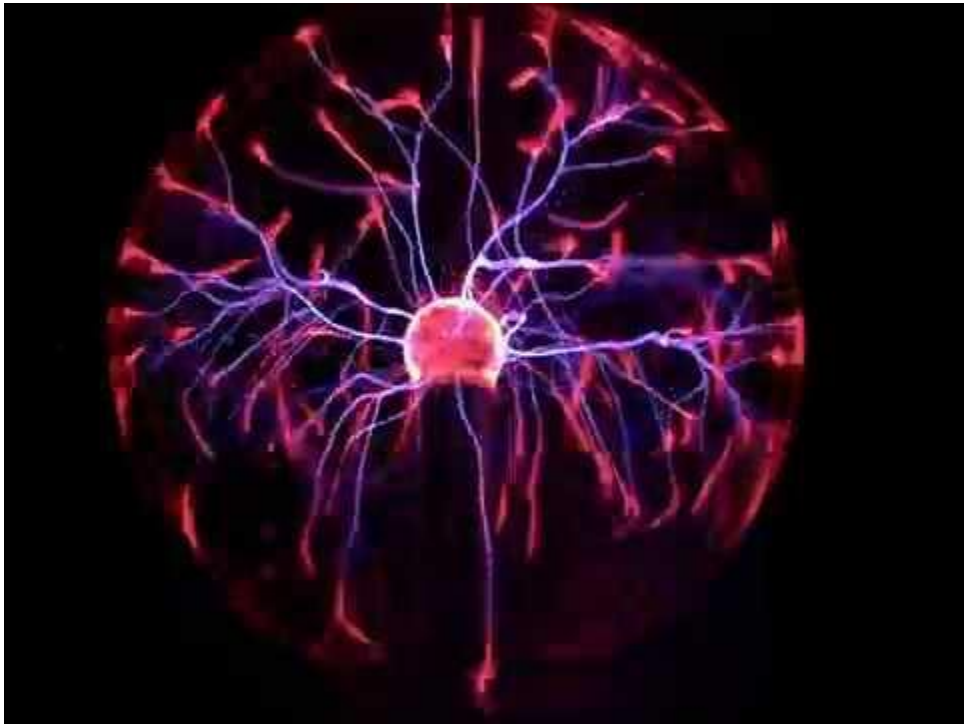
Plasma

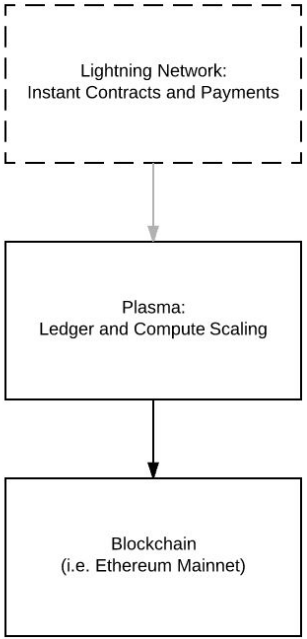
What is Plasma?



Seriously What is Plasma?

Plasma is an conceptually generalized version of of Lightning Network.

“…we seek to design a system whereby computation can occur off-blockchain but ultimately enforceable on-chain which is scalable to billions of computations per second with minimal on-chain updates.”



Review Lightning Network/Raiden Network

- Switches from a model where all transactions hit the shared ledger on the blockchain (which is the bottleneck) to a model where users can privately exchange messages which sign the transfer of value.

