Xored.io

On-chain Privacy-preserving Banking

Literature Review v1.0

October 2025

Overview

- Basic concepts
  - Security terms in privacy-preserving banking context
  - Auditability Models
  - Privacy models
  - Common Privacy-preserving approaches
- SNARKs Variations in practice
- Products overview

Literature Review

Basic concepts

Security Terms in Context (1)

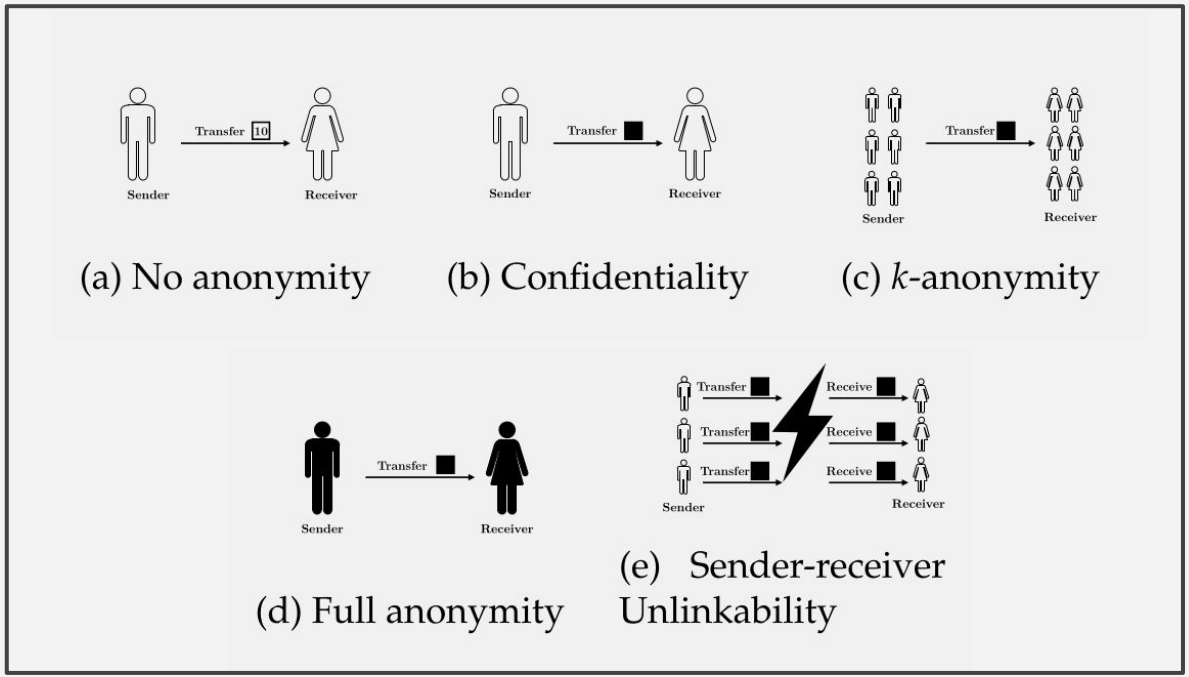
1. Confidentiality: Amount-privacy
2. Anonymity: Sender / Receiver / Amount privacy
3. Unlinkability: Prevents an adversary from linking which sender sent funds to which receiver, even if individual identities are known.

Security Terms in Context (2)

4. Accountability:
   A system is accountable if it can enforce rules or trace behavior (e.g., transaction limits, KYC, audit triggers) even under privacy constraints.

5. Auditability:
   This property ensures that an external auditor, with access to the public ledger, can provably obtain specific information required for auditing (e.g., the participants in a transaction).

Privacy Models



Reference : SoK: Privacy-Preserving Transactions in Blockchains (2025)

Auditability Models (View Keys)

1. Viewing Keys

    - A cryptographic key that allows a trusted third party (auditor) to selectively view specific transaction details (e.g., sender, receiver, or amount).

    - Properties:
        - Interactive auditability: requires the user to share the key.
        - Selective disclosure: only a subset of information is revealed.

Auditability Models (Zk-Knowledge)

2. Zero-Knowledge (ZK) Policy Compliance Proofs

    - Users prove in zero knowledge that a transaction satisfies certain rules or regulations without revealing anything else.

    - Properties:
        - Can enforce spending policies or traceability triggers.
        - non-interactive auditability.

Common Privacy-preserving approaches (1)

1. **Stealth addresses**
   a. First introduced at 2014
   b. Aztec and Monero uses it.
   c. View key to link, spent key to spend

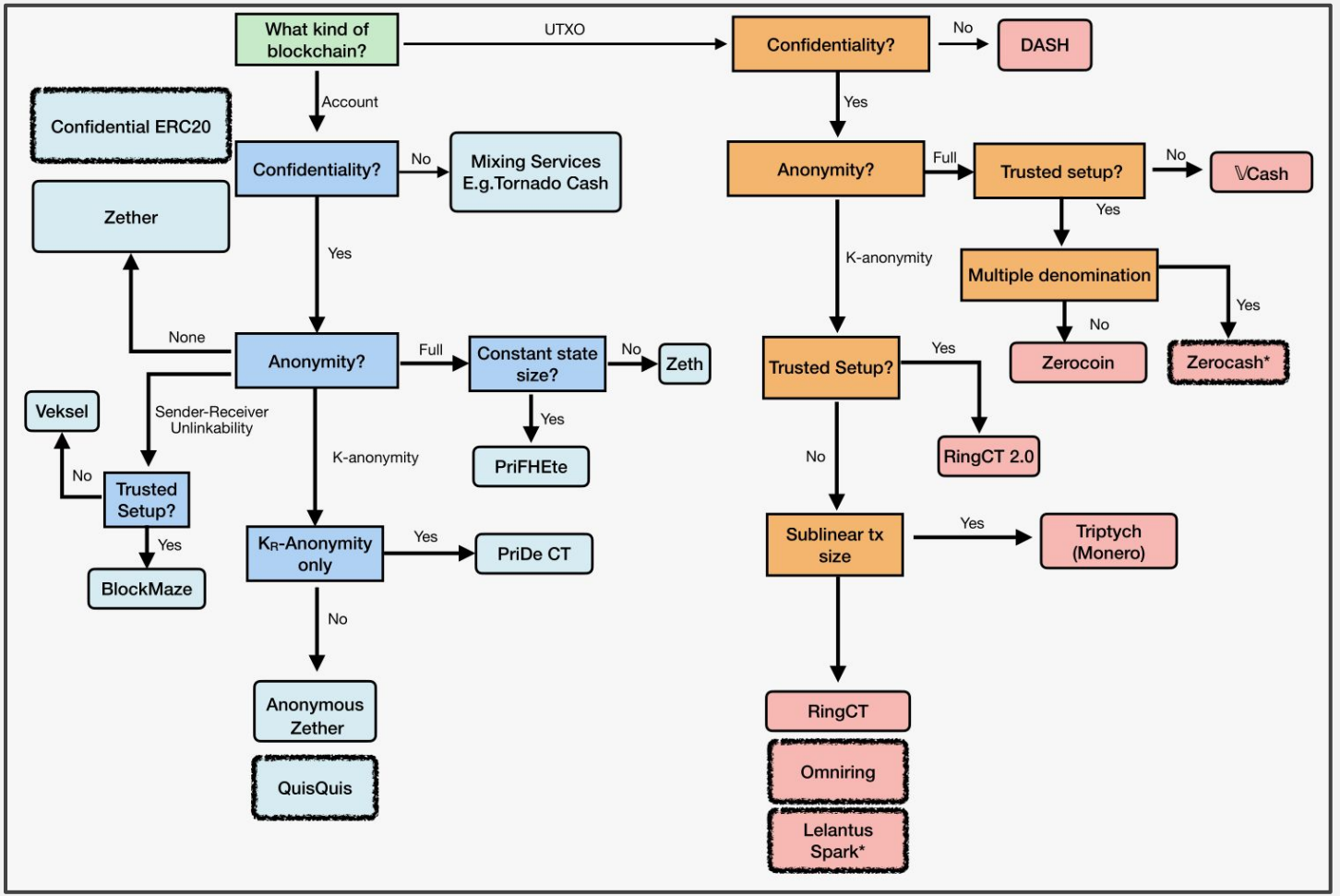
2. **Account-based** systems
   a. **[**Account → Enc(Balance)**]** mapping
   b. Common technique: Homomorphism

Common Privacy-preserving approaches (2)

3. **UTXO-based** systems
    a. UTxO: Pool of unspent transactions
    b. Privacy-preserving UTxO: pool of coin commitments
    c. Common Techniques
        i. Mint & Pour technique (commitments/ serialNumbers)
        ii. Ring signature

4. **Mixing** Techniques
    a. peer-to-peer (P2P) mixing
    b. Centralized tumblers
    c. smart contract-based mixers

SoK categorization



Reference : SoK: Privacy-Preserving Transactions in Blockchains (2025)

Literature Review

SNARKs

Popular SNARK systems

Name	Setup type	Recursion	Prove Time	Proof size	Verifier time
Groth16 (2016)	per circuit	No	O(**n**)	O(1)	O(1)
Marlin (2019)	Universal	Yes	O(**n** log(**n**))	O(**n** log(**n**))	O(log(**n**))
PLONK (2019)	Universal	Limited	O(**n** log(**n**))	O(**n** log(**n**))	O(log(**n**))
Halo2 (2021)	Universal	Yes	O(**n** log(**n**))	O(log(**n**))	O(1)

Note: None of these systems are quantum secure. They rely on assumptions such as ECC or ECC pairing that are **NOT** Quantum-secure.

