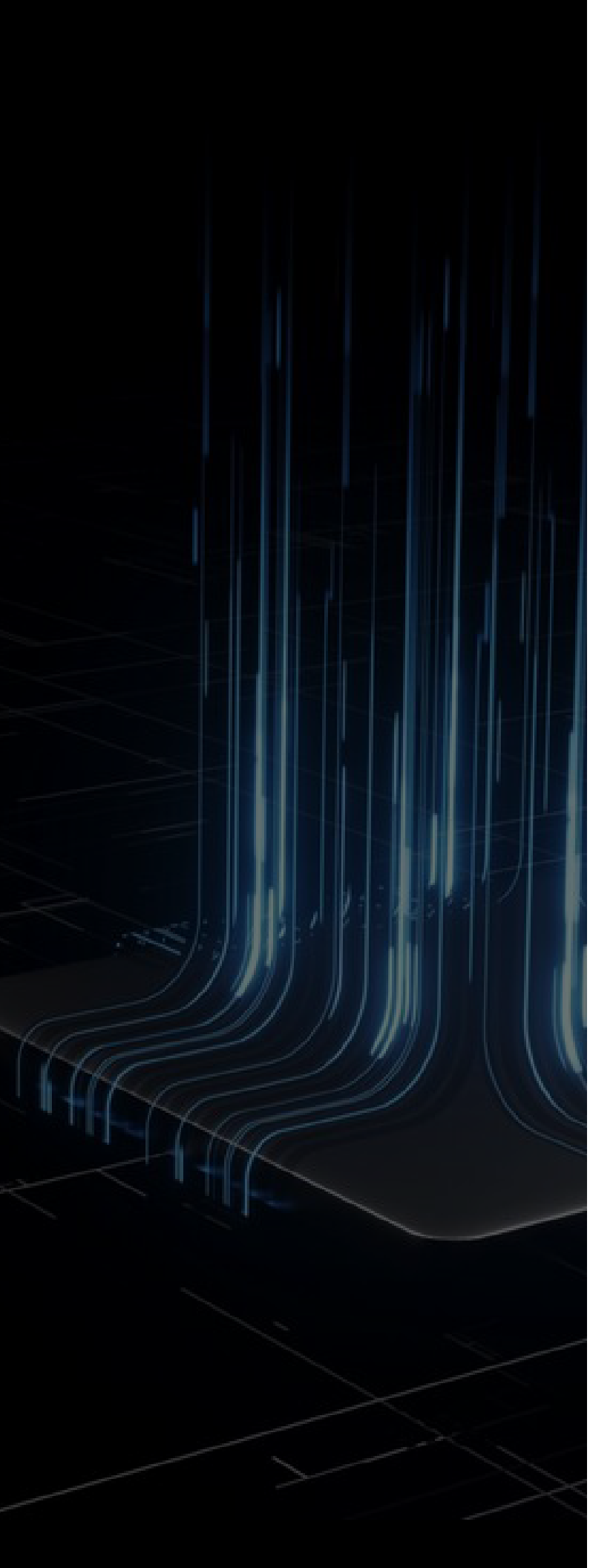
Also, it's not possible to run a full node on most mobile phones, which limits the platforms Xaya games can run on given the above configuration.

As a solution to this, we have developed and utilise open source applications and components so that players need not download the blockchain (for lite clients).

Lite Clients

Lite clients use several components. These include:

- **Electrum SPV wallet:** https://github.com/xaya/electrum-chi
  - A fork of the Bitcoin (or Namecoin) Electrum wallet.
  - Very light, yet secure
  - It has been modified to allow name (account) functionality as with the Xayad full node daemon
  - This lite wallet can be replaced with Metamask or other blockchain lite wallets for different base chains



