Q0 Which stream do you choose (answer with A or B)?

Stream B!

Q1 Scaling the future

1. Based on the above, a number of solutions have been proposed to solve this trilemma. Briefly describe the different scalability solutions and write pros and cons of each approach.

State Channels

```
State channels allow participants to transact x number of times off-chain
while only submitting two on-chain transactions to the Ethereum network.
This allows for extremely high transaction throughput.
```

pros	cons
Instant withdrawal/settling on Mainnet.	Time and cost to set up and settle a channel.
Extremely high throughput is possible.	Need to periodically watch the network or delegate this responsibility to someone else to ensure the security of your funds.
Lowest cost per transaction.	Have to lockup funds in open payment channels.
	Don't support open participation.

https://ethereum.org/en/developers/docs/scaling/state-channels/

Sidechains

```
A sidechain is a separate blockchain which runs in parallel to Ethereum
Mainnet and operates independently. It has its own consensus algorithm
```

pros	cons
Established technology.	Less decentralized.
Supports general computation, EVM compatibility.	Uses a separate consensus mechanism. Not secured by layer 1 .
	A quorum of sidechain validators can commit fraud.

https://ethereum.org/en/developers/docs/scaling/sidechains/

Plasma

```
A plasma chain is a separate blockchain that is anchored to the main
Ethereum chain, and uses fraud proofs to arbitrate disputes.
```

pros	cons
High throughput, low cost per transaction.	Does not support general computation.
Good for transactions between arbitrary users.	Need to periodically watch the network or delegate this responsibility to someone else to ensure the security of your funds.
	Relies on one or more operators to store data and serve it upon request.
	Withdrawals are delayed by several days to allow for challenges.

https://ethereum.org/en/developers/docs/scaling/plasma/

Optimistic Rollups

```
Optimistic rollups sit in parallel to the main Ethereum chain on layer 2.
They can offer improvements in scalability because they don't do any
computation by default. Instead, after a transaction, they propose the new
state to Mainnet or "notarise" the transaction.
```

pros	cons
Anything you can do on Ethereum layer 1, you can do with Optimistic rollups as it's EVM and Solidity compatible.	Long wait times for on-chain transaction due to potential fraud challenges.
All transaction data is stored on the layer 1 chain, meaning it's secure and decentralized. An operator can influence transaction ordering.	An operator can influence transaction ordering.

ZK Rollups

```
 ZK-rollups roll-up hundreds of transfers off-chain and generate a
cryptographic proof. The ZK-rollup smart contract maintains the state of
all transfers on layer 2, and this state can only be updated with a
validity proof.
```

