Substreams Pipeline Migration

Client: Lending Protocol

Migrated a legacy subgraph to a Substreams-powered pipeline, cutting sync times by 100x and enabling real-time analytics across 4 chains.

Substreams　Migration　Multi-Chain　Rust

January 2025

<div>

100x

FASTER SYNC

4

CHAINS DEPLOYED

<2 sec

DATA LAG

</div>

● Challenge

A major lending protocol was running a traditional subgraph that took over 2 weeks to sync from genesis on Ethereum mainnet. With expansion to Polygon, Arbitrum, and Base, the indexing approach couldn't scale.

Each new chain deployment required a separate subgraph with duplicated logic, and the data lag made real-time risk monitoring impossible for the protocol's treasury management team.

● Solution

We rebuilt the data pipeline from scratch using Substreams — a parallelized, streaming-first blockchain data transformation layer built on Firehose.

%, Custom Rust modules for composable data extraction across lending events

%, Shared module architecture deployed identically across all 4 chains

%, Dual sink configuration: PostgreSQL for analytics, Substreams-powered subgraph for GraphQL

%, Built a real-time monitoring dashboard feeding from the ClickHouse analytics layer

● Results

%, 100x faster sync: from 2 weeks down to approximately 3 hours on Ethereum mainnet

%, Deployed across 4 chains with a single shared Rust module architecture

%, Sub-2-second data lag from on-chain event to queryable state

%, Enabled real-time risk analytics previously impossible with the legacy approach