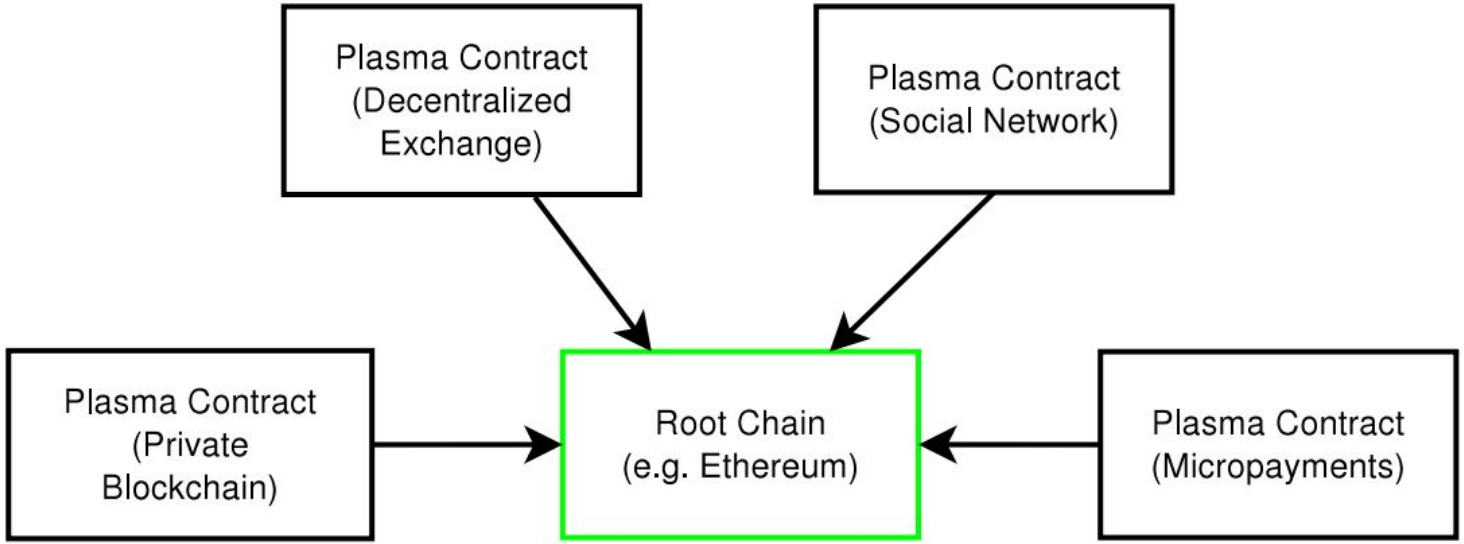
- Uses a network of p2p payment channels and deposits in Ethereum to preserve the guarantees expected from a blockchain system

- Transactional capacity is increased dramatically as channels are net-settled on the blockchain and transaction fees are reduced.

