

Session 5

Navigator sessions

Date	Topics	
15/12/23	o1js review	
19/1/24	Development workflow, design approaches, techniques and useful patterns	
26/1/24	Recursion	
9/2/24	Application storage solutions	
1/3/24	Utilising decentralisation	
15/3/24	zkOracles and decentralised exchanges	
5/4/24	Ensuring security	
26/4/24	Review Session	

Today's topics

- What's new
 - Protokit continued
 - Decentralisation
-

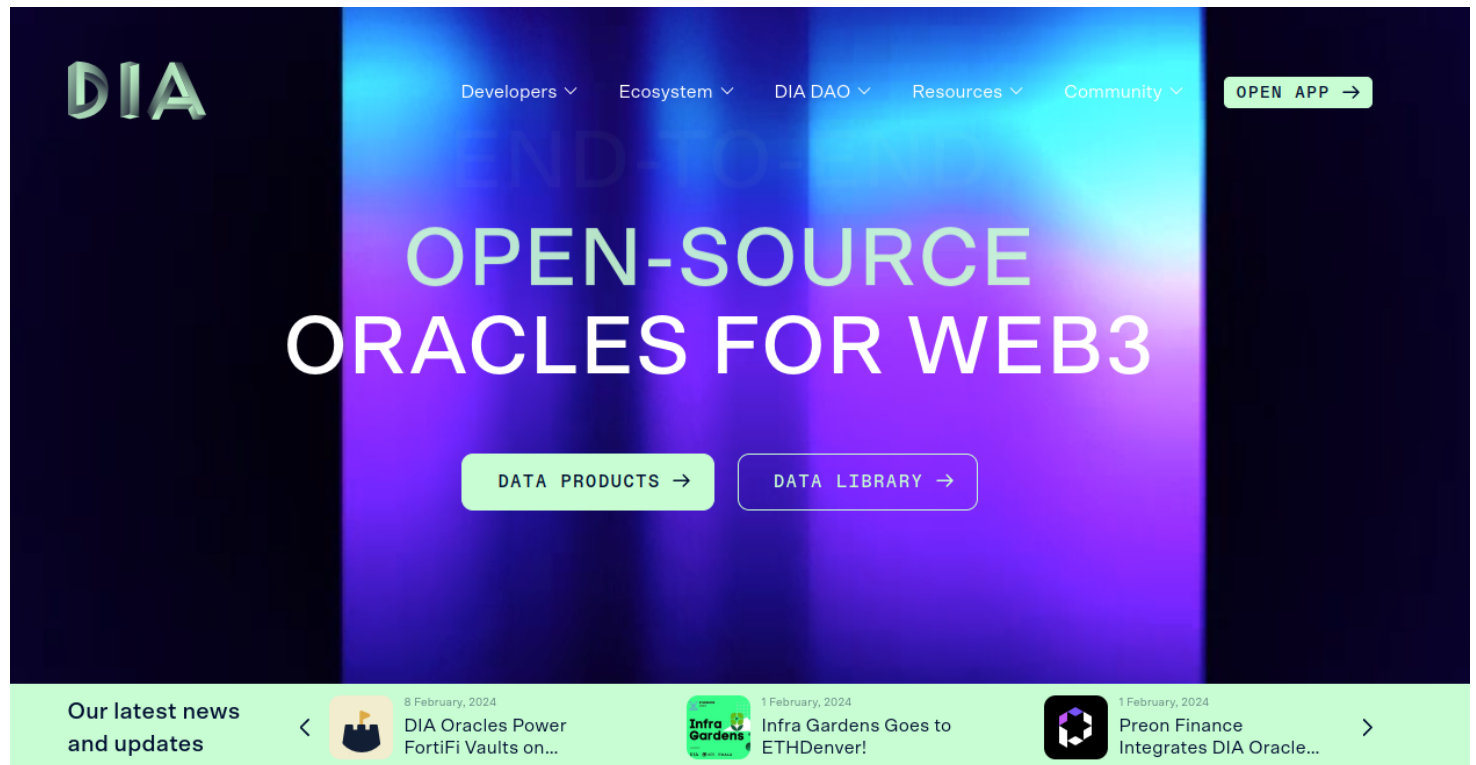
What's New

Latest [Blog](#) for February updates

Road to upgrade

Upgrade Mechanism Testing (UMT), one of the final stages of testing, will officially kick off on Monday, February 26 at 12pm UTC, and will run for an initial 14-day period

