

When should we ship Prague? Should we extend the scope? What should we include if we extend the scope?

Audience of this doc: The Core Dev community.

Goal of this doc: Pitch a scope expansion for the Prague hard fork, and convince the core dev community to ship a “stretched” Prague in 2025 including PeerDAS & EOF.

Context

We (the Reth team) are back from Kenya after Interop 2024, where devnet-0 was [successfully launched and finalized](#). In every way, with regards to the devnet-0 scope the event was a success – a heroic effort from >150 people in hackathon mode for a week, including weeks of preparation from EF DevOps & Testing teams to successfully give fast feedback loops to developers during the Interop event. Really a marvel of cross-functional collaboration. Big kudos to everyone. So let’s discuss Prague in prod!

When could we ship Prague in Ethereum mainnet?

It seems like there’s 3 options:

1. Before Devcon, around November 2024:
 - a. Pros: Memetic value in shipping the Prague hardfork around/before Devcon in November 2024, to share a positive roadmap update with the community during Devcon.
 - b. Cons: Risk of Core Devs debugging bugs during Devcon. It seems poorly motivated to ship things around conferences, as we should be shipping based on what impactful features are possible.
2. After Devcon, November/December/January 2024:
 - a. This seems unwise because many people will be on vacation due to Thanksgiving at the end of November and Christmas throughout the end of December/January.
3. By end of Q1 2025:
 - a. Pros: Gives more time for testing and for being “rested” to handle a HF transition after a long sprint.
 - b. Cons: Delays Prague as currently defined by 3-4 months.

We think there’s low value in trying to ship Prague in November 2024 in an attempt to ‘turboship’ Osaka/Verkle in Q4 2025, as it is a change that requires a lot more development & testing. We generally don’t love the November → Q1 “gap” due to vacations etc., but it is what it is, and feels like a separate topic we could discuss. We are observing that this is also what happened with Cancun, so being realistic, my vote would be to aim to ship Prague in Q1 2025.

With today’s scope, delaying Prague seems to have marginal impact on users and allows us to consider expanding the scope to include more impactful features for users.

If we were to ship Prague in Q1 2025, should we extend the CL scope?

We think the answer is yes.

CL must include PeerDAS in Prague, motivation below:

1. We must increase Ethereum's Data Availability capacity before Osaka (earliest Q1 2026 IMO) based on conversations from Interop 2024 as demand is likely to arrive before then. We should be proactive instead of waiting for price discovery to happen when it might be too late for us to react.
2. Solo stakers are already reporting high bandwidth requirements due to requiring all blobs to attest on a block. This means that increasing blob count as defined in EIP-4844 is not possible.
3. In addition to that, the fact that the blob count constant is present on both EL and CL means that the option of doing a CL-only hard fork increasing blob count is also not possible. We could use this as a chance to decouple the two.
4. PeerDAS seems to be 'ahead of schedule' given the robust progress from Interop 2024.

We think PeerDAS is the most important feature for the Rollup Centric Roadmap's success and must be shipped in Prague despite previous ACD agreement of it for Osaka. Of note PeerDAS is not technically a fork because it only requires changes to the fork choice strategy, not any consensus rules, but getting benefits from PeerDAS is - because we have to actually increase the blob count target. We don't have a strong view on what else could/should be included beyond that (MaxEB?), but we'd advise against including anything else unless absolutely required.

If we were to ship Prague in Q1 2025, should we extend the EL scope?

We think the answer is yes.

Post-interop, from the EL side it seems like we're reasonably close to actually being ready to ship things that ACDE was accepted so far:

- Devnet-0 successfully finalized with most clients.
- We need to decide on EIP7702 on ACDE, and if so, implement it as a replacement for EIP-3074. We think it makes sense to do so as it's a more native-AA compatible solution because of the new transaction type vs an opcode. We also think committing to the nonce should be optional to allow for more advanced use cases, while allowing people to be more mindful of security and always commit to it – this is specified [here](#).
- We need to patch EIP-2935 with the agreed upon replacement of the backfill mechanism.

This gives us time to ship an additional needle-moving feature for Prague LEs, which we think must be EOF, reasoning below:

1. Improve smart contract security by allowing tools like Foundry & Slither to perform more structured analysis over the bytecode.

2. Improve EVM developer experience by allowing the removal of the 24KB contract size limit. This is achieved by removing JUMPDEST analysis, which historically has been the motivator behind the contract size limit as it could be a DOS vector during syncing in the worst case. Devs have worked around this by DELEGATECALL'ing contracts, but is 1) error prone due to proxy complexity, 2) more expensive.
3. Sets up for better native account abstracted accounts by creating explicit "validation code" regions, separate from "execution code".
4. EOFv1+ accounts are easier to include in Verkle Tree witnesses because of well-defined code chunking & partial code load semantics.
5. L2 EVM experimentation. We think the L1 still has responsibility for helping L2s succeed here, because L2s are still scared to diverge L1.
6. The EOF specification is finalized, and multiple clients have implemented it.
7. It's an EVM-only change, which in Reth's case it was implemented by one person in 2 months. It's an isolated change which can be fuzzed and we know how to test EVM.
8. Vyper is onboard but haven't prioritized it due to no ACDE approval. Solidity just voiced support. Ipsilon team started migrating solidity old EOF implementation to Prague EIP.

Our suggested next steps:

1. It is finally the time for ACDE to confirm, and not just consider, the inclusion of EOF for Prague.
2. Upon inclusion, Solidity & Vyper should implement the latest EOF spec according to the meta-EIP, and make no further changes to it for L1 usage.
3. EL teams should go above and beyond to help the testing teams with EOF testing, given it'd take away their time from testing PeerDAS.

EOF is the right "t-shirt size" for the Q1 2025 deadline, and we don't think anything else should be included beyond that for Prague. Q1 2025 gives us adequate amount of time for ELs to properly test EOF, and we expect that ELs will also have time to start working on Verkle before Prague is on mainnet. Finally, we think that EL teams should stretch and start working on EIP-4444 as [discussed](#), but that is of separate interest as it doesn't require a hard fork.

Conclusion / Recap

1. Let's ship Prague in Q1 2025.
2. CL side, let's include PeerDAS, and discuss what else could be included, if anything.
3. EL side, let's include EOF, replace EIP3074 with EIP7702, patch EIP2935, and discuss what else could be included, if anything.

Succeeding at the above would result in:

1. Alleviating pressure on validators' networking layer.
2. Proactively addressing upcoming blob price increase for rollups due to demand flowing in the next 2 years.
3. Encourage EVM innovation for UX, DevEx, and performance on L2.

It would be extremely inspiring if we shipped the above, and would reinforce Ethereum's market leadership and be an irrefutable proof of the Core Devs' shipping culture being more effective than ever.