

Week 5 EPFsg Ethereum Roadmap Notes

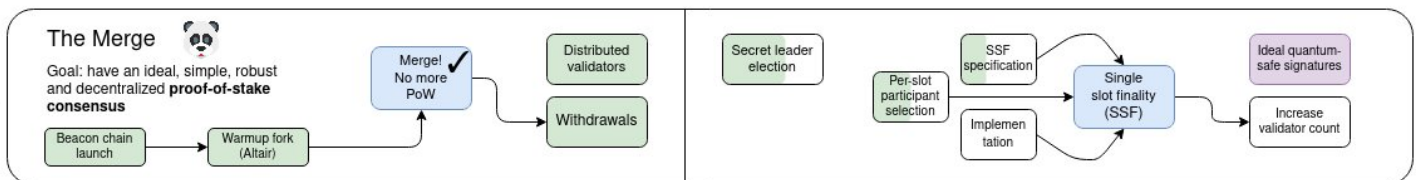
Guest speaker

- [Domothy](#) - Ethereum Foundation Research

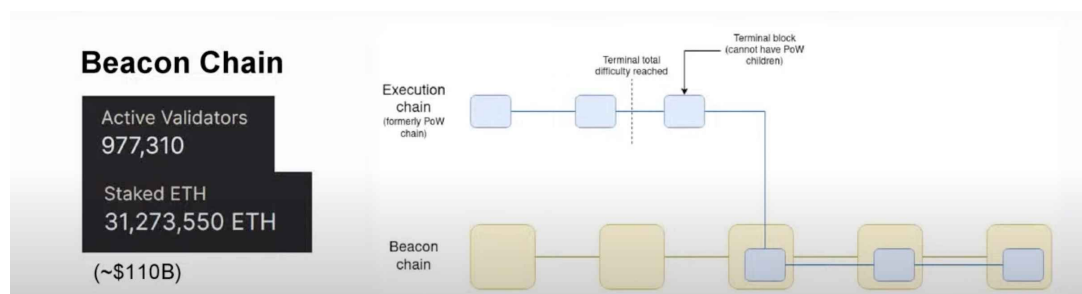
Summary notes

- Edited by [Chloe Zhu](#)
- Online version: <https://ab9jvcjkej.feishu.cn/docx/K1IGdU7Hzogqg3x0woQcBgC9nEh>

Merge: Better PoS



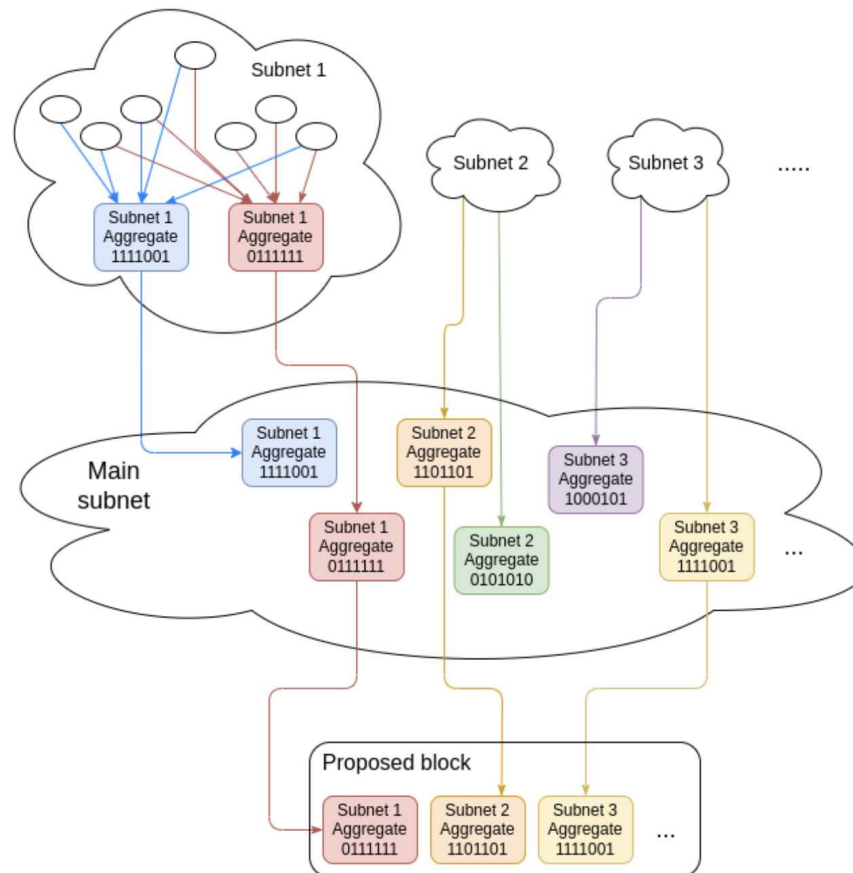
- **Beacon chain launch & Merged**
 - Currently it has almost c.1m validators, with over 31m ETH staked (c.\$110bn)