DIA Data



See [Blog](#) post

- POC using zk circuits

Blockberry API has launched

See [Announcement](#)

"Blockberry is an open API platform providing an extensive set of indexed blockchain data"

See [Mina quickstart](#)

"Blockberry MinaAPI is broken down into sections based on the entities we get from the Mina Blockchain and index ourselves, among which are the following:"

[Accounts](#)

[Transactions](#)

[Blocks](#)

[Analytics](#)

[Mips](#)

[Time Locks](#)

getAccounts

GET https://api.blockberry.one/mina-mainnet/v1/accounts

Get a list of all accounts on the Mina Blockchain.

🔗 LOG IN TO SEE FULL REQUEST HISTORY

TIME	STATUS	USER AGENT
Make a request to see history.		
<div><div></div><div></div><div></div><div></div></div> 0 Requests This Month		

QUERY PARAMS

page int32 required

Queried API page.

0

size int32 required

Number of queried entries.

20

orderBy string required

Sorting method: from the lowest element to the highest (ASC) or from the highest element to the lowest (DESC).

DESC

sortBy string required

Select sorting parameter.

BALANCE

LANGUAGE

Shell

Node

Ruby

PHP

Python

AUTHORIZATION

HEADER

Header

x-api-key

CURL

REQUEST

```
1 curl --request GET \  
2 --url 'https://api.blockberry.one/mina-mainnet/v1/  
3 --header 'accept: application/json'
```

Try It!

RESPONSE

EXAMPLES

Click **Try It!** to start a request and see the response here! Or choose an example:

