Building on the Arbitrum Nitro stack with Avail DA

What is Nitro?

Nitro is designed by Arbitrum, a software stack to build optimistic rollups, powering Arbitrum One and Arbitrum Nova. These chains are Optimistic L2 protocols that inherit Ethereum-level security.

You can read more about Nitro in:

1. The Nitro whitepaper on github(opens in a new tab)
2. Arbitrum docs.(opens in a new tab)

Arbitrum Orbit integration with Avail DA

Arbitrum Orbit chains can integrate with Avail DA using a number of different configurations. You can read more about themin our blog(opens in a new tab).

Let's take a look at the transaction Lifecycle:

1. Transaction Submission
2.
    - Process
3.
    - : Users send transactions to the Sequencer.
4.
    - Role
5.
    - : The Sequencer temporarily holds the transactions before they are batched.
6. Batch Processing
7.
    - Component
8.
    - :Sequencer
9.
    - Function
10.
    - : Collects and orders transactions into batches for more efficient processing.
11. Batch Submission on Avail
12.
    - component
13.
    - :arbnode/batch-poster
14.
    - Process
15.
    - : Sequencer posts a batch of L2 transactions onto the underlying data availability provider.
16. BlobPointer Reference Creation
17.
    - Outcome
18.
    - : Avail returns a unique transaction reference with Merkle proof of
19.
    - batch submission.
20. BlobPointer Submission to Settlement layer (Arbitrum One, Ethereum etc.)
21.
    - Process
22.
    - : BlobPointer with Avail header byte(0x0a) is being sent over
23.
    - tosequencerInbox
24.
    - rollup contract for on-chain DA verification over Avail
25.
    - bridge and batch addition to canonical chain.
26.
    - component:
27.

- SequncerInbox.sol (opens in a new tab)

28. on-chain Data availability verification
29.
    - Component
30.
    - :Avail bridge (opens in a new tab)
31.
    - Function
32.
    - : VerifyMerkle proof
33.
    - for the batch submission withdataRootCommitment
34.
    - fromVectorX (opens in a new tab)
35.
    - .
36. Replay batch execution over WASM binary STF
37.
    - Component
38.
    - :Arbitrator, ArbState
39.
    - Process: Recover payload from Avail DA (RecoverPayloadFromAvailBatch (opens in a new tab)
40.
    -),
41.
    - and re-executes the State Transition Function against input messages to determine
42.
    - the correct output block.

Arbitrum Nitro Avail-powered Orbit chains