Note: None of these systems need trusted setup.

Quantum-secure SNARK systems

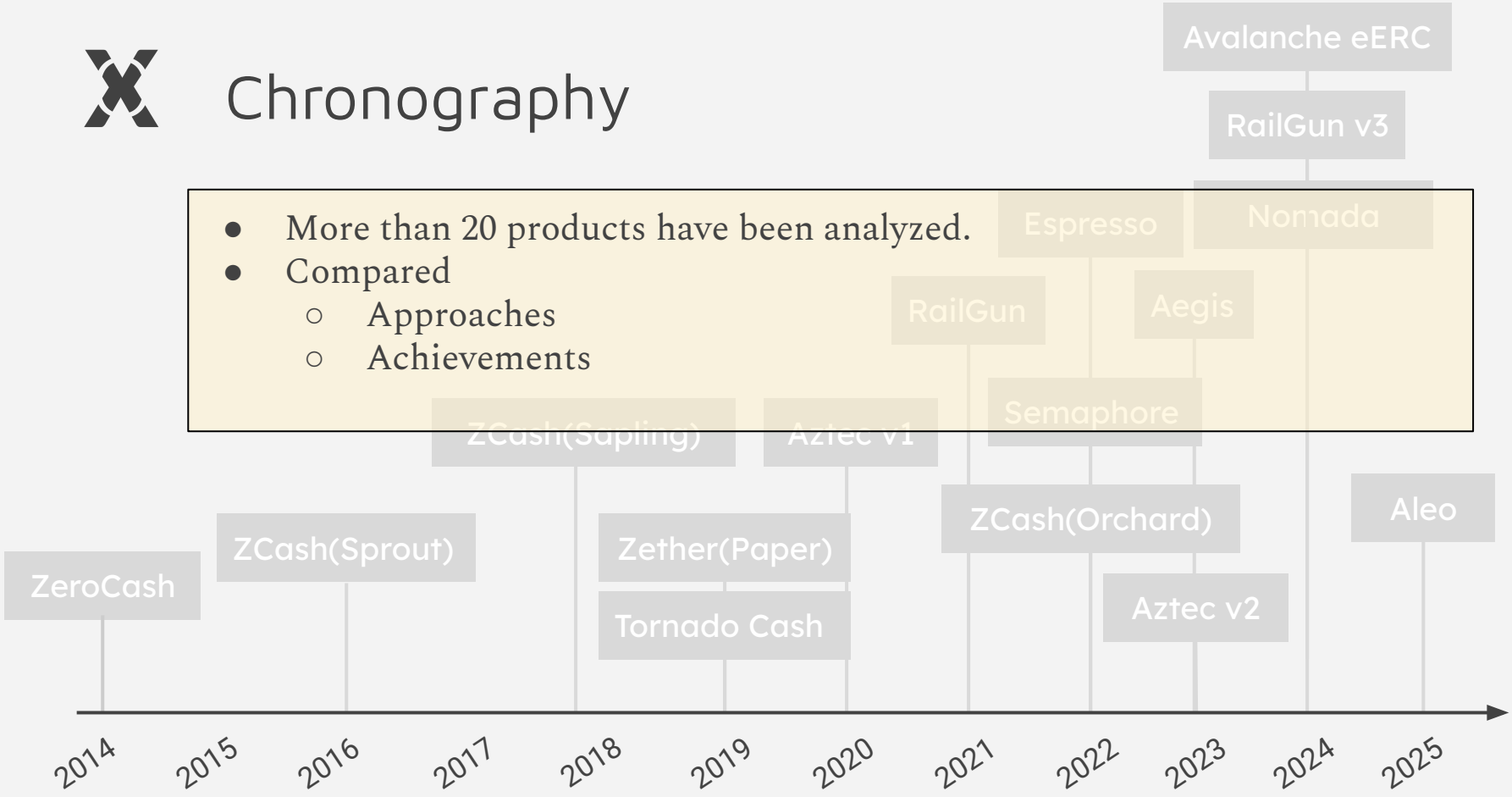
Name	Technique	Recursion	Prove Time	Proof size	Verifier time
STARKs (2019)	Hash-based	Yes	~ O(n log(n))	O(n)	O(log(n))
Spartan (2019)	Hash-based Incremental Vector commitment	No	O(n)	O(logn) or O(√n)	O(√n)
DeepFri (2019)	Hash-based Error correction techniques	Yes	~ O(n log(n))	O(n)	O(log(n))
Nova (2022)	Hash-based commitments Proof Folding	Yes	O(log(n)) Per folding	O(log(n)) Per folding	O(1)

Products Review

Privacy-preserving Products

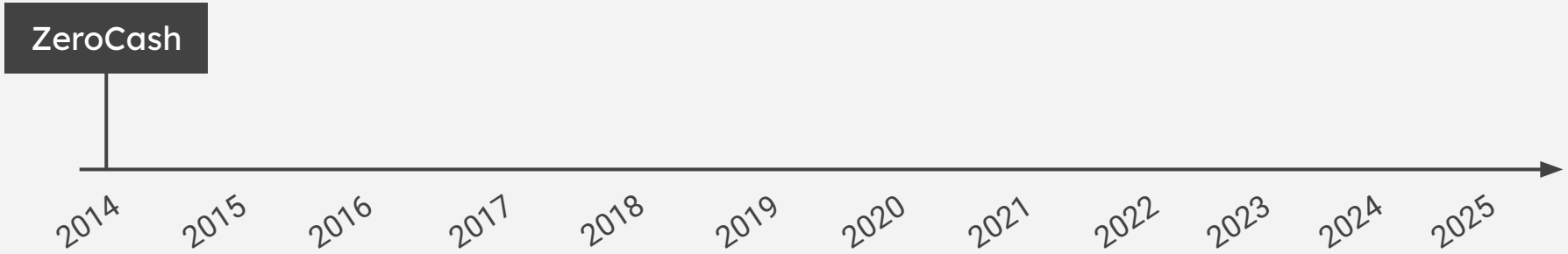
Chronography

- More than 20 products have been analyzed.
- Compared
  - Approaches
  - Achievements



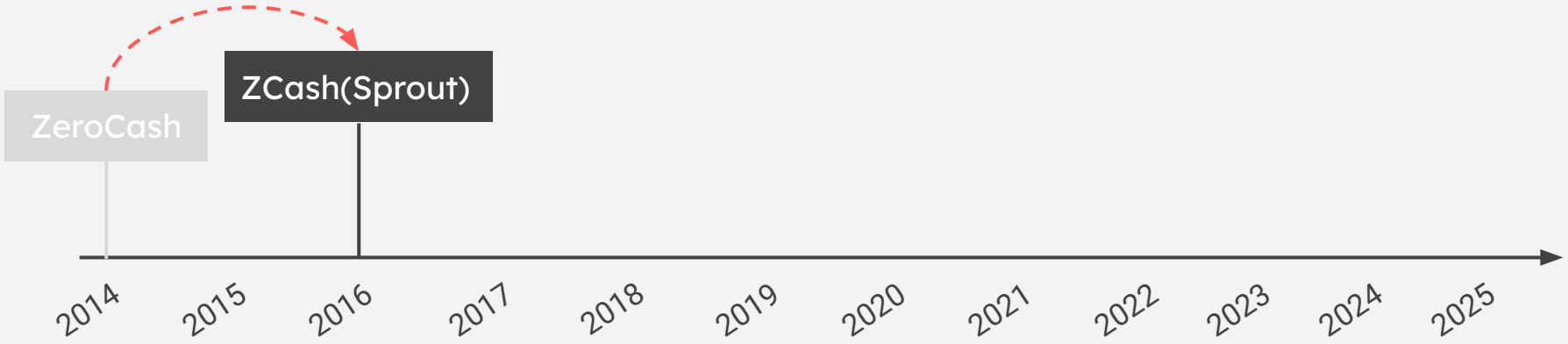
2014 - ZeroCash

- Academic paper
  - Zerocash: Decentralized Anonymous Payments from Bitcoin
- Custom pairing-based SNARKs, predates Groth16
- Anonymous transactions and unlinability
- Uses commitment, SerialNumber, MerkleTree