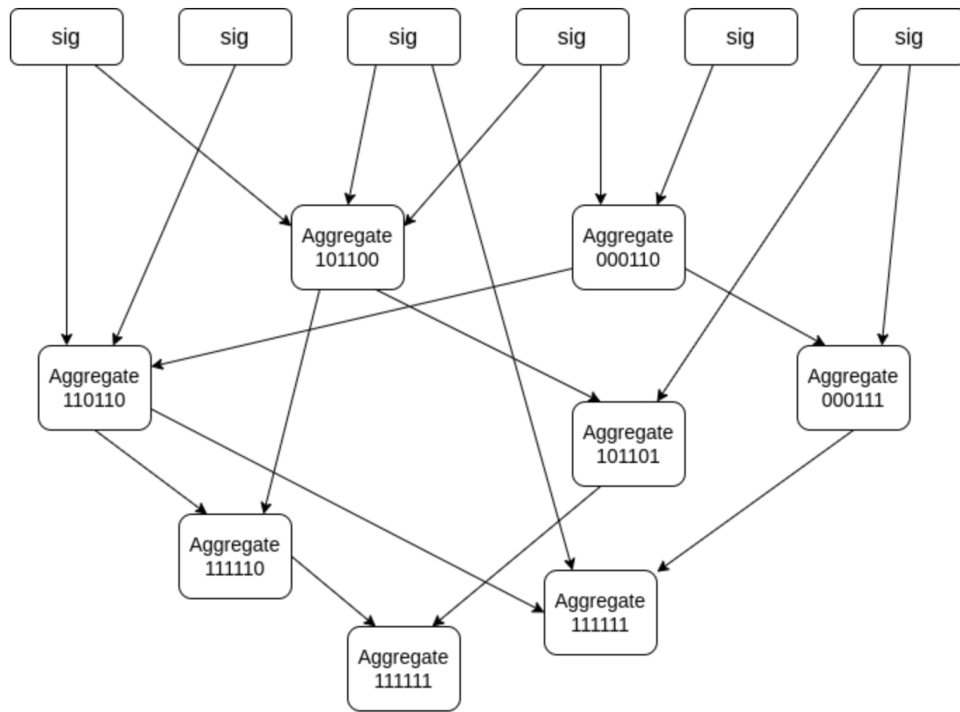
- **Warmup fork (Altair): Sync committee/ Light client protocol**
 - Sync committee
 - Instead of having every validator to verify every slot in each epoch, the Altair fork introduced the sync committee
 - Each committee has 512 validators, which rotated every 256 epochs (c.27 hours)
 - Altair link: <https://github.com/ethereum/annotated-spec/blob/master/altair/sync-protocol.md#introduction>
 - Light client protocol