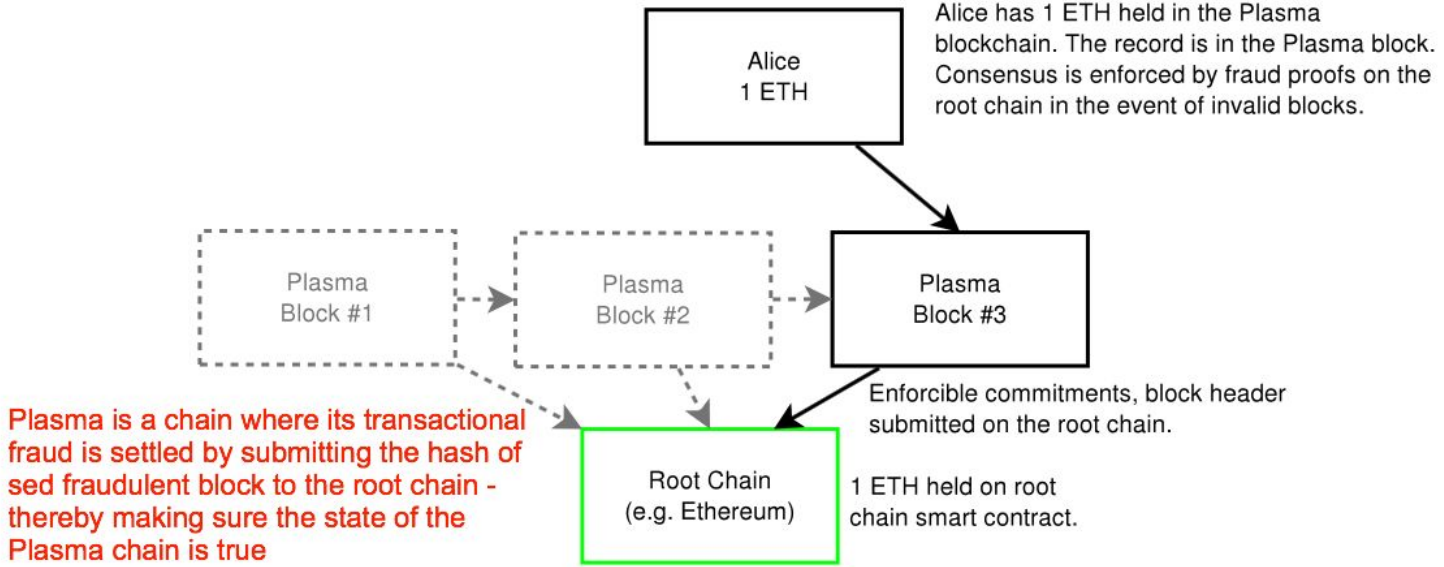
Design Goal

- One dominant blockchain and countless child chains
- Minimization of the trust in a usable way
- Child chain should be scalable
- Localized computations
- Fraud Proofs
- Every chain can be unique

Plasma Construction



Plasma Blockchain



Deposit

1. The coins/tokens (e.g. ETH or ERC-20 token) are sent into the Plasma contract on the root blockchain.
2. The Plasma blockchain includes an incoming transaction proof, not spendable yet.
3. Activating the transaction on the Plasma chain by signing the transaction.