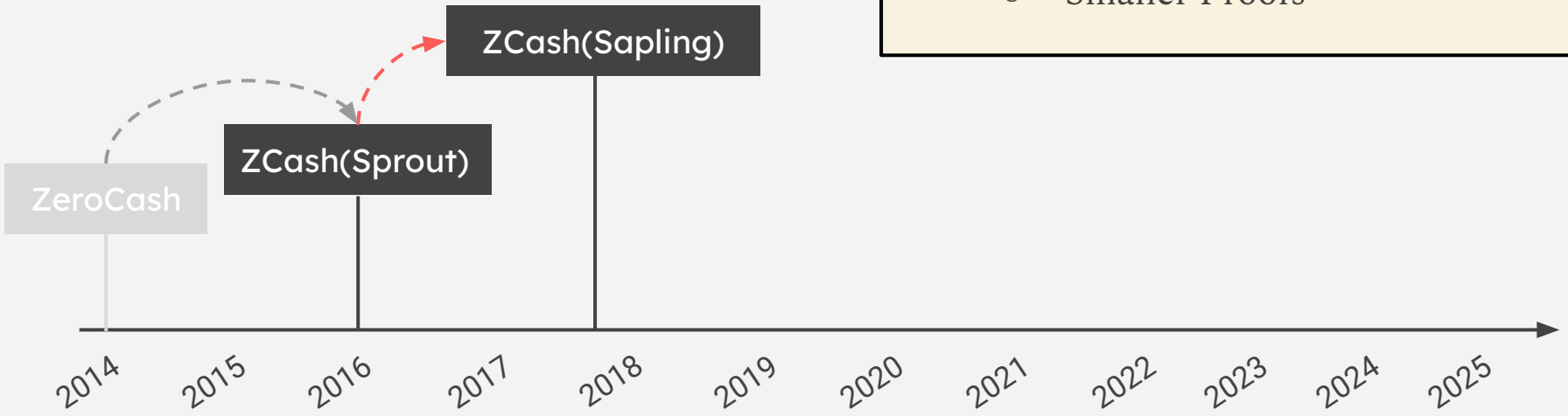
2016 - ZCash (Sprout)

Groth16 - Real-world implementation based on ZeroCash - L1 solution



2018 - ZCash (Sapling)

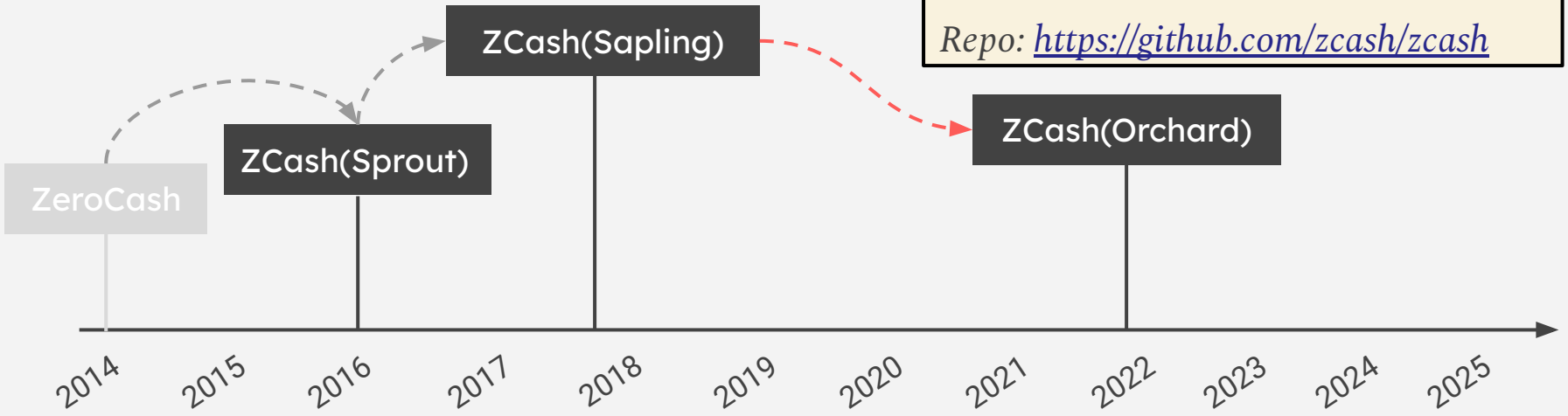
- Still Groth16
- Major Efficiency upgrades
  - Better Prover time
  - Smaller Proofs



2022 - ZCash (Orchard)

- Halo2 instead of Groth16
- No trusted setup needed
- Same performance as *Sapling*
- Recursion friendly

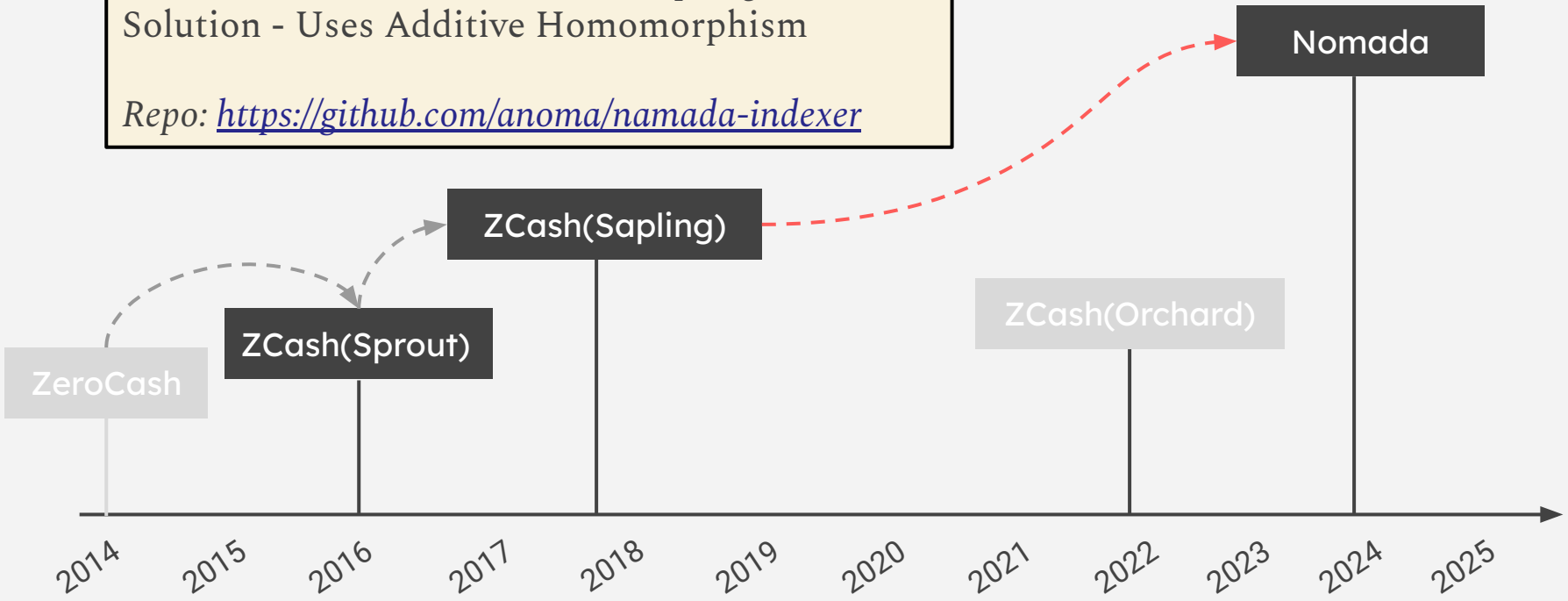
Repo: https://github.com/zcash/zcash



2024 - Nomada

Groth 16 - Extension of ZCash Sapling - L1 Solution - Uses Additive Homomorphism

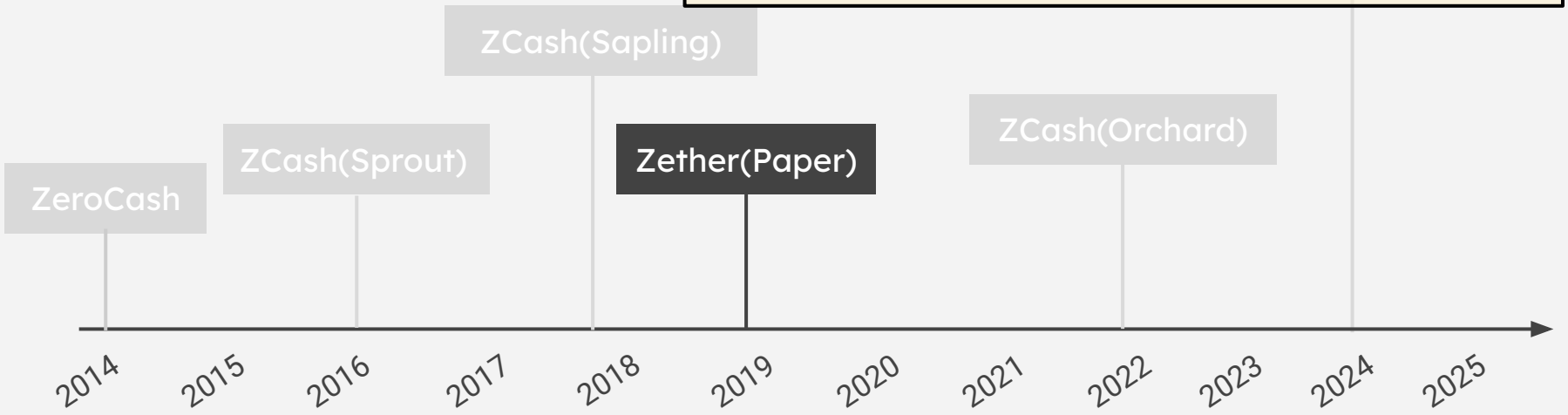
Repo: https://github.com/anoma/namada-indexer



2019 - Zether

ElGamal - Additive homomorphic - BulletProofs - Designed for EVM-like machines - Gas-costly - By default just hides the amount not sender/receiver - Additional AddOn for sender receiver anonymity - using Ring signatures.