- The purpose of the sync committee is to allow light clients to keep track of the chain of the beacon block headers
- Key feature
 - Light-weight: 512 signatures to check VS c.1m validators to check previously
 - Trust-minimized rather than trustless
- Further link on light clients : <https://a16zcrypto.com/posts/article/an-introduction-to-light-clients/>
- **Secret Leader Election (SLE)**
 - Current problem
 - Leader/ proposer (i.e. the validator that is in charge of proposing a block at each slot) is revealed a bit ahead of time. Thus, it is exposed to DoS attacks in theory.
 - SLE solution
 - [EIP 7441](#) Upgrade block proposer election to Whisk
 - Upgrade the block proposer election mechanism to Whisk, a single secret leader election protocol
 - Allow elected block proposers to remain private until block publishing, to prevent DoS attacks
 - Currently, SLE is relatively in low priority. But priority can change if such DoS attacks happen.
- **Single Slot Finality (SSF)**
 - Current problem
 - The current finality time is after 2 epochs (c.12.6min) as too many signatures to check and aggregate.
 - And the devs want to enhance the finality speed into 1 slot (12s)
 - Solution path
 - Fewer validators through Max EB ([EIP 7251](#))
 - Fewer active validators eg. rotating cap
 - Way fewer validators (8,192) + Distributed validators tech (DVT)
 - Better signature aggregation schemes
 - Vitalik's blog on Paths towards SSF: https://notes.ethereum.org/@vbuterin/single_slot_finality
- **Ideal quantum-safe signature:** Quantum proof beacon chain

- Current problem
 - Ethereum beacon chain currently relies on BLS aggregation to aggregate signatures into a single combined aggregate.
 - However, the current approach is vulnerable to quantum computers, and it's not SNARK-friendly.

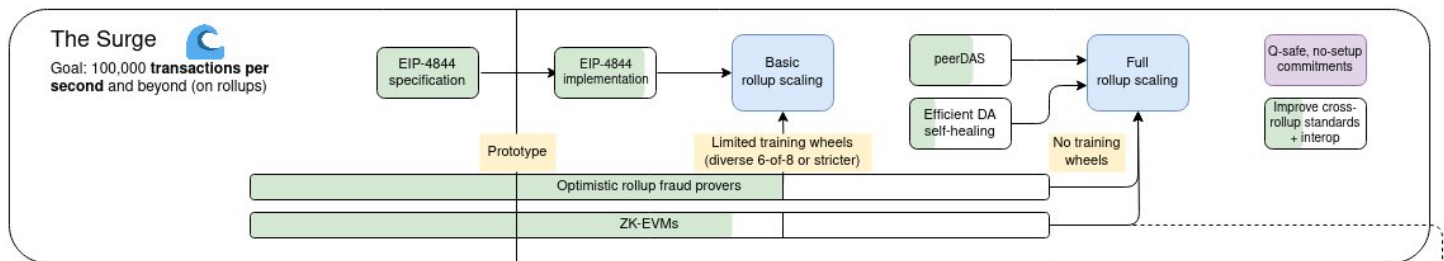


- A better approach
 - A recursive route, where aggregation happens in multiple layers. This allows the network to be highly unstructured and quantum-proof.



- Vitalik's blog on STARK signature aggregation:
https://hackmd.io/@vbuterin/stark_aggregation





















Surge: More data availability for rollups



• Basic rollup scaling

- Scaling ethereum
 - Safely scaling L1 execution is hard, but scaling L1 data is easier
 - What rollups do is to covert L1 data into L2 execution
- Rollup-centric roadmap
 - Optimistic rollup
 - Assume all txns are valid
 - Slash sequencer if not through fraud proofs
 - Zero-knowledge rollup
 - Sequencer proves txns are valid
 - Succint proofs verified by L1

- All rollups' data must be available on L1
- All rollups should be able to force L2 txn inclusion (i.e. To exit back to L1)
- Further links
 - Vitalik's blog on An Incomplete Guide to Rollups: <https://vitalik.eth.limo/general/2021/01/05/rollup.html>
 - Vitalik's post on A rollup-centric ethereum roadmap: <https://ethereum-magicians.org/t/a-rollup-centric-ethereum-roadmap/4698>
- **Limited training wheels on Rollups**
 - Upgradability/ mutability
 - Multisig/ limited-governance
 - Permissioned elements
 - End game for rollups: To be as trustless as L1, but as of today they are still not