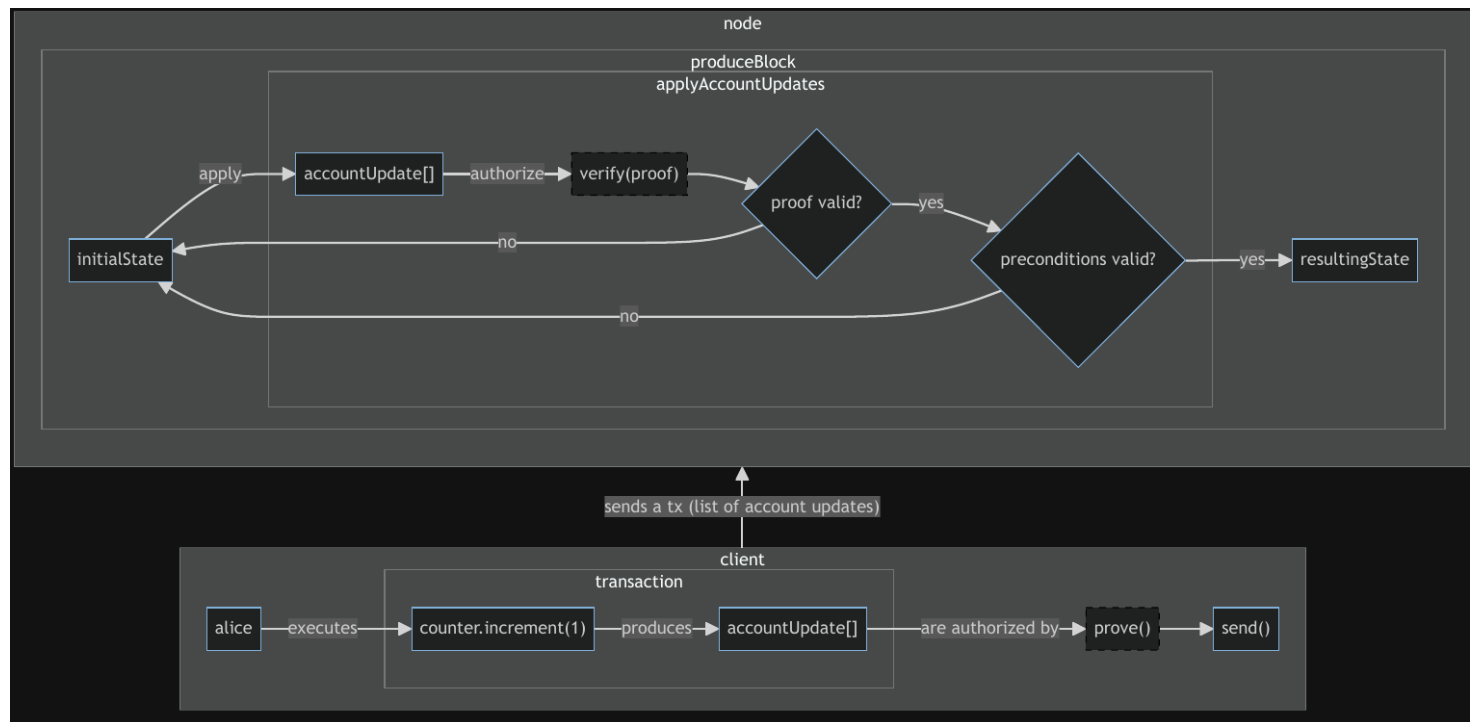
application/json

200

Protokit

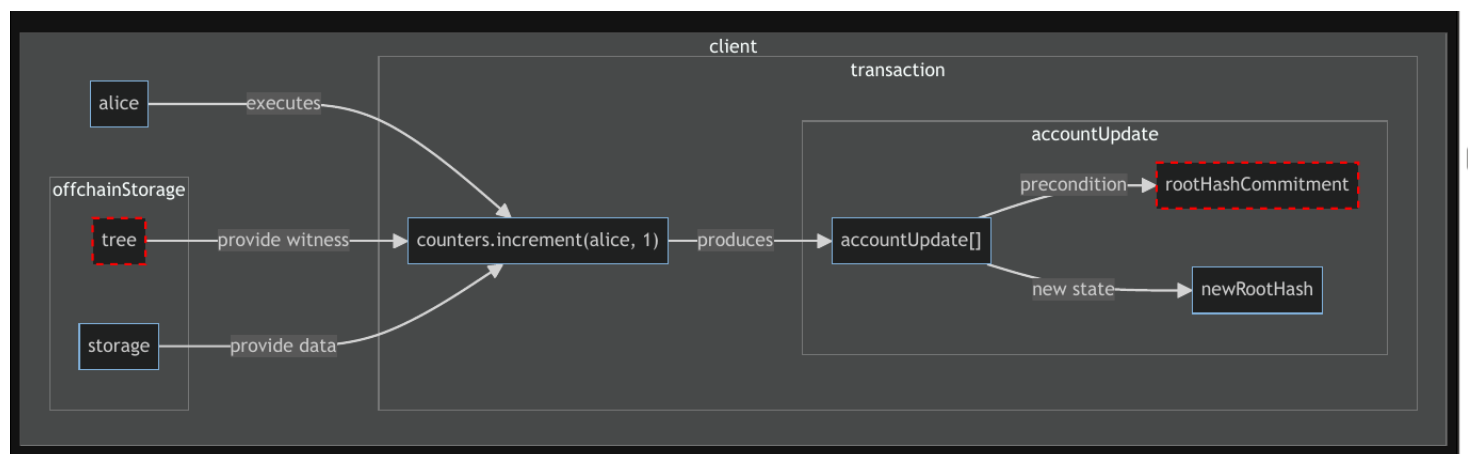
From Protokit slides

Current process on Mina



Since users are executing the code off chain without implicit reference to other users we run into the problem of synchronising updates without race conditions.

Two users looking to update the same piece of state would see the same preconditions, the first transaction to hit the chain would invalidate the second, since the precondition for the second one will have changed.