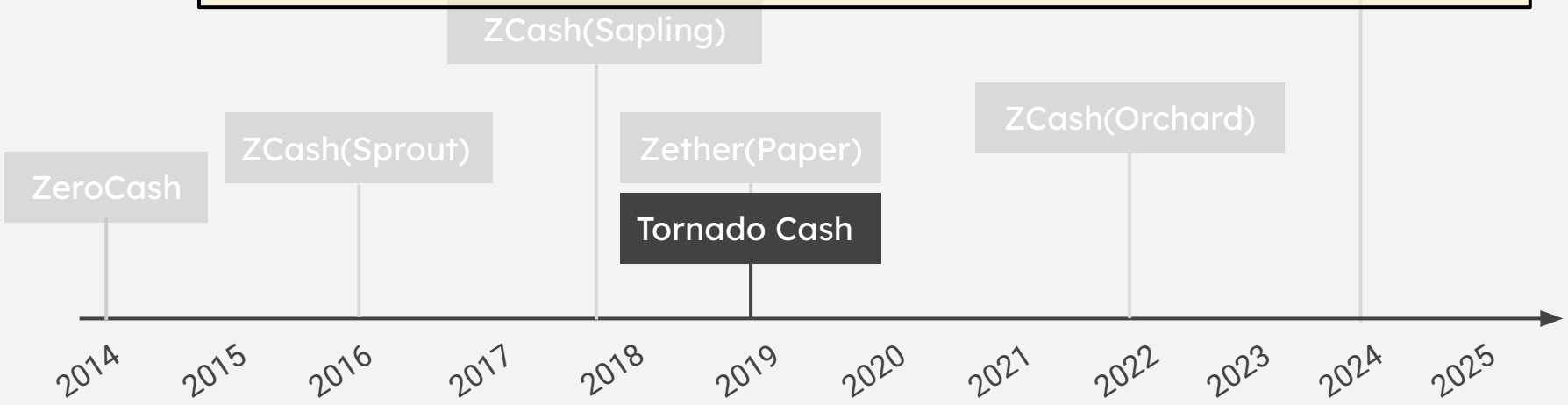
Repo: https://github.com/Consensys/anonymous-zether



2019 - Tornado Cash

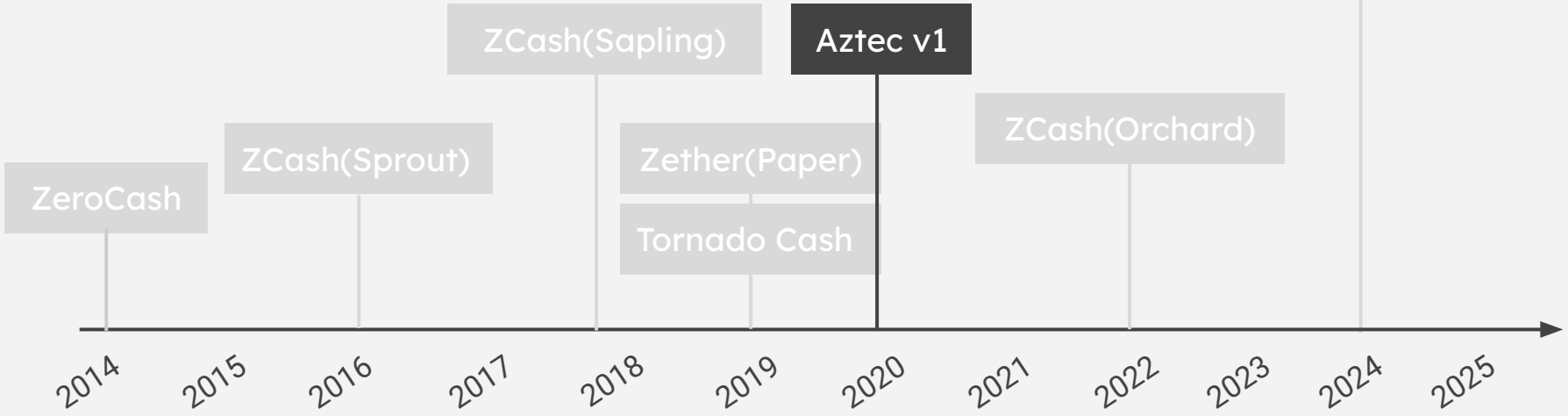
Groth16 - fixed size transfer - sender/receiver privacy - Needs Trusted setup - UTXO style - no account model - Commitment/Nullifier mechanism - no composability for transactions.

Repo: https://github.com/tornadocash



2020 - Aztec v1

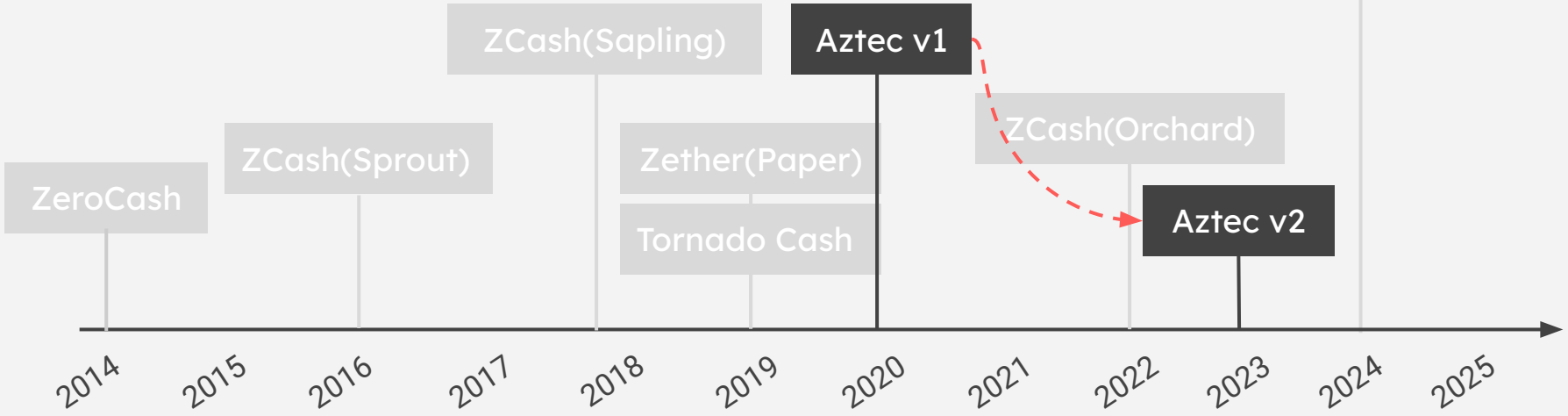
Groth 16 - Trusted setup required - No composability of transactions - Encryption path for selective disclosure and auditability - ElGamal encryption with view key - shut down due to gas inefficiency