pros	cons

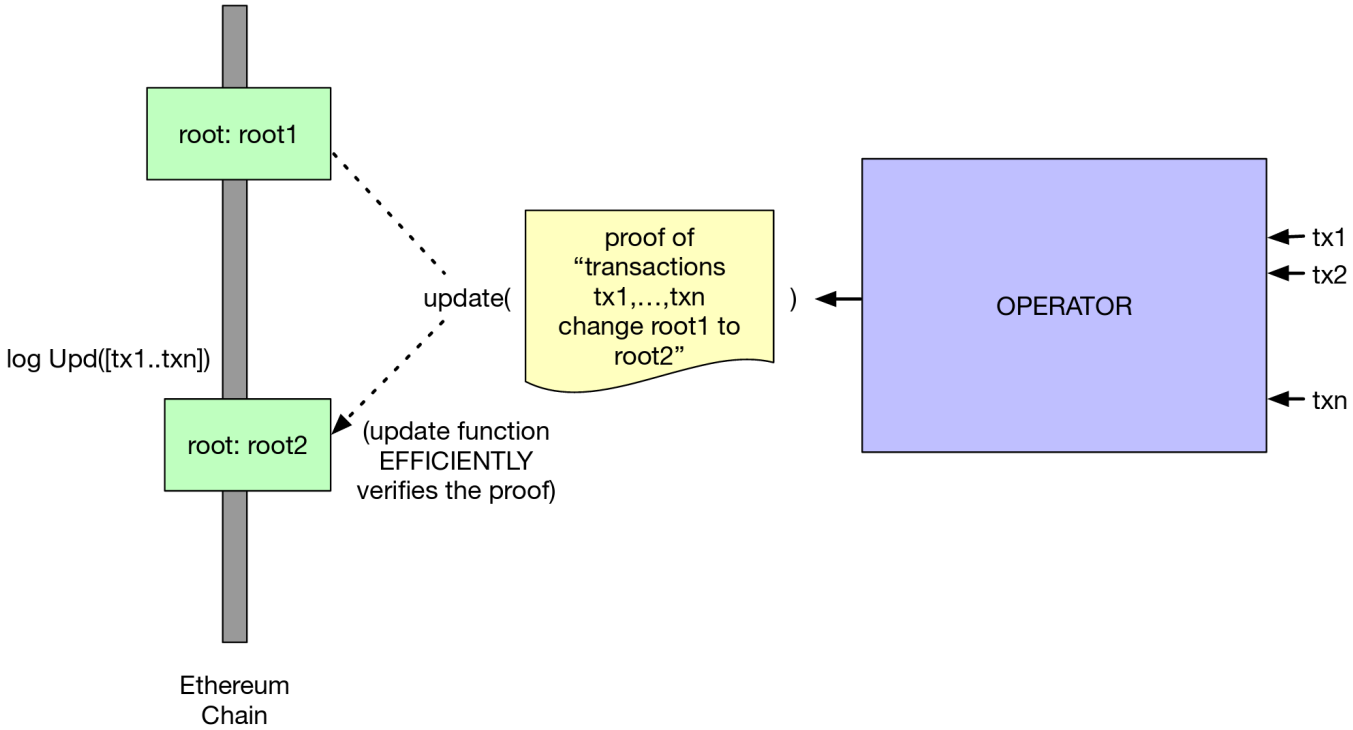
pros	cons
Faster finality time since the state is instantly verified once the proofs are sent to the main chain.	Some don't have EVM support.
Not vulnerable to the economic attacks that Optimistic rollups can be vulnerable to.	Validity proofs are intense to compute.
Secure and decentralized, since the data that is needed to recover the state is stored on the layer 1 chain.	An operator can influence transaction ordering.

https://ethereum.org/en/developers/docs/scaling/zk-rollups/

1. What was the biggest problem with the Plasma approach?

The biggest problem with the Plasma approach is "data availability attack"! Each Plasma chain requires an operator posting the Merkle root commitments to the mainchain. This requires us to rely on a third party to accurately post the Merkle root commitments on the chain. Unfortunately, operators can perform what's known as a "data availability attack" where they withhold posting certain transactions onto the mainchain for malicious reasons. In this case, operators can convince the network to accept invalid blocks, and there's no way to prove invalidity. Data availability attacks, unlike fraud, are not uniquely attributable. We have no way of knowing the attack is happening.

2. One of the solutions that has been gaining a lot of traction lately is zkRollups. With the use of a diagram explain the key features of zkRollups. Argue for or against this solution highlighting its benefits or shortcomings with respect to other solutions proposed or in use.



https://github.com/rollupnc/RollupNC

> In zkRollups you can generate succinct proofs for large computations, which are much faster and computationally easier to verify than they are to compute. This provides a way to "compress"

> expensive operations by computing them off-chain, and then only verifying the proof on-chain.

I think zkRollups is the most promising solution.
One of the reason is that it utilizes zero knowledge proof technology.
ZKP has large potential for solving key problems of blockchain.
For example, "recursive proofs" which is the act of verifying another proof inside a proof will be able to improve scalability and privacy.
Among current solutions zkRollups is the best solution which can fully utilize these ZK technology.

3. Ethereum is a state machine that moves forward with each new block. At any instance, it provides a complete state of Ethereum consisting of the data related to all accounts and smart contracts running on the EVM. The state of Ethereum modifies whenever a transaction is added to the block by changing the balances of accounts. Based on the massive adoption of Ethereum across the globe, this state has become a bottleneck for validators trying to sync with the network as well as validate transactions. Briefly describe the concept of stateless client, and how they help resolve this issue? Explain how Zero-Knowledge improves on the concept of stateless client?