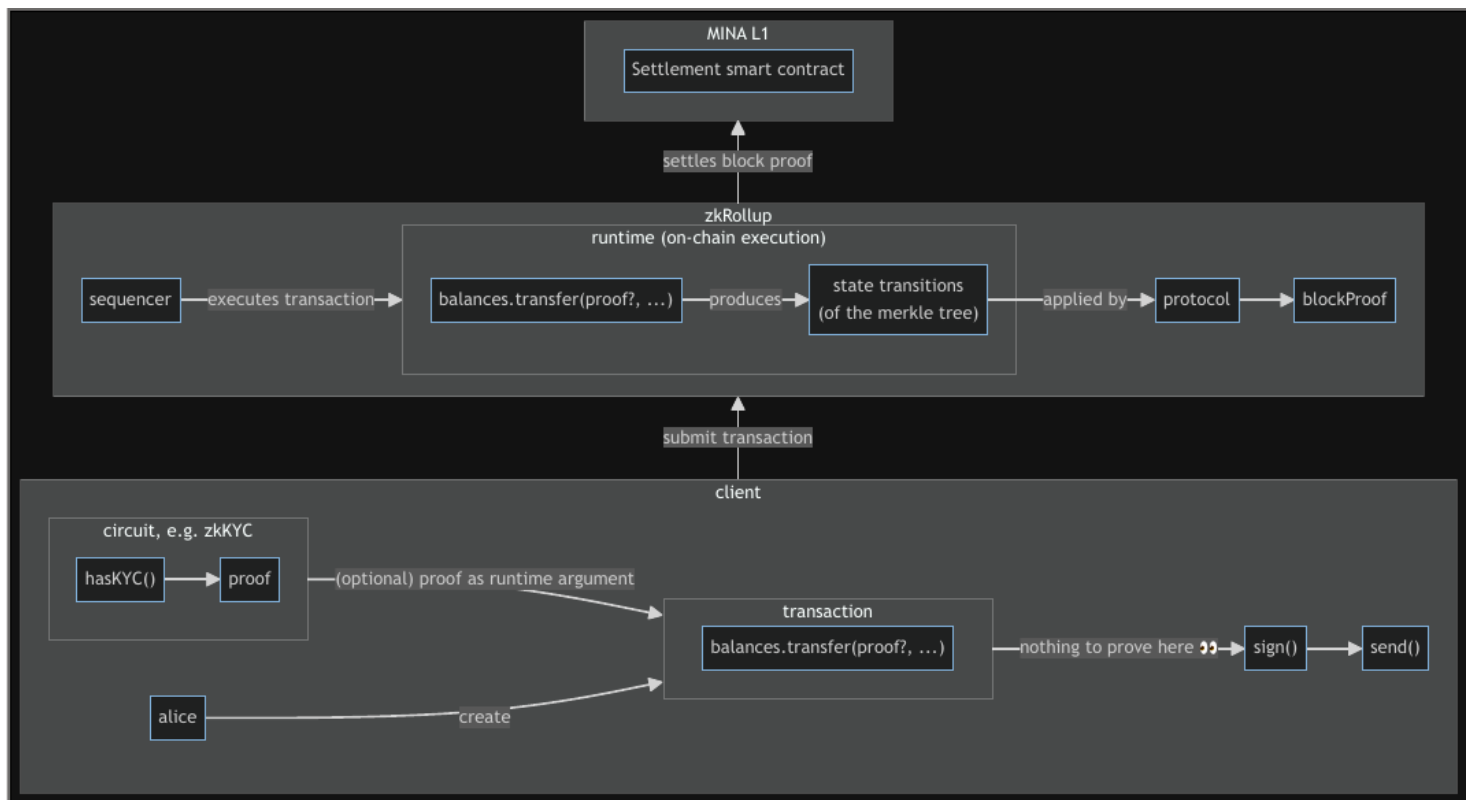


Protokit Features

- Custom built zkVM
- Hybrid execution on and off chain
- Sequenced transactions
- Modular - allowing customisable app chains
- Large amount of on chain storage (State / State Map)

When you create an app chain with Protokit, you get

- Sequencer - controlling mempool, block production and interaction with L1
- Execution environment
- L1 contract for settlement



Differences to 01js

As you can see in the example, we have specific additions from protokit

- State and StateMap
- assert
- there is a query api available to allow you to get state from the chain

Protokit and privacy

Because of the hybrid execution module we can have private computation being done off chain, with a proof of that being sent on chain.

