

Agenda

1. Stateless today
2. High-level zkEVM discussion
3. Assessing stateless urgency
4. Break (10m)
5. Interaction with other changes (CR & AA)

Stateless today

Ignacio Hagopian (@jsign)

2025.06.13

Definition

Verify a block without requiring all the state


Roadmap fit


- Scale L1
- Scale L2
- Improve UX

**Where does
stateless fit here?**

Roadmap fit

- **Scale L1**
 - Attestors can keep attesting
 - Nodes can keep validating trustlessly
- **Scale L2**
- **Improve UX**
 - Enable more scaling -> lower fees

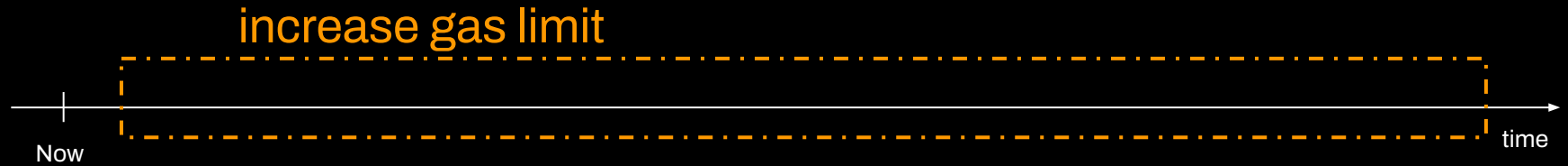
 Must do; otherwise, the network or core values break

 Important, but the network won't break without them

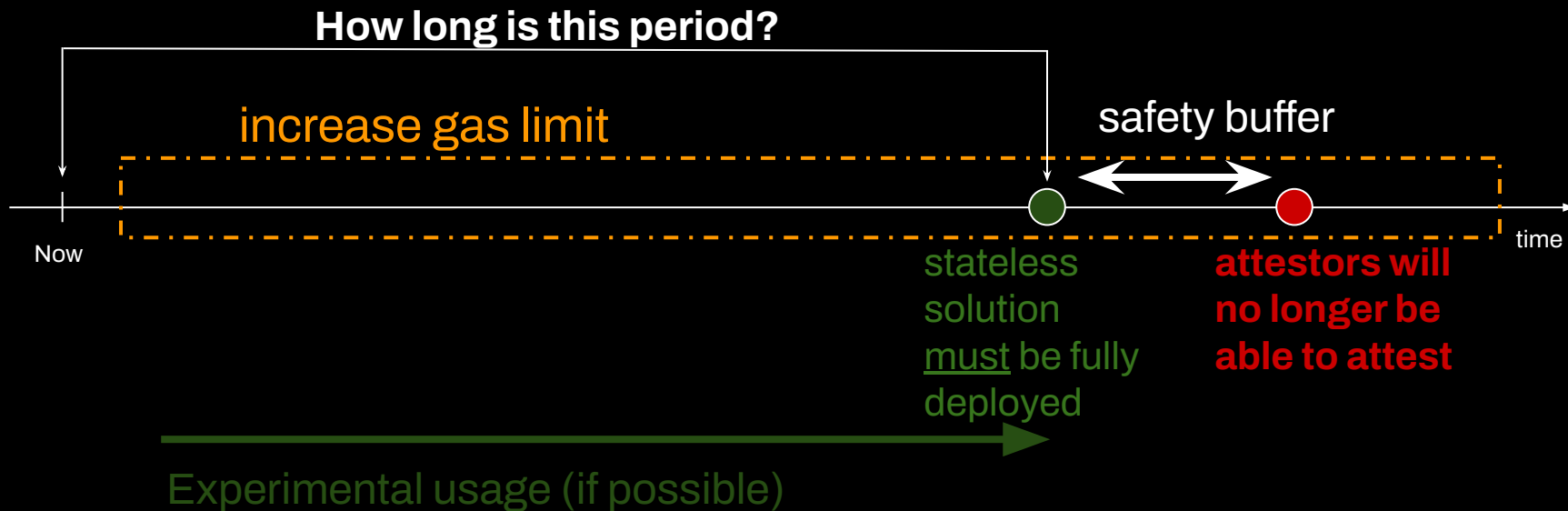
Not only why but also when

- If unclear can lead to:
 - Put the protocol at risk
 - Opportunity costs
 - Ineffective discussions
- Core-dev time usage efficiency

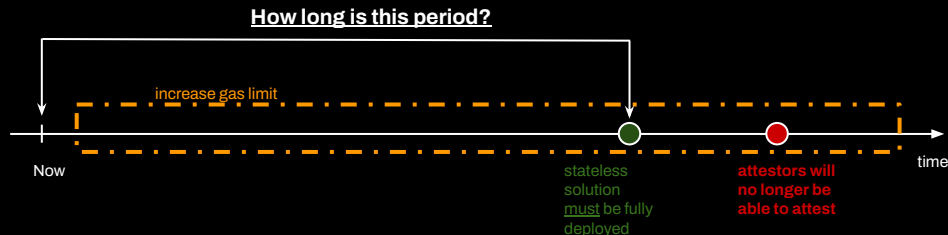
Timeline



Timeline



Timeline



Trigger event variables

- DB size
- Total read / writes
- DB compactions
- Syncing time

Knobs / Trends

- Gas limit
- Access patterns
- HW improvements

Tree changes

- What happened to Verkle?
- Binary tree:
 - Blocker or an “efficiency improvement”?
 - Merkelization hash function / field?
 - AA tx witness efficiency?

Code-chunkification

- Worst-case proving
- AA tx witness efficiency?



Merkelizing Bytecode: Options & Tradeoffs

■ Execution Layer Research ■ stateless



Public mempool health

- Stateless nodes don't have the data to check tx validity
- FOCIL quality depends on healthy mempool
- AA changes tx validity rules

Public mempool health

- Partially stateful nodes (VOPS)
 - No tx witness
 - Requires some form of syncing
- Tx witnesses + proved state-diff
 - No syncing required
 - Tx sender must resolve getting witness

State expiry

- Long history of research (hard problem!)
- State usage efficiency
- Active state size
- In-protocol vs out-protocol

zkEVMs

- Fundamental change to mental model:
 - Constraint (HW, power, assumptions)
 - Gas costs
 - Next bottlenecks?
- A new tool to solve other hard problems

Measure²

- Tree change, hash func, code-chunking...
- Invest in building tooling if needed:
 - Long-lasting vs single-shot efforts
 - Tunable for protocol changes simulation
 - Composable -> Higher leverage