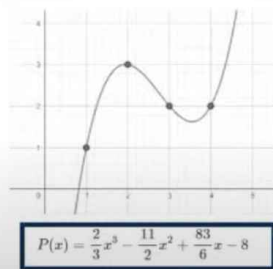
#	NAME	RISKS ⓘ	TYPE ⓘ	STAGE ⓘ	PURPOSE ⓘ
1	 Arbitrum One ⓘ		Optimistic Rollup ⓘ	STAGE 1	Universal
2	 OP Mainnet ⓘ		Optimistic Rollup ⓘ	STAGE 0	Universal
3	 Blast ⓘ		Optimistic Rollup ⓘ	STAGE 0 ⓘ	Universal, DeFi
4	 Base ⓘ		Optimistic Rollup ⓘ	STAGE 0	Universal
5	 Starknet		ZK Rollup ⓘ	STAGE 0	Universal
6	 zkSync Era ⓘ		ZK Rollup ⓘ	STAGE 0	Universal
7	 Linea ⓘ		ZK Rollup	STAGE 0 ⓘ	Universal
8	 dYdX v3 ⓘ		ZK Rollup ⓘ	STAGE 1	Exchange
9	 Mode Network ⓘ		Optimistic Rollup ⓘ	STAGE 0	Universal
10	 Polygon zkEVM ⓘ		ZK Rollup ⓘ	STAGE 0	Universal

l2beat.com

- **Data Availability Sampling (DAS)**
 - Ultimate question: To prove that the data is available
 - Approach

- One approach is to download all the data to prove it's available. However, this doesn't scale well.
- Another approach is to take the data and make it a polynomial equation extended by evaluating that equation at multiple points. Then use a polynomial commitment scheme to conduct random sampling.
 - For a polynomial equation, 50% of the data & extension can recover 100% of the data.
 - Through polynomial commitment schemes, it's possible to verify the data availability through a few sample checks without get the whole burden of downloading all the data.

Polynomial Commitment Schemes



$$P(x) = \frac{2}{3}x^3 - \frac{11}{2}x^2 + \frac{83}{6}x - 8$$

Data: (1, 3, 2, 2)
Extension: (7, 21, 48, 92)

In practice, $P(x)$ has thousands of coefficients

$C = \text{commit}(P) =$ a few bytes (*like a hash*) known to all nodes

- Ask for random data point (e.g. the 3rd one)
- Receive the value 2 along with proof π
- Verify proof π against C , is satisfied that $P(3) = 2$
- At most 50% odds of "being fooled"
- Ask for another random data point, odds become 25%
- Another sample: 12.5%
- 30 samples = $1 / 2^{30} \approx 1$ in a billion chance of being fooled

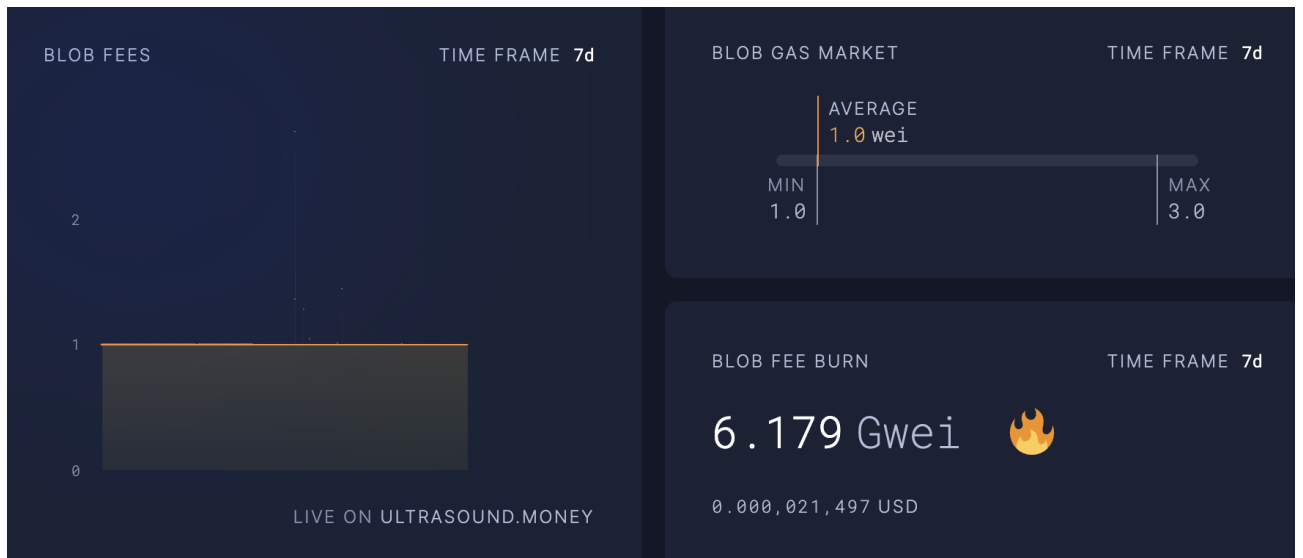
- **EIP 4844 introduces blobspace**

- DAS

- No fancy sampling yet, so every node needs to download all the blobs
 - But EIP 4844 sets the stage for DAS using KZG commitment scheme

