A Data-Availability Layer (DAL) for Tezos

François Thiré

May 31, 2022

Nomadic Labs

Scalability for Blockchains

- Execution scalability
- State scalability
- Bandwith scalability

Scalability for ~~Blockchains~~ Tezos

- Execution scalability ↪ enshrined rollups
- State scalability ↪ enshrined rollups
- Bandwith scalability

Rollups: TORU, SCORU, ZK-rollups

Scalability for ~~Blockchains~~ Tezos

- Execution scalability ↪ enshrined rollups
- State scalability ↪ enshrined rollups
- Bandwith scalability ↪ **Data-availability layer (DAL)**

Rollups: TORU, SCORU, ZK-rollups

Vocabulary

In the following:

- L1 refers to the current chain in the Tezos protocol
- L2 refers to the implicit chains built by the rollups
- An L1 operation is an operation for the L1
- An L2 operation is an operation for the L2 dedicated to a particular rollup

Optimistic rollups

Question: What is the **network flow** of an L2 operation?

Optimistic rollups

Question: What is the **network flow** of an L2 operation?

1. The user sends its operation to the rollup operator
2. The rollup operator batches operations
3. The batch is sent to the L1 via the gossip network (using the mempool)
4. The batch is included into a block by a baker
5. The operation can be executed by the rollup operator

Optimistic rollups

Question: What is the **network flow** of an L2 operation?

1. The user sends its operation to the rollup operator
2. The rollup operator batches operations
3. The batch is sent to the L1 via the gossip network (using the mempool)
4. The batch is included into a block by a baker
5. The operation can be executed by the rollup operator

Consequences:

- Every L2 operation is included in an L1 block
- The bandwidth of all the nodes in the network becomes the limiting factor for scalability

The data-availability problem

- Safety of optimistic rollups depends on the availability of L2 operations
- Not including every L2 operation into an L1 block breaks down the bandwidth limiting factor
- If L2 operations are not in L1 blocks, how to guarantee the availability of L2 operations?

The data-availability layer

- A layer between the L1 and the L2 to ensure data-availability
- Permission-less and optional
- L2 operations only go through the data-availability layer

How it works

- The L1 provides for the DAL a list of slots (the number and the size of slots is fixed)

How it works

- The L1 provides for the DAL a list of slots (the number and the size of slots is fixed)
- A rollup subscribes to one or several slots (several rollups can use the same slot)

How it works

- The L1 provides for the DAL a list of slots (the number and the size of slots is fixed)
- A rollup subscribes to one or several slots (several rollups can use the same slot)
- A user posts a commitment of the data onto the L1 which allows to prove that a blob of data belongs this slot.

How it works

- The L1 provides for the DAL a list of slots (the number and the size of slots is fixed)
- A rollup subscribes to one or several slots (several rollups can use the same slot)
- A user posts a commitment of the data onto the L1 which allows to prove that a blob of data belongs this slot.
- Stakeholders commit on whether the data are available via endorsements

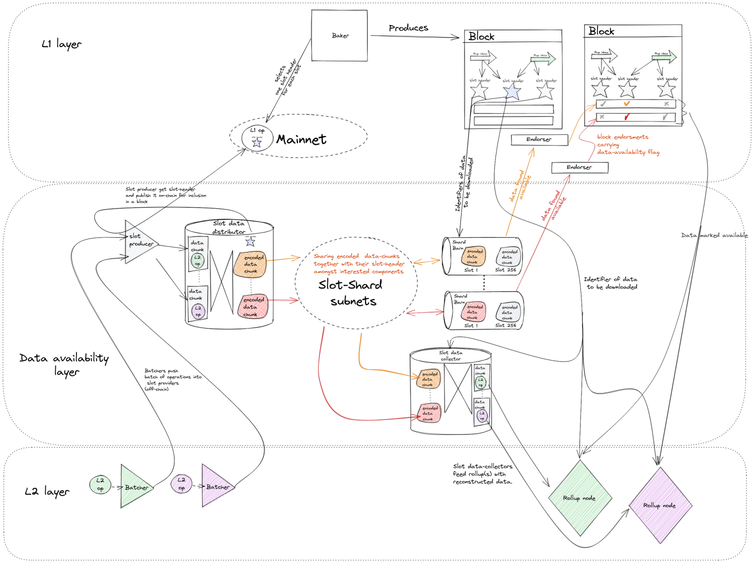
How it works

- The L1 provides for the DAL a list of slots (the number and the size of slots is fixed)
- A rollup subscribes to one or several slots (several rollups can use the same slot)
- A user posts a commitment of the data onto the L1 which allows to prove that a blob of data belongs this slot.
- Stakeholders commit on whether the data are available via endorsements
- The L1 uses those commitments of the stakeholders to decide whether the data is available

How it works

- The L1 provides for the DAL a list of slots (the number and the size of slots is fixed)
- A rollup subscribes to one or several slots (several rollups can use the same slot)
- A user posts a commitment of the data onto the L1 which allows to prove that a blob of data belongs this slot.
- Stakeholders commit on whether the data are available via endorsements
- The L1 uses those commitments of the stakeholders to decide whether the data is available
- If the data are available, it is the responsability of the rollup to download those data and execute the operations contained in it.

How it works: Big picture



Ensuring data-availability

How the data-availability layer can ensure that the data are available?

Ensuring data-availability

How the data-availability layer can ensure that the data are available?

- Trusting the stakeholders of the L1
- Using also all the nodes of the L1

Data-availability is solely ensured by the stakeholders

Consequences:

- If enough endorsers lie, the L1 will declare the data as available while they are not.
- If enough endorsers are lazy (always declare data are unavailable), the DAL cannot be used.

Thanks to cryptography and erasure-codes, we only need 20% of honest stakeholders.

Asking nodes to sample data

- Is the hypothesis of trusting 20% of honest stakeholders is too strong? This includes a bug in the software.

Asking nodes to sample data

- Is the hypothesis of trusting 20% of honest stakeholders is too strong? This includes a bug in the software.

This hypothesis can be mitigated using the sampling method:

- all the nodes of the L1 (regular nodes, indexers, stakeholders) sample the data onto the DAL
- A block is propagated only if:
    1. data are declared available by the protocol
    2. The sampling for this block succeeded (the fork chain rule is changed).

Challenge

Open question

Under which hypothesis sampling is better than trusting part of the stakeholders or vice versa?

Challenge

Open question

Under which hypothesis sampling is better than trusting part of the stakeholders or vice versa?

- On how many levels should the nodes sample data? (bootstrapping could be stuck)

Challenge

Open question

Under which hypothesis sampling is better than trusting part of the stakeholders or vice versa?

- On how many levels should the nodes sample data? (bootstrapping could be stuck)
- How many samples a node needs to download at each level? What about a change of protocols?

Challenge

Open question

Under which hypothesis sampling is better than trusting part of the stakeholders or vice versa?

- On how many levels should the nodes sample data? (bootstrapping could be stuck)
- How many samples a node needs to download at each level? What about a change of protocols?
- What should happen when sampling failed?

Challenge

Open question

Under which hypothesis sampling is better than trusting part of the stakeholders or vice versa?

- On how many levels should the nodes sample data? (bootstrapping could be stuck)
- How many samples a node needs to download at each level? What about a change of protocols?
- What should happen when sampling failed?
- What about an L1 network topology where all the bakers are a clique?

Bibliography

- A formal presentation of data-availability:
  `https://arxiv.org/pdf/1809.09044.pdf`
- The current in-progress specification for the Tezos DAL:
  `https://nomadic-labs.gitlab.io/das-design/`