2023 - Aztec v2

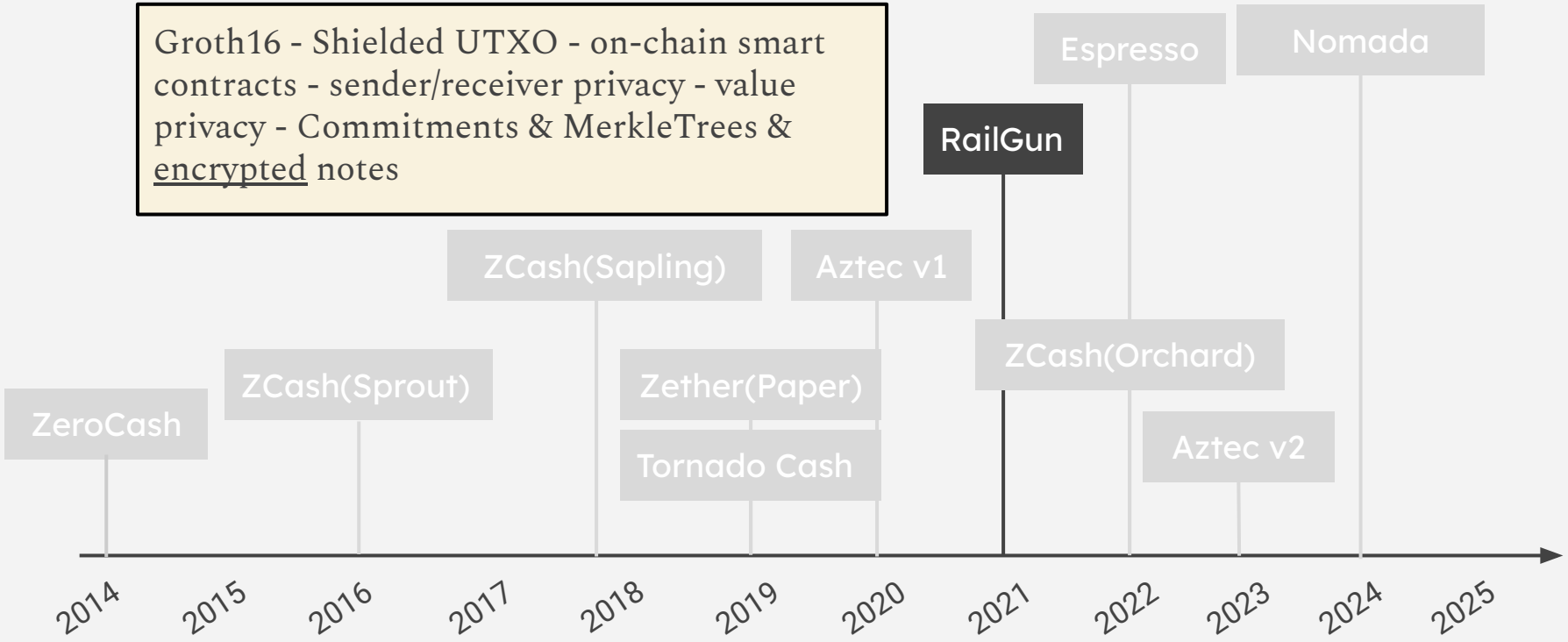
Aztec Connect - transfer & function call - shut down in March 2024 - batching - PLONK based

Repo: https://github.com/AztecProtocol/aztec-connect



2021 - RailGun v1

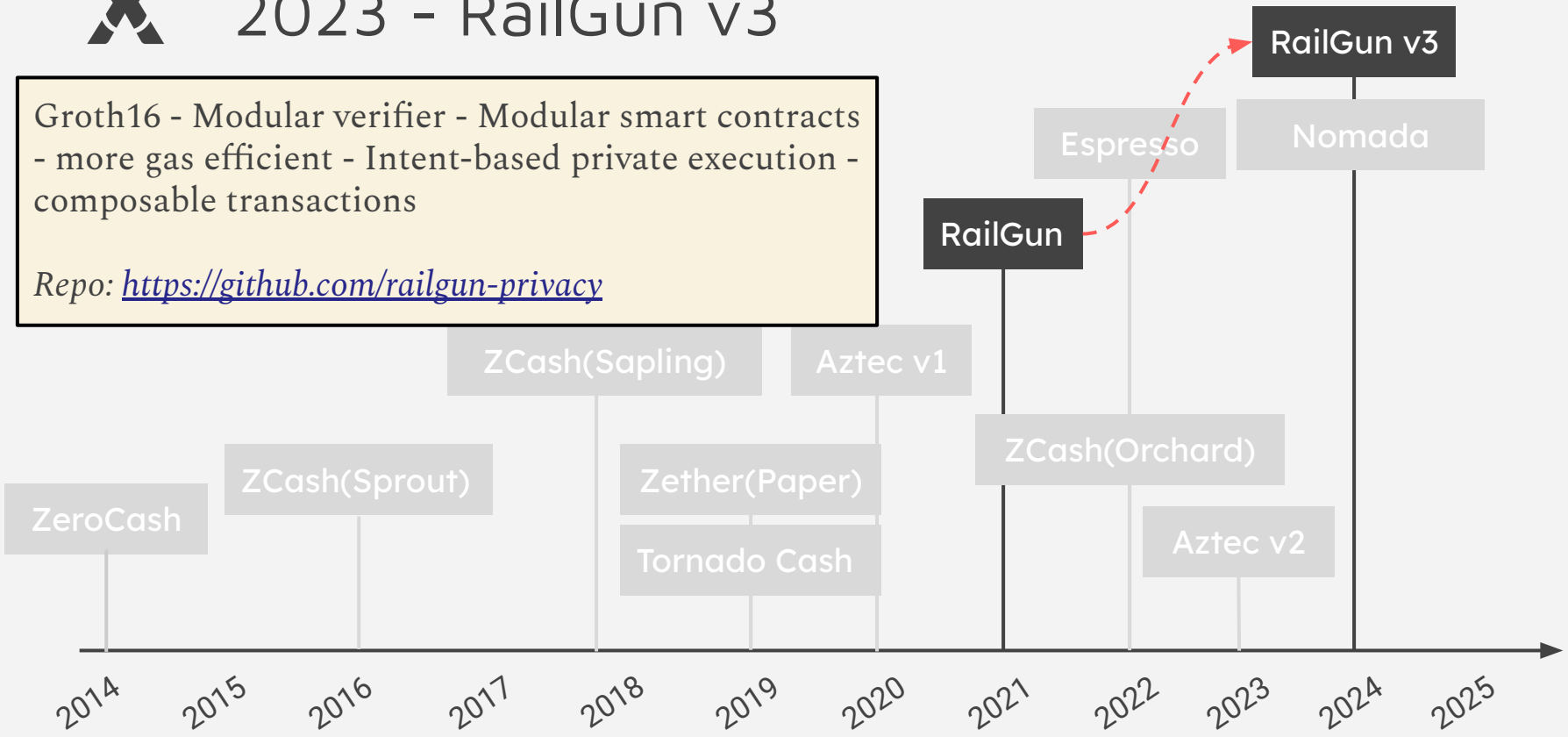
Groth16 - Shielded UTXO - on-chain smart contracts - sender/receiver privacy - value privacy - Commitments & MerkleTrees & encrypted notes



2023 - RailGun v3

Groth16 - Modular verifier - Modular smart contracts - more gas efficient - Intent-based private execution - composable transactions

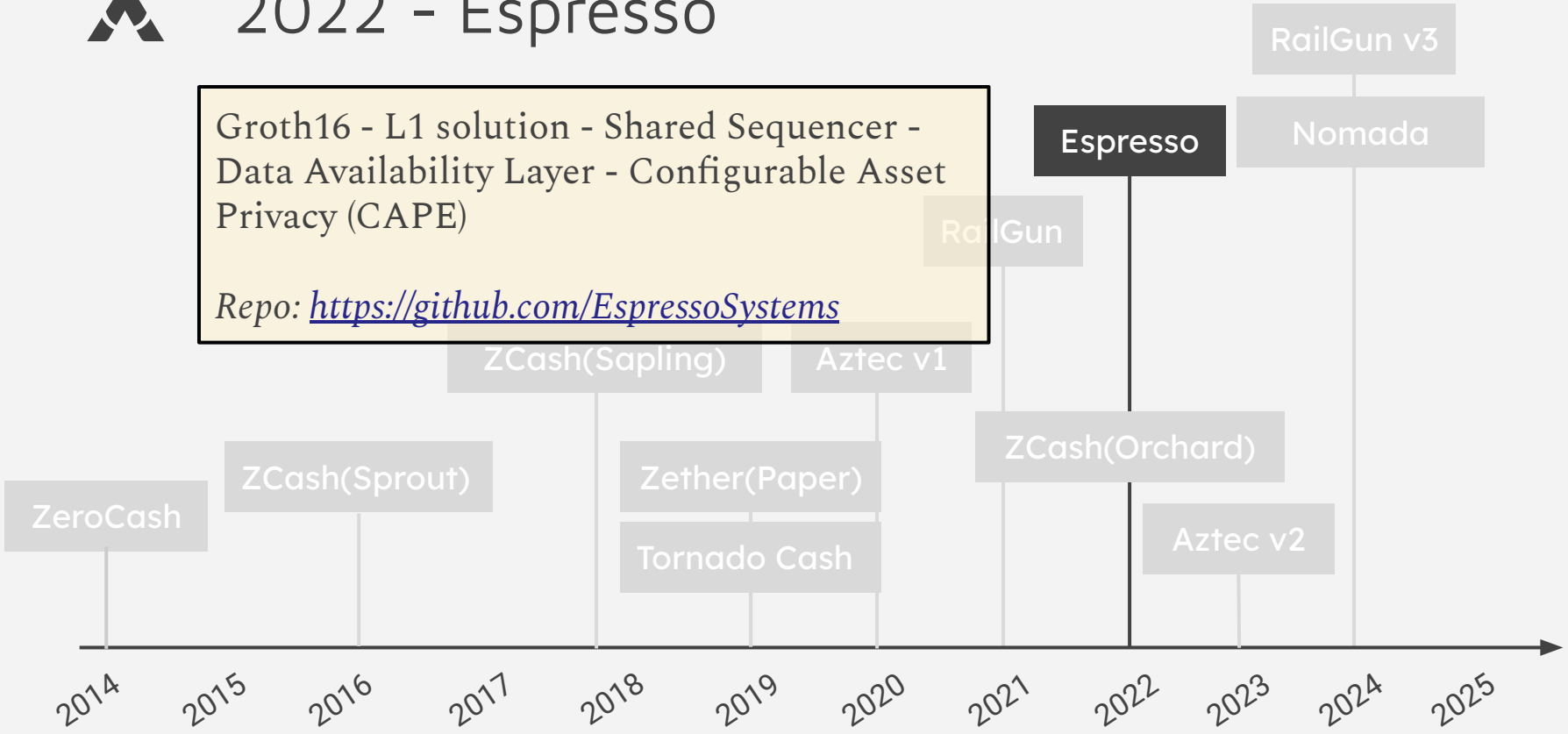
Repo: https://github.com/railgun-privacy