- Conservative initial values

- Target of 3 blobs/ block, with max 6 blobs/ block
 - Pricing of blobs is similar to EIP 1559 and the base fee is burned: If the block has 3+ blobs, the price will increase, and vice versa



<https://ultrasound.money/#blobs> (as of Mar 22nd, 2024)

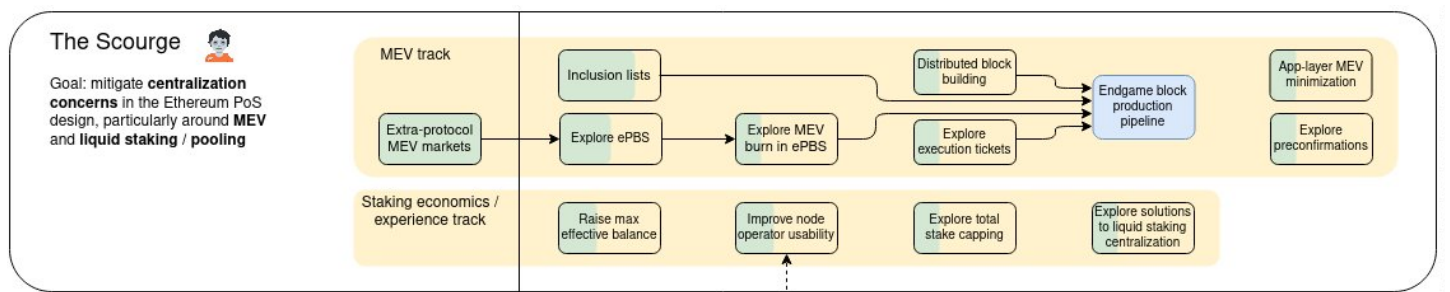
- **Quantum-proof blobspace**

- Current problem
 - KZG drawbacks: Not quantum-proof and required a trusted setup (>140k contributors)
- Endgame solution
 - Hot-swap KZG for something based on STARKs or Lattices

- **Cross-rollup interoperability**

- Current problem: Liquidity fragmentation between rollups
- Solution
 - Establish standards between rollups
 - Based rollups, preconfirmations, shared sequencing

Scourge: Less MEV downsides



- MEV track

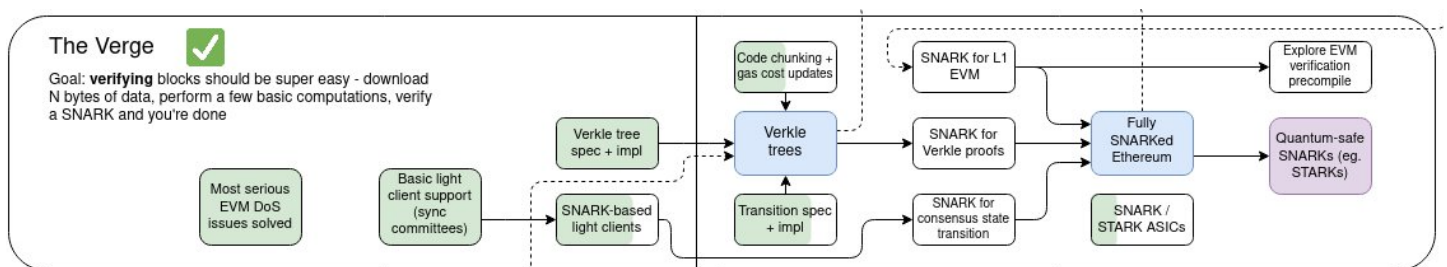
- **Proposer/ Builder Separation (PBS)**