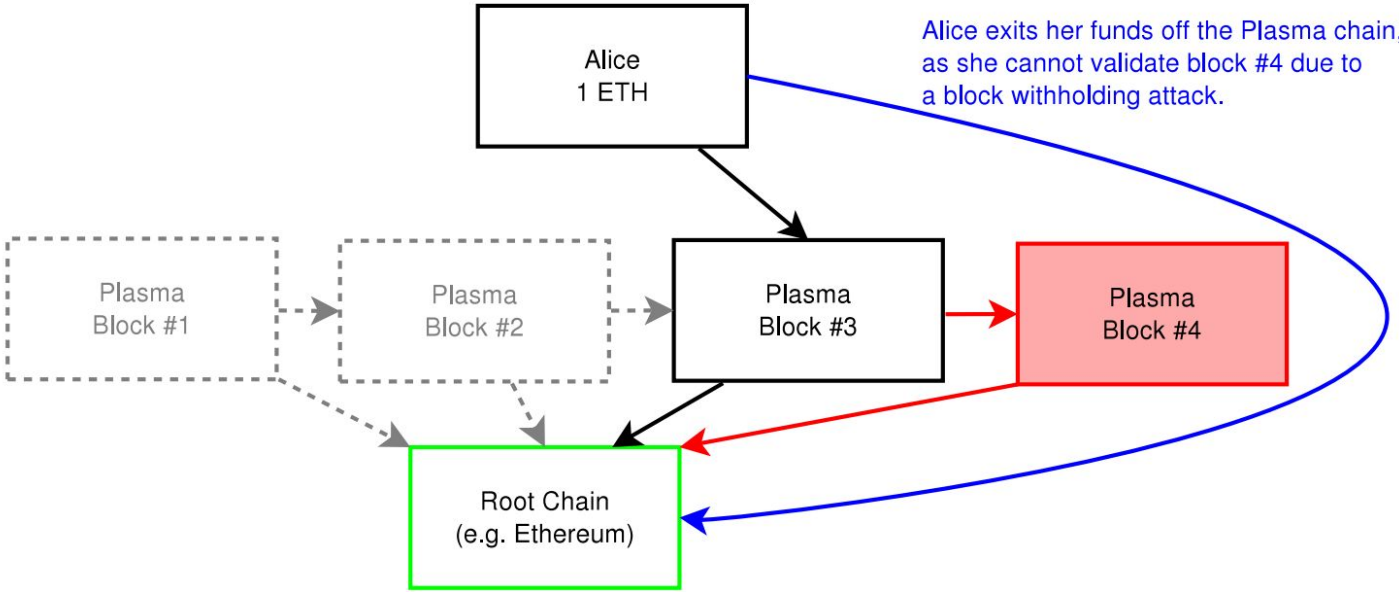
Transfer or State Transition

1. Alice wants to spend her output in the Plasma chain to Bob in the same Plasm chain (without the full transaction record being submitted on the blockchain). She creates a transaction which spends one of her outputs in the Plasma chain, signs it, and broadcasts the transaction.
2. The transaction is included in a block by validator(s) of the Plasma chain. The header is included as part of a block in the parent Plasma chain or root blockchain.
3. Alice and Bob observes the transaction and signs an acknowledgement that he has seen the transaction and block, which be included in another Plasma chain.

Withdrawal

1. A signed withdrawal transaction is submitted to the root blockchain or parent Plasma chain, additional bond is placed as part of the withdrawal to penalize false requests.
2. A predefined timeout period exists to allow for disputes. If the fraud proof of spent outputs is provided, then the bond is lost and the withdrawal is cancelled.
3. A second time delay exists to wait for timeouts of any other withdrawal requests with a LOWER block confirmation height.
4. Withdrawal will succeed automatically and redemption becomes effective once time period defined in the Plasma smart contract has elapsed.

Mass-exit

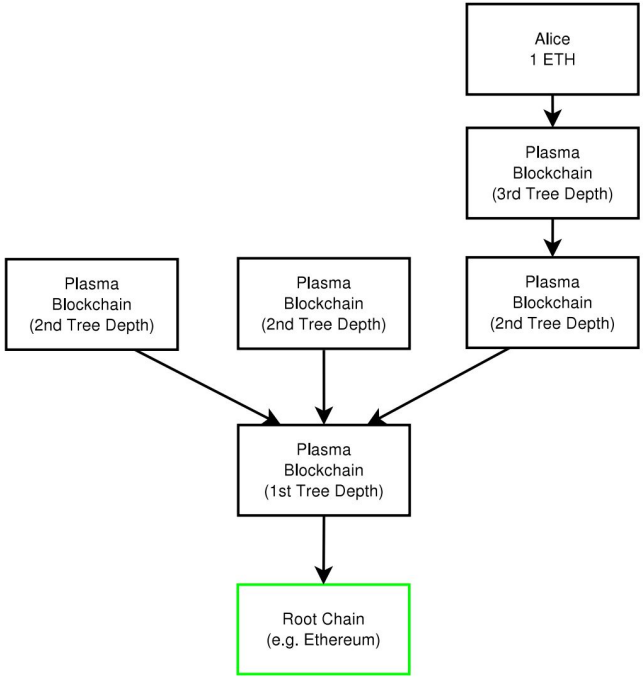

Alice
1 ETH
Alice exits her funds off the Plasma chain, as she cannot validate block #4 due to a block withholding attack.
Plasma Block #1
Plasma Block #2
Plasma Block #3
Plasma Block #4
Root Chain (e.g. Ethereum)

