Team Information

- Team Members:
- @dmarz
- @dmarz

Project Repository

- GitHub Link:

sui/narwhal at main · dmarzzz/sui · GitHub

Project Goal

[

Screenshot 2024-06-24 at 11.52.45 AM

2006×1118 111 KB

](https://collective.flashbots.net/uploads/default/original/2X/9/91ab18ef219ff75d08e95cfe8d3435bf78a25dc7.jpeg)

- Source : https://www.youtube.com/watch?v=K5ph4-7vvHk
- Brief Description:
- “We design and evaluate a mempool protocol, Narwhal, specializing in high-throughput, reliable dissemination and storage of causal histories of transactions. Narwhal tolerates an asynchronous network and maintains high performance despite failures. Narwhal is designed to easily scale-out using multiple workers at each validator, and we demonstrate that there is no foreseeable limit to the throughput we can achieve… To make a robust Mempool we design Narwhal, a DAG-based, structured Mempool which implements causal order reliable broadcast of transaction blocks, exploiting the available resources of validators in full. Combining the Narwhal Mempool with HotStuff (Narwhal-HS) provides good throughput even under faults or unstable network conditions (but at an inevitable higher latency). To reduce latency under faults and asynchrony, we can extend Narwhal with a random coin to provide asynchronous consensus, which we call Tusk. Tusk is a fully asynchronous, wait-free consensus where each party decides the agreed values by examining its local DAG without sending any additional messages.” - from the Mysten Labs paper that introduces Narwhal
- Goal

: Run Mysten Labs’ Narwhal implementation of a DAG-based mempool locally with multiple primaries and multiple workers per primary and then add that into Reth.

- “We design and evaluate a mempool protocol, Narwhal, specializing in high-throughput, reliable dissemination and storage of causal histories of transactions. Narwhal tolerates an asynchronous network and maintains high performance despite failures. Narwhal is designed to easily scale-out using multiple workers at each validator, and we demonstrate that there is no foreseeable limit to the throughput we can achieve… To make a robust Mempool we design Narwhal, a DAG-based, structured Mempool which implements causal order reliable broadcast of transaction blocks, exploiting the available resources of validators in full. Combining the Narwhal Mempool with HotStuff (Narwhal-HS) provides good throughput even under faults or unstable network conditions (but at an inevitable higher latency). To reduce latency under faults and asynchrony, we can extend Narwhal with a random coin to provide asynchronous consensus, which we call Tusk. Tusk is a fully asynchronous, wait-free consensus where each party decides the agreed values by examining its local DAG without sending any additional messages.” - from the Mysten Labs paper that introduces Narwhal
- Goal

: Run Mysten Labs’ Narwhal implementation of a DAG-based mempool locally with multiple primaries and multiple workers per primary and then add that into Reth.

Challenges

- Challenges Faced:
- Narwhal seems to be no longer actively maintained by Mysten Labs
- Narwhal Benchmark Broken · Issue #17988 · MystenLabs/sui · GitHub- Issue I created since the benchmark code is no longer working. I fixed a few bugs but then ended up ditching this approach as I am unfamiliar with the python benchmarking tool that was used and that was giving me more python related issues than I had patience for.

- Issues on running Narwal locally - Technical Support - Sui Developer Forum- After posting in their discord one of their devs forwarded my message with more details to the forum.
- Narwhal Benchmark Broken · Issue #17988 · MystenLabs/sui · GitHub- Issue I created since the benchmark code is no longer working. I fixed a few bugs but then ended up ditching this approach as I am unfamiliar with the python benchmarking tool that was used and that was giving me more python related issues than I had patience for.

- Issues on running Narwal locally - Technical Support - Sui Developer Forum- After posting in their discord one of their devs forwarded my message with more details to the forum.

- I am a rust noob
- Narwhal seems to be no longer actively maintained by Mysten Labs

- Narwhal Benchmark Broken · Issue #17988 · MystenLabs/sui · GitHub- Issue I created since the benchmark code is no longer working. I fixed a few bugs but then ended up ditching this approach as I am unfamiliar with the python benchmarking tool that was used and that was giving me more python related issues than I had patience for.
- Issues on running Narwal locally - Technical Support - Sui Developer Forum- After posting in their discord one of their devs forwarded my message with more details to the forum.
- Narwhal Benchmark Broken · Issue #17988 · MystenLabs/sui · GitHub- Issue I created since the benchmark code is no longer working. I fixed a few bugs but then ended up ditching this approach as I am unfamiliar with the python benchmarking tool that was used and that was giving me more python related issues than I had patience for.
- Issues on running Narwal locally - Technical Support - Sui Developer Forum- After posting in their discord one of their devs forwarded my message with more details to the forum.
- I am a rust noob

Current State

- Project Status at the End of Hackathon:
- I was able to get 2 Primary Nodes and 2 Work nodes running

2024-06-14T20:19:02.483201Z DEBUG narwhal_primary::state_handler: state handler: received 4 sequenced certificates at round 1754

- The primaries are able to progress DAG rounds amongst each other with no transactions in any rounds
- The workers are able to talk to each other
- The workers ARE NOT

able to send transactions to each other

- There is a script to send a transaction to the workersui/examples2/src/main.rs at main · dmarzzz/sui · GitHub

- I was able to get 2 Primary Nodes and 2 Work nodes running
- The primaries are able to progress DAG rounds amongst each other with no transactions in any rounds
- The workers are able to talk to each other
- The workers ARE NOT

able to send transactions to each other

- There is a script to send a transaction to the workersui/examples2/src/main.rs at main · dmarzzz/sui · GitHub

Additional Notes

- Any other information:
- I will eventually try to run this again once I can chat with Mysten Labs team and figure out which release works properly
- Huge S/O to @halo3mic and @ferranbt for the Rust help

- I will eventually try to run this again once I can chat with Mysten Labs team and figure out which release works properly

- Huge S/O to @halo3mic and @ferranbt for the Rust help