**Block-Level Access Lists (BALs)**

R&D Session at Forschungsingenieurtagung – June 2025

**Duration**: 1 hr
**Moderator**: Toni Wahrstätter (@nero_eth)
**Note taker**: TBA

## Agenda

- **[10 mins]** Overview and motivation for block-level access lists
- **[30 mins]** (Breakout) discussions: parallelization, zk use cases, builder complexity
- **[20 mins]** Recap and agree on path to potential fork inclusion

## Summary

Block-Level Access Lists (EIP-7928) introduce explicit per-transaction access lists and state diffs into blocks. This enables deterministic parallelization of EVM execution and disk I/O, especially improving worst-case block processing time.

In addition to performance benefits, BALs support executionless validation for zkEVM nodes, improve sync/indexing, and enable new primitives like "fair" block-level warming and pre-execution validity checks. The burden of generating BALs falls on block builders, while clients and validators benefit from reduced overhead.

This session will explore the technical readiness, trade-offs, and implementation paths for BALs with the goal of assessing their suitability for inclusion in the Glamsterdam fork.

## Goal(s)

- Determine consensus on whether BALs unlock enough performance and protocol value to justify inclusion in a hard fork
- Evaluate open design questions (e.g., reads vs. writes in BALs)
- Identify potential implementation blockers or coordination challenges across EL clients

## Pre-reads

- EIP-7928: Block-Level Access Lists (https://eips.ethereum.org/EIPS/eip-7928)
- Block-Level Access Lists ethresear.ch thread (https://ethresear.ch/t/block-level-access-lists-bals/22331)

## Notes

*To be added after session.*