Mass Exit

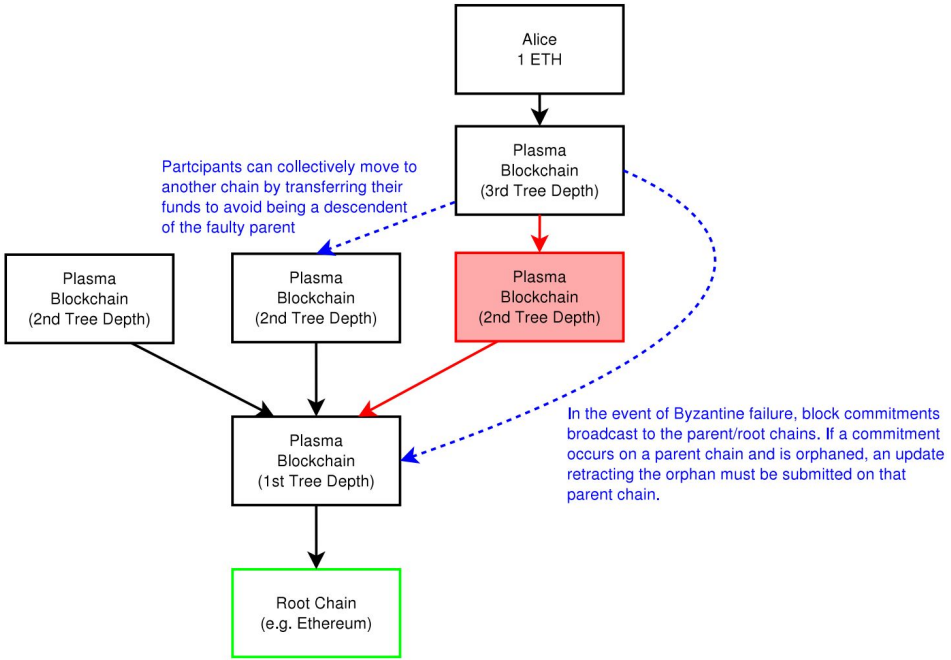
1. Alice coordinates with others on the Plasma chain to conduct a mass exit. Person making the duplicate withdrawal is penalized. Earlier transactions take higher precedence.
2. Pat the exit processor is willing to organize this exit.
3. Pat verifies the blockchain that all participants have the right to exit for up until the highest point of data availability
4. Users sign off on the mass withdrawal again after downloading all signatures.
5. If there are no challenges, then after the predefined finalization period for the MEIT proceeds and the users receive their funds.

Blockchains within Blockchains

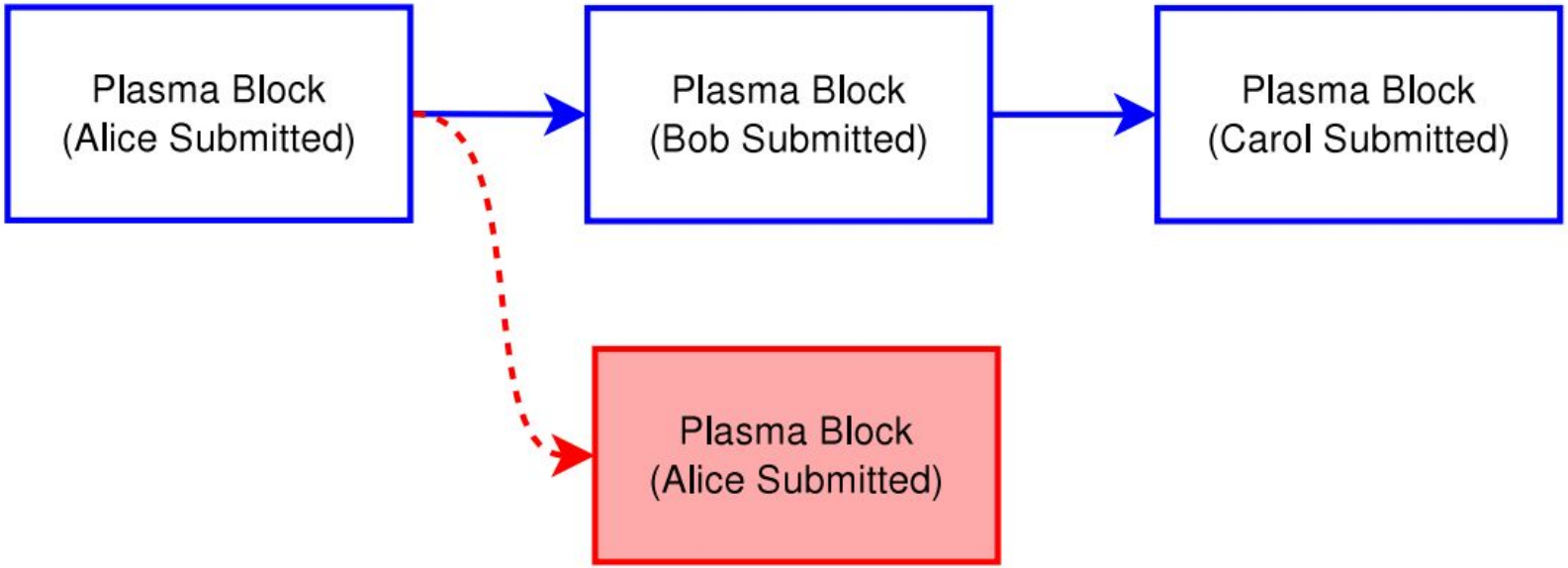
- Plasma composes blockchains in a tree.
- Intermittently updating to the parent chains (if needed).
- Each subchain can have it's own consensus.
- In case of Byzantine, it has the option of going to any of its parents to continue operation or exit with the current committed state.