2022 - Espresso

Groth16 - L1 solution - Shared Sequencer - Data Availability Layer - Configurable Asset Privacy (CAPE)

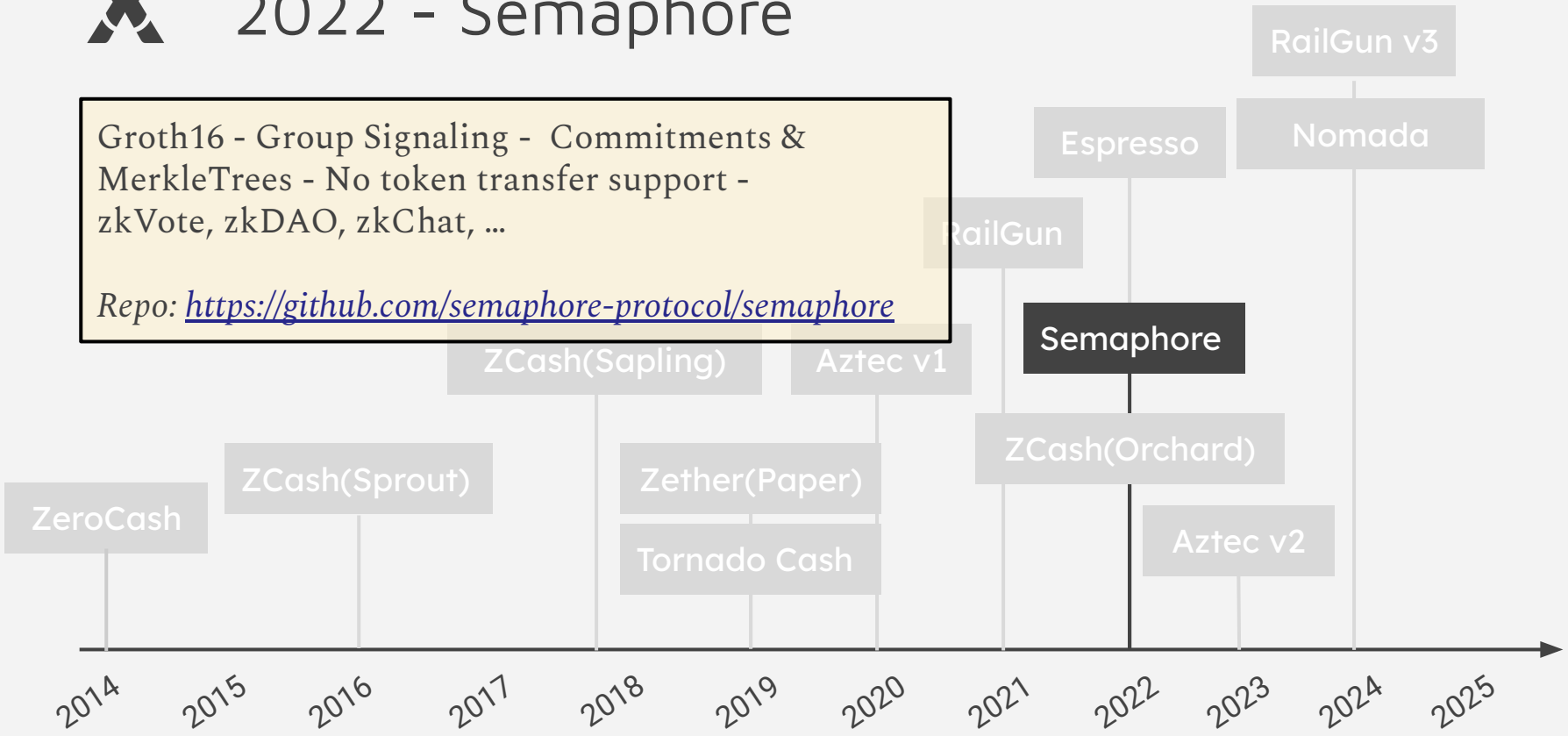
Repo: https://github.com/EspressoSystems



2022 - Semaphore

Groth16 - Group Signaling - Commitments & MerkleTrees - No token transfer support - zkVote, zkDAO, zkChat, ...

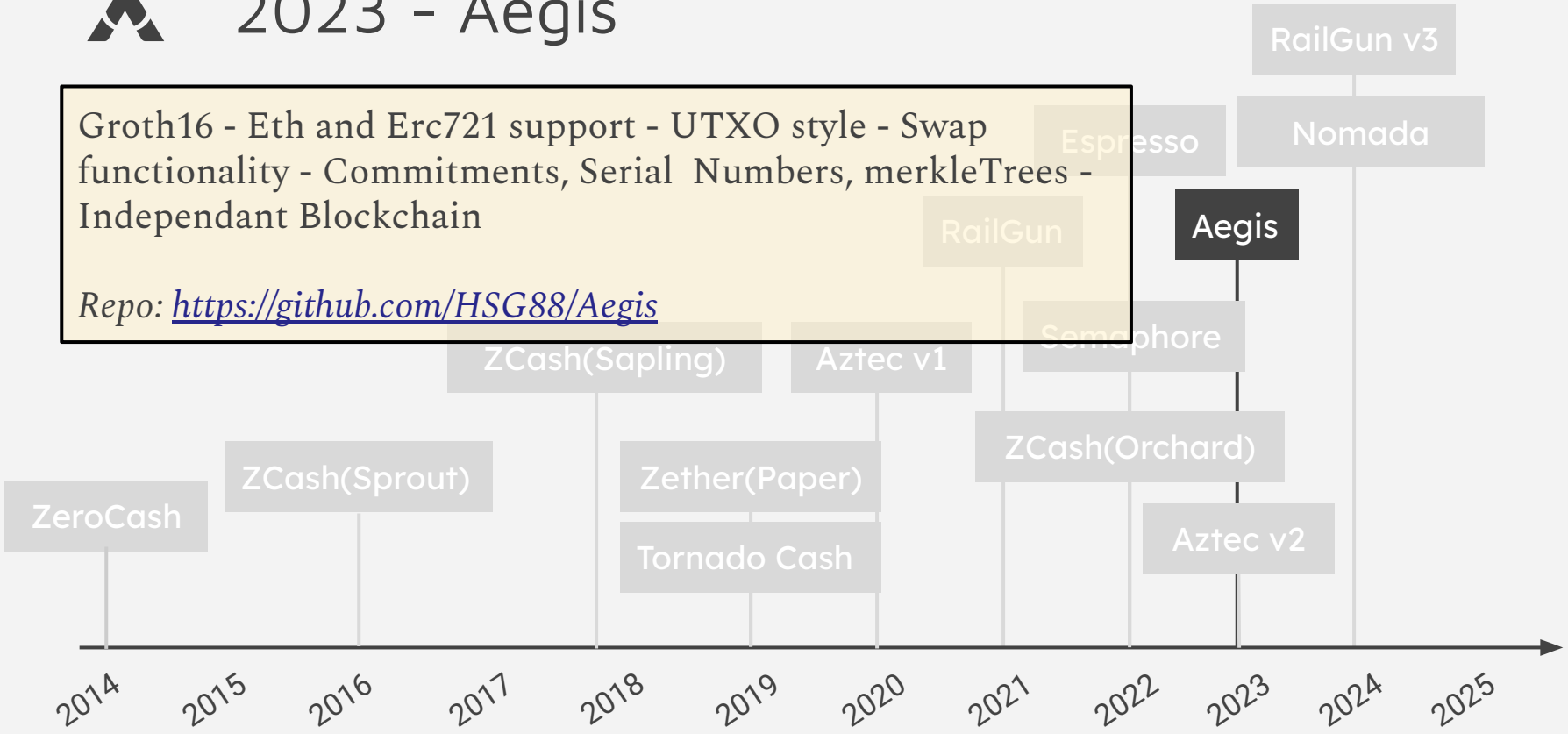
Repo: https://github.com/semaphore-protocol/semaphore