The sequence would be

- Perform off chain computation with private inputs to create a proof , and a public output.
- Submit the proof to the sequencer as part of the transaction
- The runtime module can verify the proof
- The proof's public output can be used on chain for further business logic.

Using Protokit

Starter kit

See [Repo](#)

Setup

```
git clone https://github.com/proto-kit/starter-kit my-chain
cd my-chain

# ensures you have the right node.js version
nvm use
pnpm install
```

Running the sequencer & UI

```
# starts both UI and sequencer locally
pnpm dev

# starts UI only
pnpm dev -- --filter web
# starts sequencer only
pnpm dev -- --filter chain
```

Navigate to `localhost:3000` to see the example UI, or to `localhost:8080/graphql` to see the interface of the locally running sequencer.

Resources

[Video](#) - Goes through an example of a private airdrop

Note the use of the nullifier pattern to break the link between accounts

Protokit 101 [thread](#) on Discord

Mina and rollups

This [article](#) also discusses Mina and rollups

Their conclusion

"Zero knowledge technology is one of, if not the most, innovative solutions in crypto. It's primarily been used for scaling purposes. However, new solutions like Mina have emerged that extend zero knowledge's capabilities, creating an incredible opportunity to bring real world data on-chain while leveraging scaling potential.

Ultimately, all projects should be looking into zkRollups as a potential end-state scaling solution for their particular chain.

Further development is needed for zkEVMs to overcome the inherent challenge of building a performant solution while remaining EVM compatible. Still, we're excited to see what is built and developed when zero knowledge is fully available and relevant for all developers."

Decentralisation

Goals of decentralisation

- Participation
- Diversity
- Conflict resolution
- Flexibility
- Moving power to the edge (user)

Scalability and centralisation

Even though this is ostensibly the goal for all blockchains, in practice participation may not be feasible for many people.

As the hardware requirements to run a node increase, nodes that fail to meet the requirements will be unable to process transactions quickly enough, they will fall behind, lose consensus and effectively fall off the network, leading to an increase in centralisation.

From [Roadmap trust minimisation documentation](#)

"We imagine a world where everyone through their phones and other digital devices plays a role in democratically securing Mina. That way, we can each play a part in securing and decentralising the services we rely upon and use."

...

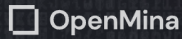
Mina was developed with the purpose of taking blockchain's solution to trust minimization and extending it to be both sustainable and scalable. By recursively rolling up all of its blocks and transactions into a single proof, Mina's state remains accessible and instantly verifiable over time—independently of how much throughput is being processed by it and its zkApps.

...

This track exists to take this even further: To make the protocol even more invulnerable to attacks and bias pressing from centralization and add decentralized data storage to Mina's feature set.

We imagine a world where everyone, through their phones and other digital devices, plays a role in democratically securing Mina. That way, we can each play a part in securing and decentralizing the services we rely upon and use.

Web node



With the Web Node, anyone can verify blocks and transfer funds directly through their browser

An in-browser Mina node capable of validating blocks

Blog post about [Web Node](#)

If you are interested in what goes into syncing a node, there is this [guide](#)

From Vitalik's [blog](#)

Showing traditional stacks and their decentralised counterparts

Traditional stack	Decentralized stack
Banking system	ETH, stablecoins, L2s for payments, DEXes (note: still need banks for loans)
Receipts	Links to transactions on block explorers
Corporations	DAOs
DNS (.com , .io , etc)	ENS (.eth)
Regular email	Encrypted email (eg. Skiff)
Regular messaging (eg. Telegram)	Decentralized messaging (eg. Status)
Sign in with Google, Twitter, Wechat	Sign in with Ethereum , Zupass, Attestations via EAS , POAPs, Zu-Stamps... + social recovery
Publishing blogs on Medium, etc	Publishing self-hosted blogs on IPFS (eg. using Fleek)
Twitter, Facebook	Lens , Farcaster ...
Limit bad actors through all-seeing big brother	Constrain bad actors through zero knowledge proofs

It would seem that there is plenty of scope for decentralising systems.