Byzantine Handling



Plasma Proof-of-Stake



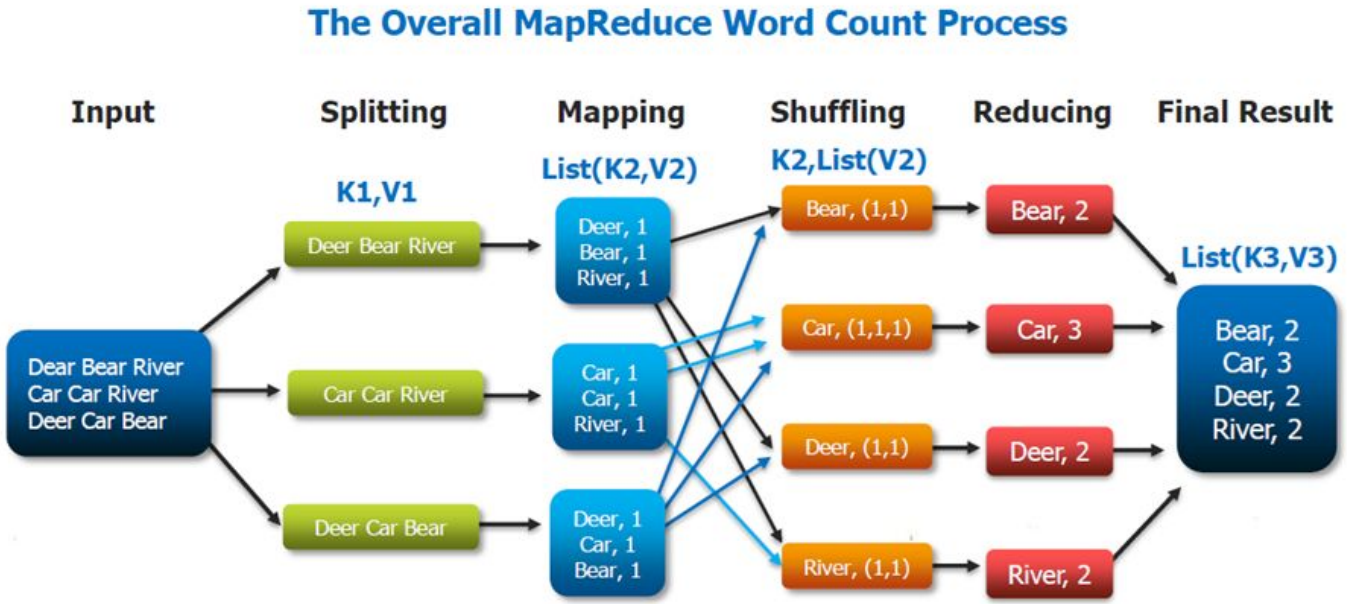
Plasma Proof-of-Stake

- Creates incentives for validators to represent the past 100 blocks to match the current staker ratio by rewarding more transaction fees to be paid out to accurate representation.

- The correct chain tip is the chain with summed weight of the highest fees.