The concept of stateless client is to enable nodes to validate a block without storing the Ethereum state. The key mechanism to enable the stateless client is a block witness which will validate a given state change against the previous state. Block witnesses will allow stateless nodes to store only the state root instead of the entire merkle patricia tree. This mechanism is similar to zkRollups in both utilizing ZK technologies for validating state.

Q2 Roll the TX up

3. ZKSync 2.0 was recently launched to testnet and has introduced ZKPorter. Argue for or against ZKPorter, highlighting the advantages or shortcomings in this new protocol.

ZKPorter increases scalability because it uses off-chain data availability.
The data availability of ZKPorter is secured by the so-called Guardians who participate in proof of stake with the ZKSync token. It looks less secure but still more secure than PoS in other systems because Guardians can not steal funds. They can only halt block production.
ZKPorter will be fully interoperable with ZKSync's zkRollups so it doesn't sacrifice user convenience and users can choose either option considering security and gas cost.

Q3 Recursive SNARK's

1. Why would someone use recursive SNARK's? What issues does it solve? Are there any security drawbacks?

With legacy blockchains like Bitcoin and Ethereum, when a new participant joins, they have to check every transaction since the beginning of the network to verify correctness. But this approach becomes increasingly expensive as the blockchain keeps growing. Instead of verifying the entire chain from the

beginning of time, with recursive SNARK's participants fully verify the network and transactions using recursive zero knowledge proofs. Nodes can then store the small proof, as opposed to the entire chain. So all participants can be full nodes and anyone can take part in consensus. Recursive SNARK's can also solve scalability problems which ZK-Rollups didn't solve in real and complicated use cases.

https://minaprotocol.com/ja/tech

2. What is Kimchi and how does it improve PLONK?

Kimchi is a collection of improvements, optimizations, and alterations made on top of PLONK. For example, it overcomes the trusted setup limitation of PLONK by using a bulletproof-style polynomial commitment inside of the protocol. This way, there is no need to trust that the participants of the trusted setup were honest (if they were not, they could break the protocol).

And there are some other implementation techniques for improving PLONK.

Kimchi adds 12 registers to the 3 registers PLONK already had.
More registers means that we now can have gates that take multiple inputs instead of just one.

Gate can directly write its output on the registers used by the next gate.
This is useful in gates like "poseidon", which need to be used several times in a row.

Lookup tables.
Kimchi builds the table and allows gates to simply perform a lookup into the table to fetch for the result of the operation.

https://minaprotocol.com/ko/blog/kimchi-the-latest-update-to-minas-proof-system

Q4 Final Project Ideas

ZK NFT ticket verification

It enables users to prove their NFT ownership without revealing their address. I think anonymity is more important in real events so NFT ticketing service needs ZKP.

pros	cons
Expand Web3 into real world use case.	Need centralized server.
Very little real use case for now.	

ZK Survey

Some events drop NFT for Proof Of Physical Attendance(POPA). Basic idea is the same above, this system can prove their POPA NFT ownership without revealing their address and send feedback anonymously. When collecting feedback it's better to be anonymous.

pros	cons
Expand Web3 into real world use case.	Need centralized server.

pros	cons
	Very little real use case for now.

ZK Random Chat

For example, recruiting 100 participants and randomly matching them, for a certain period they anonimously chat with mathcing partner. participants have to pay participation fee for a relayer server which assigns partner and relays transactions.

pros	cons
It looks interesting if participants are only some NFT holders.	Need trustful server.

I feel it's very diffucult to make ZK dApps trustless and full onchain😥

Question 5: Thinking in ZK

Question for Mina.

I'm very interested in the concept of "A Private Gateway between the Real World and Crypto".
When I tried to utilize ZKP for dApps, I found there are difficulties in interacting with offchain world.
I looks Mina solves this problem but I couldn't find how Mina actualy solves.
I think it's a not easy problem.
For example how to trustlessly interact with offchain world?
It seems that proofs are easy to transfered or copied.
If so, credit score example doesn't work.
Are there any technical documents about these Mina functions?

https://minaprotocol.com/ja/blog/building-a-private-gateway-between-the-real-world-and-crypto-three-use-cases