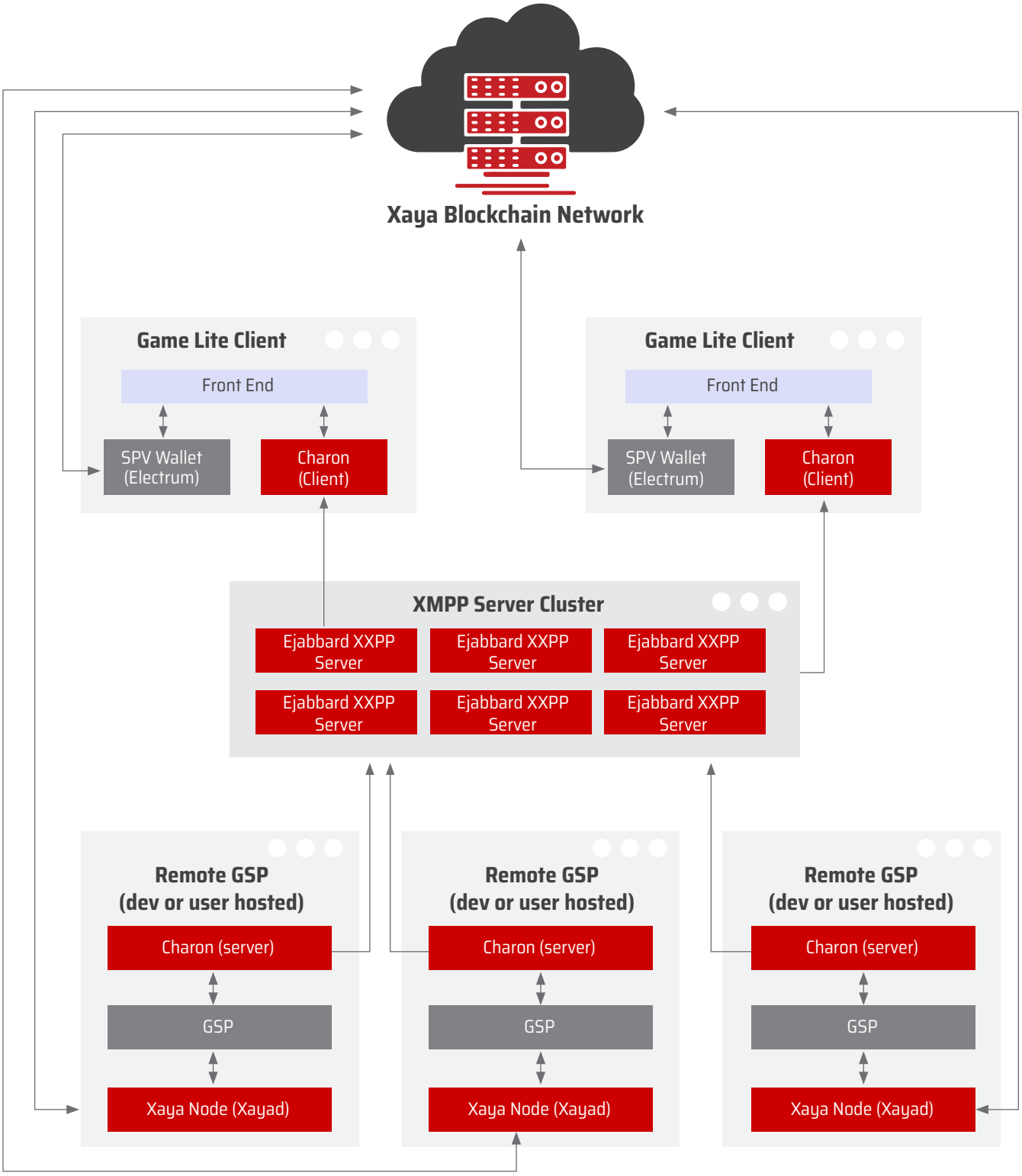
- **Charon:** https://github.com/xaya/charon/blob/master/doc/protocol.md
  - "Charon allows users to access game-state data from a GSP running on a remote server through XMPP. In other words, one or more GSP processes can connect to an XMPP server and listen there for requests, and players can connect as well using a custom XMPP client, which is then able to send requests to the GSP connections."

- **XID:** https://github.com/xaya/xid
  - "Xaya ID (XID) is an application built on the XAYA platform that turns each Xaya name into a secure digital identity."
  - "These identities are meant to be used inside the Xaya ecosystem, e.g. on chat systems or market places in Xaya games. They can, however, be used by any other application as well. For instance, websites can enable "login with Xaya" to use secure, password-less authentication. Or messaging systems can use Xaya identities for the secure exchange of key fingerprints for end-to-end encryption."

- **XMPP Server (Ejabberd) > https://ejabberd.im/** (note: this is not on the users device)
  - A high performance messaging application
  - Utilises a plugin for external authentication with XID name associations for users to login to the system without requiring the server to save any username or passwords. Essentially, Xaya accounts can log in securely onto the XMPP server.
  - https://github.com/xaya/xid/tree/master/ejabberd

All of the components are open source.

By utilizing these components, we allow users to be able to play and run games with one click while also allowing the user to keep possession of their private keys. This scalable architecture is illustrated in the diagram below:



Sacrificing Decentralisation for Lite Mode?

For games being developed (or co-developed) by the Xaya team, we provide the servers (remote GSPs and XMPP) that provide the state of the game to the lite clients. This adds a level of trust in that the player needs to trust that we will send them the correct state of the game.

This is essentially sacrificing some trust for convenience. We believe that at least initially this will be the preferred choice for most users.

However, due to how this architecture has been designed, it is possible for users/players to deploy their own "remote GSPs" to allow for trustless lite clients.

With a little technical knowledge it is possible to set up the "remote GSPs" with 1 command using the provided docker containers. This can be done on a <$5 VPS.

What if you don't have the motivation or technical know-how to do this?

Any one can run a remote GSP and provide the service (paid or for free), so it's possible to have multiple independent providers (or even friends). And it's also possible to receive the states from multiple independent sources at the same time to "compare".

Can We Make "Lite Mode" Trustless and Secure, and Still be Lite?

Yes, this can be done using a system not dissimilar to DPoS style blockchains.

As part of the game logic (in a GSP) it's possible to delegate service providers to distribute signed states of the game and be rewarded for doing so (e.g. through staking in-game currency).

Validators (or those also running a full node) can check/verify that these states provided are correct.

If a provider is found to be malicious (sending incorrect game states) then the validator can prove this as he was sent a signed state of the game. The provider can then be punished by taking some of their staked in-game currency.

Although this is not implemented, it would not be difficult to do so once it becomes a concern.

For more information

www.xaya.io

Join discord >www.discord.com/invite/VQQ6TC3

Join telegram > www.t.me/xaya_en