- In Byzantine env, non-Byzantine participants conduct a mass compact withdrawal on the parent/root blockchain

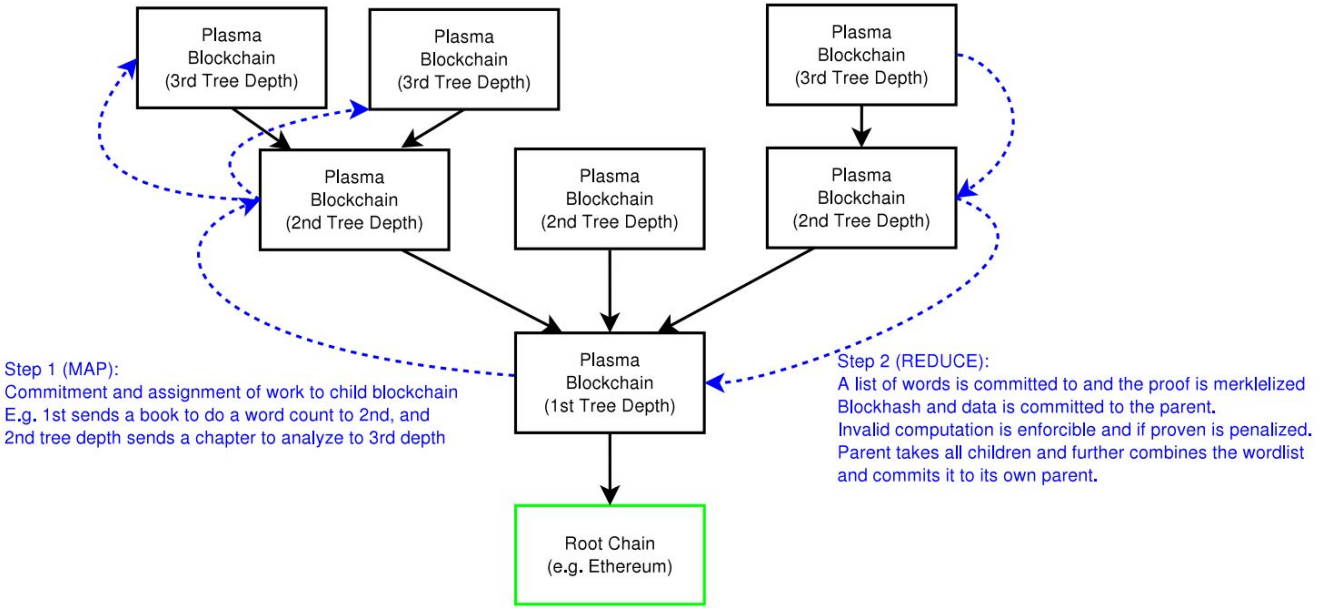
Blockchains as MapReduce



Blockchains as MapReduce

blockchain : git :: Plasma : Hadoop (MapReduce)