- Current problem: MEV is inevitable, and untamed MEV markets will hurt solo stakers
- Goal: Minimize the choices validators have to make and reduce incentive to specialized validators

- Solution
 - Current solution: Out-of-protocol with MEV boost, where relays act as trusted brokers
 - Future solution: **Enshrined PBS (ePBS)**, which removes relays and allows MEV burning to smooth the staking yield
 - Future solution: **Inclusion list**, which puts constraints on builders and reduces censorship by forcing transactions inclusion
- Endgame block production
 - Centralized block production
 - Decentralized validation
 - Strong anti-censorship protection
- Vitalik's blog on Endgame: <https://vitalik.eth.limo/general/2021/12/06/endgame.html>
- **Execution tickets**
 - Solution to deal with MEV and distorted yield to solo stakers
 - Sell the right to propose a block ahead of time, like a lottery ticket
 - Even more role separation e.g between attesting & proposing
 - Key features
 - Attesters remain simple, while proposers can specialize (constrained by inclusion lists)
 - Permissionless degen MEV lottery (cost of tickets \approx expected value of MEV per block)
 - EthResearch on Execution tickets: <https://ethresear.ch/t/execution-tickets/17944>
- **App-layer MEV minimization**
 - Develop better Dapps with MEV in mind
 - A few examples: <https://www.mev.wiki/solutions/mev-minimization>
- **Preconfirmations**
 - Receive next-block inclusion guarantee from builder
 - Pair well with execution tickets and restaking schemes
- Staking economics
 - **Raise max effective balance (MaxEB)**
 - Current EB: Min 32 ETH, Max 32 ETH
 - After MaxEB: Min 32 ETH, Max 2048 ETH

- MaxEB can enable rewards automatic compounding, and fewer validators for the same amount of stake
- Lower overhead of validators could reduce the number of P2P messages over the network and become a pathway toward Single Slot Finality
- **Explore total stake capping**
 - Related to overhead/ SSF
 - Research in progress:
 - Changing issuance curve (possibly into negative), Stake targeting
 - EthResearch on Endgame Staking Economics: A Case for Targeting: <https://ethresear.ch/t/endgame-staking-economics-a-case-for-targeting/18751>
- **Liquid staking centralization**
 - Research in progress: Enshrine? Cap slashing penalties?

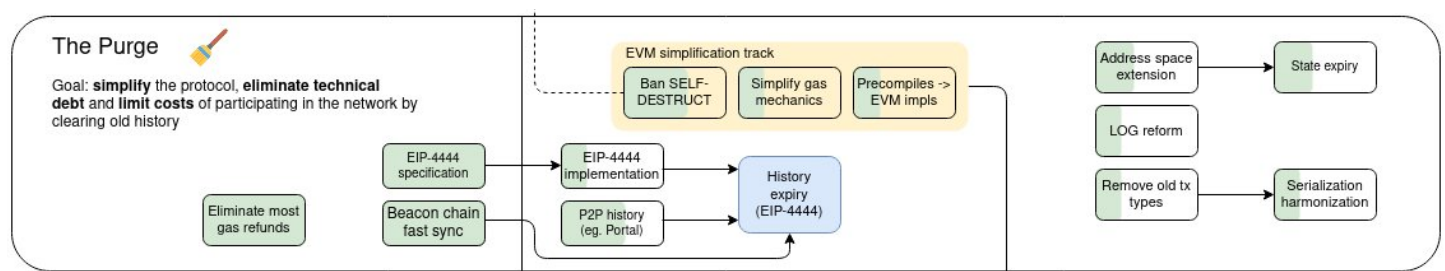
Verge: Easier verification



- **Verkle trees**
 - State vs History
 - State: All current balances
 - History: All past transfers/ txns
 - Current method:
 - Merkle-prove: Receive new nodes, Compute the intermediary nodes, and Check if state root matches with block header
 - Nodes need to sync the history, compute the state, then check balances and validate new txns. However, the Merkle proofs could become much larger and unmanageable as the state size grows.
 - Future method:
 - Verkle-prove: Every node is a polynomial commitment over its children. Siblings are no longer necessary as only paths, intermediary nodes and open proofs are needed for the proof.