2023 - Aegis

Groth16 - Eth and Erc721 support - UTXO style - Swap functionality - Commitments, Serial Numbers, merkleTrees - Independant Blockchain

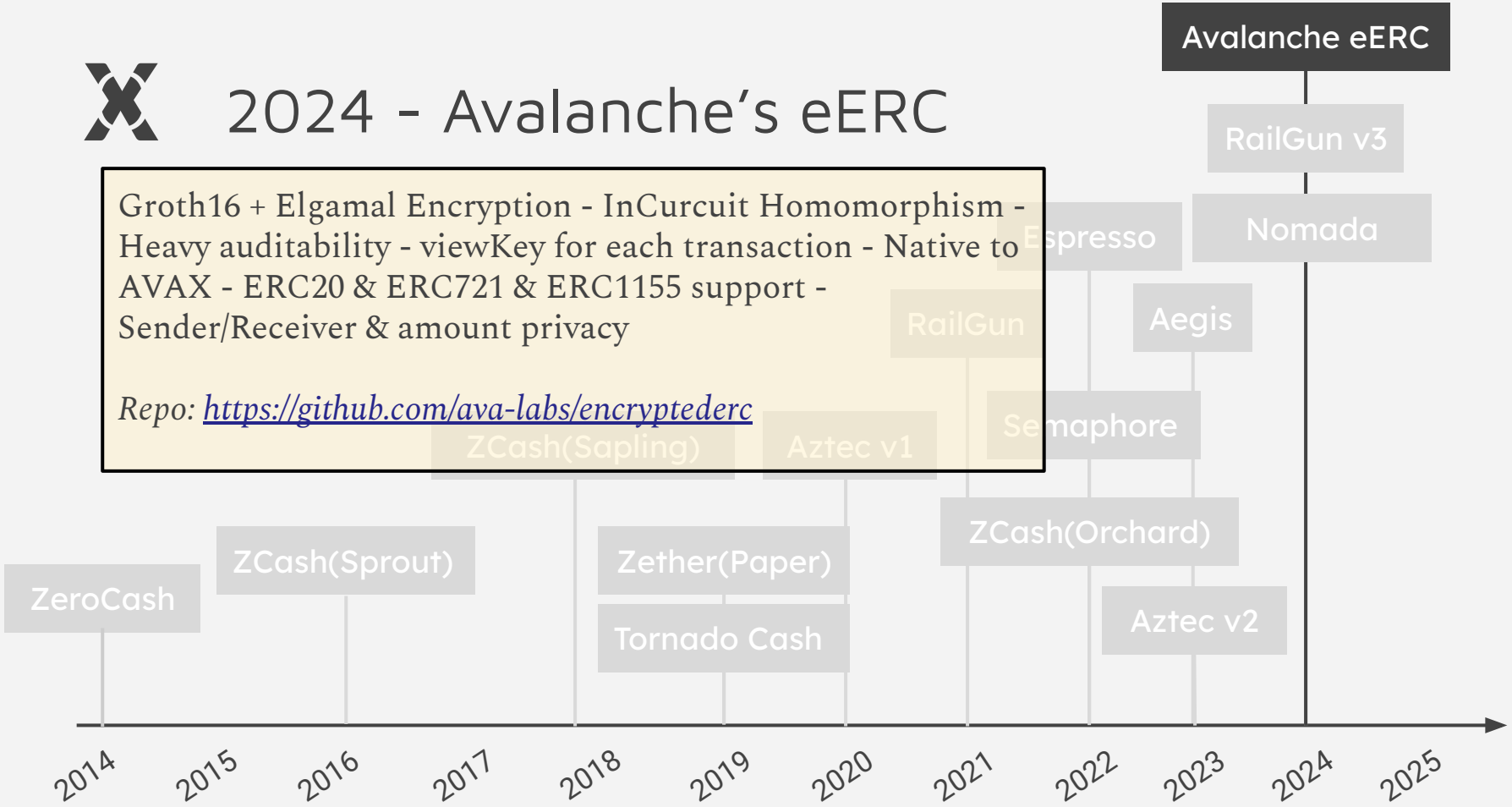
Repo: https://github.com/HSG88/Aegis



2024 - Avalanche's eERC

Groth16 + Elgamal Encryption - InCurcuit Homomorphism - Heavy auditability - viewKey for each transaction - Native to AVAX - ERC20 & ERC721 & ERC1155 support - Sender/Receiver & amount privacy

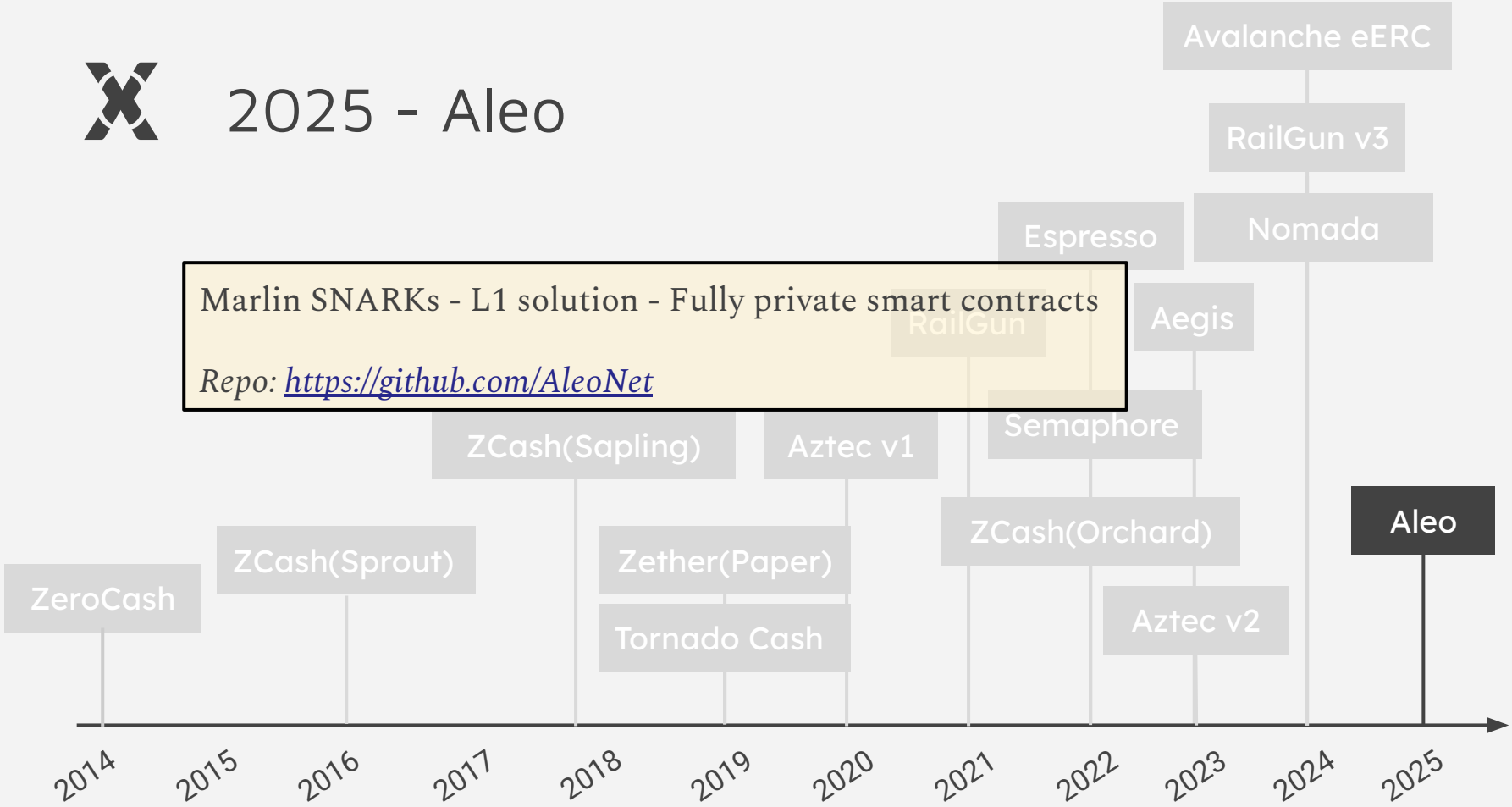
Repo: https://github.com/ava-labs/encryptederc