Blockchains as MapReduce



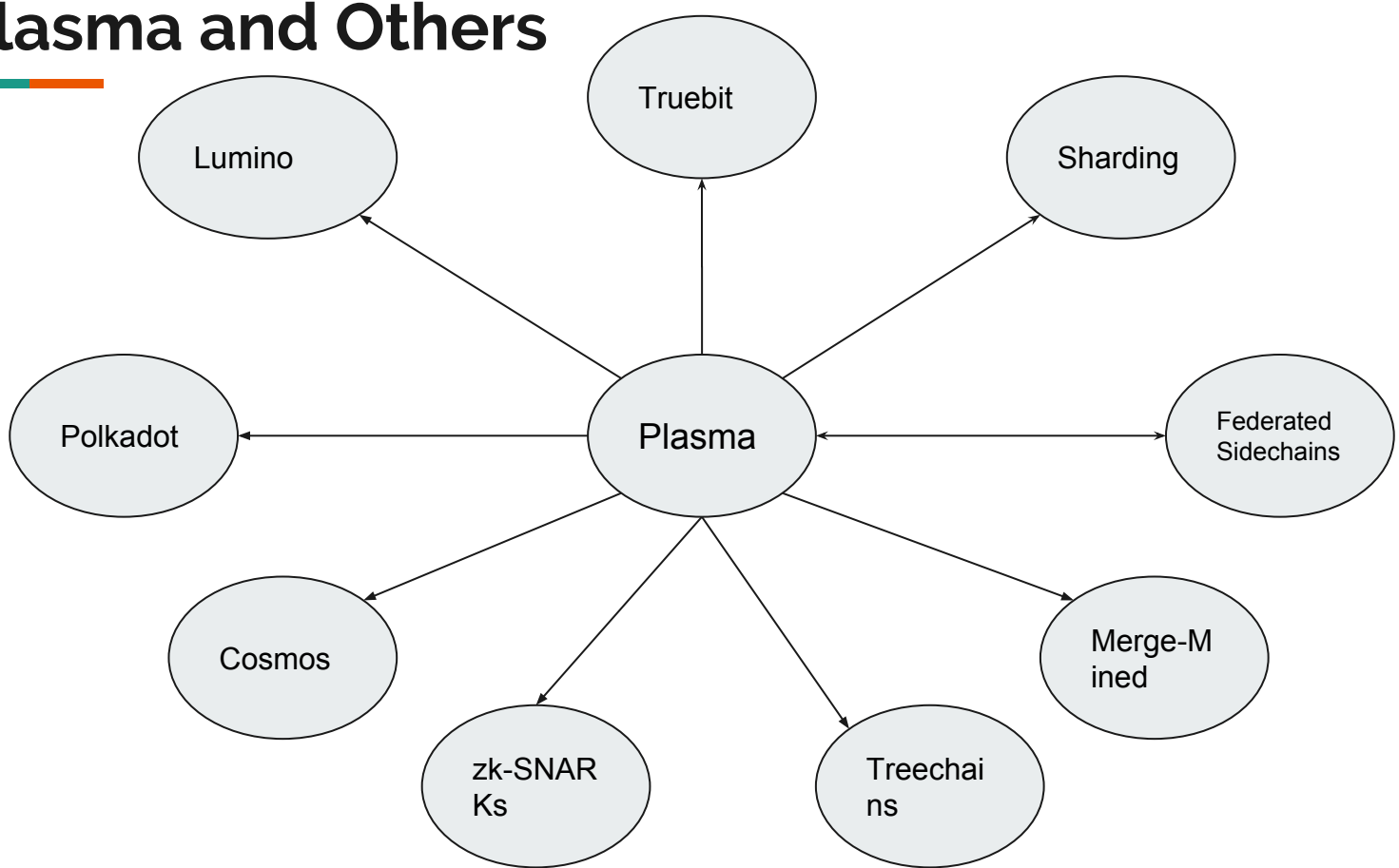
Blockchains as MapReduce

1. The map phase publishes the data commitments.
2. Reduce phase includes a merkleized proof of state transition when returning the result.

Economic Incentives

- Every Plasma chain is represented by a set of contracts. Fraudulent incurs significant penalization.
- Plasma chain requires the token to secure the network in a Proof-of-Stake structure, Byzantine behavior would cause a loss in value of the token.
- Stakeholders have incentives to continue operating the network as they receive transaction fees for operating the network.
- TX fees paid out to the network stakers creates long-term value for the token.
- Stakers will persistently run the chain and are bound by the fraud proofs defined in the contracts in the root blockchain.

Plasma and Others



Examples of Plasma protocol

- Cosmos / Polkadot
- OmiseGo — Decentralized exchange, currency agnostic

Issues

- Mass exits in child plasma chain might lead to congestion on the main network and lead to delays crossing the challenge period.
- If no one is monitoring a plasma chain, there is no scope for submitting fraud proofs.
- Closing of transactions on main chain even before they are closed on the plasma chains.
- Root chain is under 51% attack that will affect complete plasma chain ecosystem.
- If parent chain has issue and stops producing blocks, child chains are responsible for fixing the parent chain.
- Once consensus is established, changing it will be very tough. You will have to exit plasma chain and move function to another plasma chain.