Blobstream: Streaming modular DA to Ethereum

What is Blobstream?

Blobstream is the first data availability solution for Ethereum that securely scales with the number of users. Formerly known as theQuantum Gravity Bridge (QGB) , Blobstream relays commitments to Celestia's data root to an onchain light client on Ethereum, for integration by developers into L2 contracts. This enables Ethereum developers to build high-throughput L2s using Celestia's optimised DA layer, the first with Data Availability Sampling (DAS).

An implementation of Blobstream, bySuccinct , calledBlobstream X , is out and will be used in the upcoming deployments. This implementation proves the validity of Celestia block headers on a target EVM chain using zero-knowledge (ZK) proofs, which allow inheriting all the security guarantees of Celestia.

Blobstream vs. data availability committees (DACs)

Decentralization and security

Blobstream is built on Celestia, which uses a CometBFT-based proof-of-stake system. Blobstream shares the same security assumptions as Celestia. In contrast, data availability committees (DACs), are typically centralized or semi-centralized, relying on a specific set of entities or individuals to vouch for data availability.

Mechanism of verification

Blobstream uses data availability attestations, which are Merkle roots of the batched L2 data, to confirm that the necessary data is present on Celestia. The L2 contract on Ethereum can check directly with Blobstream if the data is published on Celestia. Similarly, a DAC would rely on attestations or confirmations from its permissioned members.

Flexibility and scalability

Blobstream is designed to offer high-throughput data availability for Ethereum L2s, aiming to strike a balance between scalability and security. It operates independently of Ethereum's gas costs, as Celestia's resource pricing is more byte-focused rather than computation-centric. On the other hand, the scalability and flexibility of a DAC would depend on its specific design and implementation.

In summary, both Blobstream and DACs aim to ensure offchain data availability, but Blobstream offers a more decentralized, secure, and scalable solution compared to the potential centralized nature of DACs.

What is Blobstream X?

Blobstream X is an implementation of Blobstream with a ZK light client that bridges Celestia’s modular DA layer to Ethereum to allow high-throughput rollups to use Celestia’s DA while settling on Ethereum.

Optimistic or ZK rollups that settle on Ethereum, but wish to use Celestia for DA, require a mechanism forbridging Celestia’s data root to Ethereum as part of the settlement process. This data root is used during inclusion proofs to prove that particular rollup transactions were included and made available in the Celestia network.

Bridging Celestia’s data root to Ethereum requires running a Celestialight client as a smart contract on Ethereum, to make the latest state of the Celestia chain known on Ethereum and available to rollups. Blobstream X utilizes the latest advances in ZK proofs to generate asuccinct proof that enough Celestia validators have come to consensus (according to the CometBFT consensus protocol) on a block header, and verifies this proof in the Blobstream X Ethereum smart contract to update it with the latest Celestia header.

The Blobstream X ZK proof not only verifies the consensus of Celestia validators, but it also merkelizes and hashes all the data roots in the block range from the previous update to the current update, making accessible all Celestia data roots (verifiable with a Merkle inclusion proof against the stored Merkle root) to rollups.

Blobstream X is built and deployed withSuccinct's protocol .

Integrate with Blobstream X

The following docs go over how developers can integrate Blobstream X.

You canfind the repository for Blobstream X along with code for:

- The Blobstream X smart contract -BlobstreamX.sol
- The Blobstream X circuits

- The Blobstream X contract Golang bindings

The first deployments of Blobstream X will be maintained on the following chains: Arbitrum One, Base and Ethereum Mainnet. Every 1 hour, the prover/relayer will post an update to the Blobstream X contract that will include a new data commitment range that covers a 1-hour block range from thelatestBlock in the Blobstream X contract. On Ethereum Mainnet, the Blobstream X contract will be updated every 4 hours.

NOTE

Custom ranges can be requested using theBlobstreamX contract to create proofs for specific Celestia block batches. These ranges can be constructed as[latestBlock, customTargetBlock) , withlatestBlock is the latest block height that was committed to by theBlobstreamX contract, andlatestBlock > customTargetBlock , andcustomTargetBlock - latestBlock <= DATA_COMMITMENT_MAX .

Block ranges that are before the contract'slatestBlock can't be proven a second time in different batches.

More information can be found in therequestHeaderRange(...) method.

How Blobstream X works

As shown in the diagram below, the entrypoint for updates to the Blobstream X contract is through theSuccinctGateway smart contract, which is a simple entrypoint contract that verifies proofs (against a deployed onchain verifier for the Blobstream X circuit) and then calls theBlobstreamX.sol contract to update it.Find more information about theSuccinctGateway .

NOTE

If the Blobstream X contract is not deployed on a desired chain, it needs to be deployed before it can be used by your rollup. See thedeployment documentation for more details.

How to integrate with Blobstream X

Integrating your L2 with Blobstream X requires two components: youronchain smart contract logic , and youroffchain client logic for your rollup . The next three sections cover these topics:

- Integrate with Blobstream contracts
- Integrate with Blobstream client
- Querying the Blobstream proofs

Deployed contracts

You can interact with the Blobstream X contracts today on testnet. The Blobstream X Solidity smart contracts are currently deployed on the following Ethereum testnets:

Contract EVM network Contract address Attested data on Celestia Blobstream X Ethereum MainnetNot yet deployed Mainnet Beta Blobstream X Arbitrum One Not yet deployed Mainnet Beta Blobstream X Base Not yet deployed Mainnet Beta Blobstream X Ethereum Sepolia 0x48B257EC1610d04191cC2c528d0c940AdbE1E439 Mainnet Beta Blobstream X Arbitrum Sepolia 0xf6b3239143d33aefc893fa5411cdc056f8080418 Mocha testnet [][Edit this page on GitHub] Last updated: Previous page Prompt Scavenger Next page Integrate with Blobstream contracts []