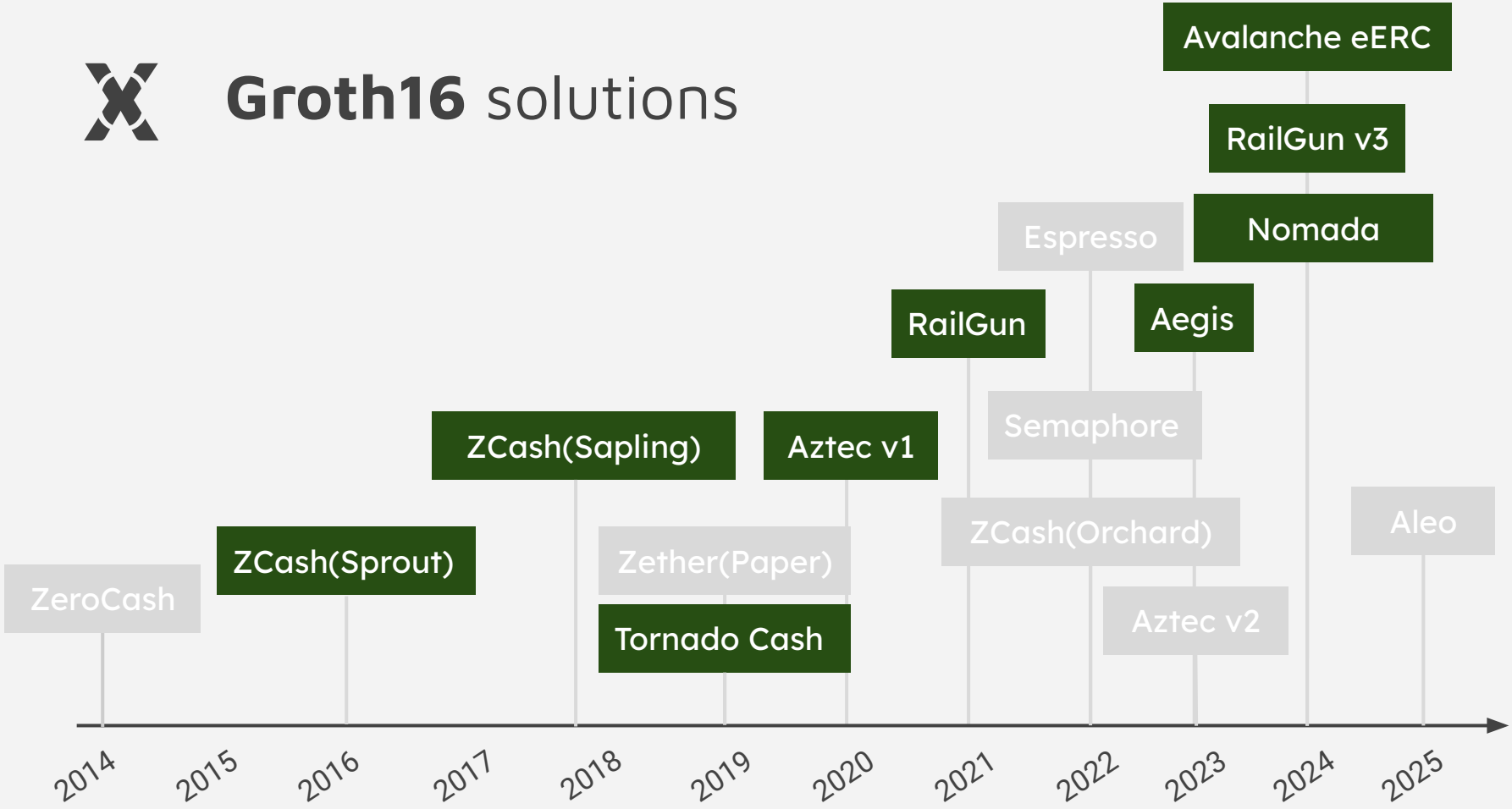
2025 - Aleo

Marlin SNARKs - L1 solution - Fully private smart contracts

Repo: https://github.com/AleoNet



Groth16 solutions

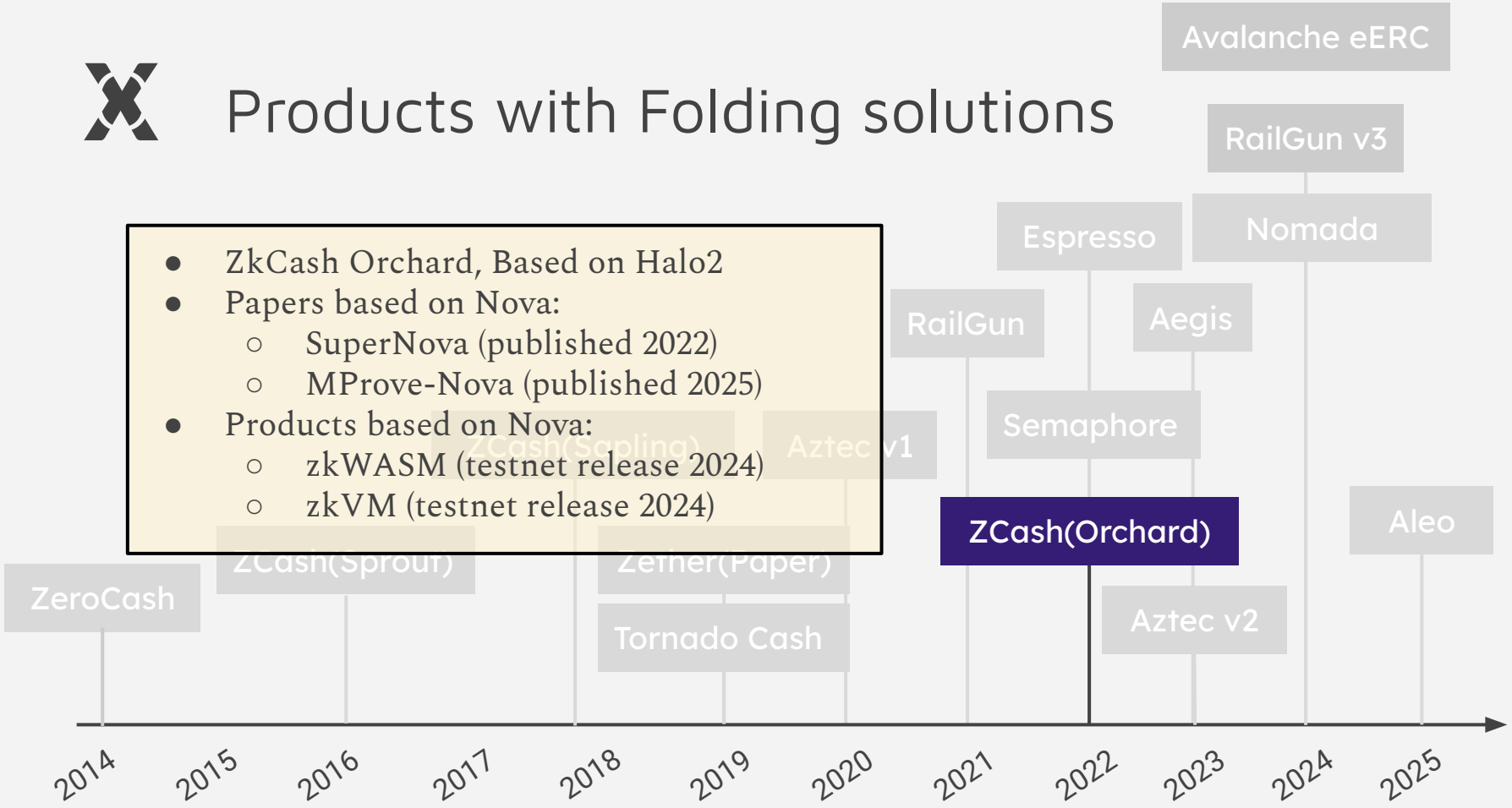


Groth16 Solutions

Name	Main functionalities	Privacy keywords	Note
ZCash Sapling	zeroCash in real-world	Shielding commitments, SerialNumbers, unlinkability	Discontinued
Tornado Cash	Mixer, Non-composable Txns	Unlikability, Fixed size mixing No amount privacy	Not much flexibility
Semaphore	zkVote, zkDAO, zkChat	Shielding commitments, SerialNumbers, unlinkability, Signal Grouping, Identity commitments	No transfer support
RailGun v3	Supports ERC20,721,1155 Composable transactions	Shielding commitments, SerialNumbers, unlinkability, Encryption	No strong auditability
Encrypted ERC	Supports ERC20,721,1155 Composable transactions	Commitments, SerialNumbers, unlinkability, Elgamal Encryption, homomorphism, native selective auditability	Avax chain dependant Prover-heavy

Products with Folding solutions

- ZkCash Orchard, Based on Halo2
- Papers based on Nova:
  - SuperNova (published 2022)
  - MProve-Nova (published 2025)
- Products based on Nova:
  - zkWASM (testnet release 2024)
  - zkVM (testnet release 2024)



Products using Encryption