- Features of Verkle trees
 - Much shorter state proofs
 - Wider tree: 256 siblings vs 16 in merkle tree
 - ZK-friendly proofs
 - Allow stateless validators: No history needed, instant sync
 - Light clients become even lighter
 - Lower dev reliance on centralized indexers
- More info on Verkle: [Verkle.info](https://verkle.info)
- **Fully SNARKed Ethereum**
 - Snarkify light client protocol (sync committee transitions)
 - Snarkify all beacon chain transitions (signatures, balance changes, etc.)
 - Snarkify verkle state across proofs/ block witness
 - Eventually snarkify all EVM execution: zkRollups are working on zkEVMs, that could be brought back to the core protocol in the future.
- **zkEVM opcode/ precompile**
 - Verify EVM execution proof inside the EVM (or inside an EVM execution proof)

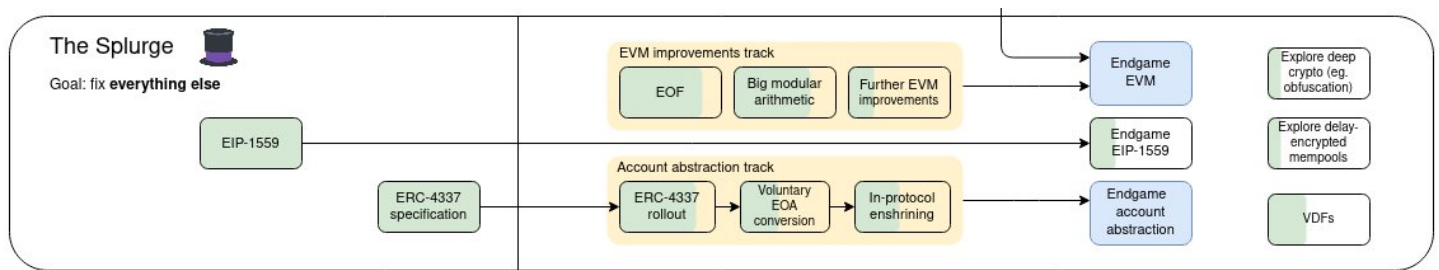
Purge: Simpler protocol



- **History expiry (EIP 4444):** Autoprune history older than 1 year
 - Simplifies client codebases: No need to support earlier forks
 - Alleviate node storage requirements
 - History must reliably be accessible by other means eg. Portal network, torrents, block explorers, etc.
- **State expiry**
 - Lower priority now compared to PBS and Statelness
 - Requite many breaking changes eg. Address length
- **Various harmonizations**

- Serialization: RLP for EL and SSZ for CL
- Slowly phase out old tx types eg. pre-EIP 1559 legacy type

Splurge: Miscellaneous goodies



- **EVM improvements/ EVM Object Format (EOF)**

- Series of EIPs to restructure aspects of EVM, making future upgrades easier
- Notes of EOF overview: <https://notes.ethereum.org/@ipsilon/evm-object-format-overview>

- **Account Abstraction**

- UX around externally owned accounts (EOAs) is pretty bad
 - Features/ function that requires further work: Gas sponsorship, tx batching, key security, spending conditions, social recovery
- [EIP 3074](#) to delegate control of EOAs to smart contract
- [ERC 4337](#) for smart wallet standards across EVM chains/ rollups (potential eventual enshrinement)

- **Endgame EIP 1559**

- More like an AMM curve
 - Track excess gas instead of previous block's gas usage
 - Higher censorship cost: Target the entire fee vs priority fee currently
- Multidimensional EIP 1559
 - Similar to gas/ blob today, but for more resources eg. Call data, state reads/ writes, block size, witness etc.
 - More efficient pricing: Demand for one resource won't affect price for other resources
 - Time-aware base fee calculation ([EIP 4396](#)): Avoid treating missed slots as sudden spike in demand

- **Deep crypto**

- Fully homomorphic encryption
- One-shot signatures:

- Related paper: <https://eprint.iacr.org/2020/107>

- **Encrypted mempools**

- Toxic MEV will disappear completely under encrypted mempools

- **Verifiable delay functions (VDF)**

- Non-parallelizable proof of work
 - Enhance beacon chain randomness