What does decentralisation give us ?

- Increased participation
- Participation requiring fewer resources

What can developers do to ensure decentralisation

- See centralisation as a risk
 - Flag it in audit reports
 - Large percentage of exploits are rug pulls

Increased participation and the network

- Because of Mina's architecture increased participation scales better than on other chains. downloading 22kb does not require much bandwidth.
- Ethereum with Danksharding will have 30Mb blocks

Effects of increased participation

- We can routinely check integrity of systems
- Checking of corruption in systems becomes easier

We also need to think about the 'unhappy' path of verification.

What would we do ?

- stop using system ? - a negative response
- have robust governance to fix problems

Transparency versus privacy

Although these may look like mutually exclusive ideals, zkp tech does allow us to have both.

"Human dignity demands that personal information, like medical and forensic data, be hidden from the public. But veils of secrecy designed to preserve privacy may also be abused to cover up lies and deceit by institutions entrusted with Data, unjustly harming citizens and eroding trust in central institutions." - Starkware

To take this further, the fact that we have effective privacy in a system, can give us confidence to allow greater transparency.

Centralising forces

Some chains are seeing centralising forces because of their cryptoeconomics.

There is concern that we will see increased censorship, for example Tornado Cash

Mina is uniquely placed in the web3 space to foster decentralisation.

We need to champion the opportunities which Mina provides.

Interesting projects / ideas

General ideas

- Prediction markets
 - Wisdom of crowds
 - Wisdom of AI crowds ?
 - Disintermediation
 - Shared ownership - really tokenising everything
-

Some existing / proposed projects on Mina

From Cohort 3

[Mina Login](#)










[zkCoudworker](#)

[Rate limiting nullifier](#)

Examples from previous cohorts

- zkLocus

See [Site](#)

 Geolocation Authentication With Full Privacy Prove your whereabouts within specific regions, within a specific time interval, without exposing your exact coordinates or timestamp with zkLocus.	 Inherent Zero-Knowledge Security zkLocus isn't just using zero-knowledge technology; it is zero-knowledge from the ground up. Every circuit is designed to ensure privacy and security are not just features, but fundamental attributes of the application.	 Native on the Mina Blockchain zkLocus was born on, and is natively implemented in O1JS on the Mina Blockchain. It can be easily integrated into any zero-knowledge application.
 Blockchain-Agnostic zkLocus generates zkSNARKs proofs which can be used on any blockchain, such as Ethereum and Cardano. It's fully EVM-compatible. This flexibility, rooted in zkLocus' vision, ensures wide applicability for decentralized systems.	 DeFi Geolocation Programmability zkLocus brings geolocation programmability to DeFi, enabling the embedding of secure, private location data into NFTs, ERC-20 tokens, and other digital assets.	 Authenticated Geolocation Source Whether the zkLocus, the source of geolocation data is fully authenticated through zero-knowledge proofs, ensuring complete privacy of the underlying location details, while making its source fully transparent.
 Selective Coordinate Sharing While zkLocus champions privacy, it also provides the option to share exact coordinates securely when transparency is required. The same is true for timestamps.	 Runs On Mobile Geolocation proofs can be generated on any mobile device, including directly on the web browser app, such as Chrome, Safari, and Firefox.	 Flexible Geographical Assertions Craft assertions about being within or outside specific areas with zkLocus. Define arbitrary polygons or merge areas to create custom geofences for nuanced location proofs.

- DAO voting with [zkSNAP](#)
- [Sealed Bid](#) auctions
- Verifying Bitcoin [headers](#)