Your Docusaurus site did not load properly.

A very common reason is a wrong site **baseUrl configuration**.

Current configured baseUrl = / (default value)

We suggest trying baseUrl = **/D:/Desktop/whitepapers/whitepapers_html/whitepapers/12402_POWER_Powerloom.html/**

Skip to main content

⚠ Powerloom wind-down is complete. Read the wind-down timeline for details.

- Introduction
- Protocol Overview
- DSV Mainnet
- BDS Data Market
- Agents & BDS
- Build with Powerloom
- Chain Migration
- Participate
- Wind-Down

CRYP2NOVA

Archived copy · cryp2nova.com

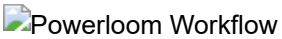
-
- Introduction

On this page

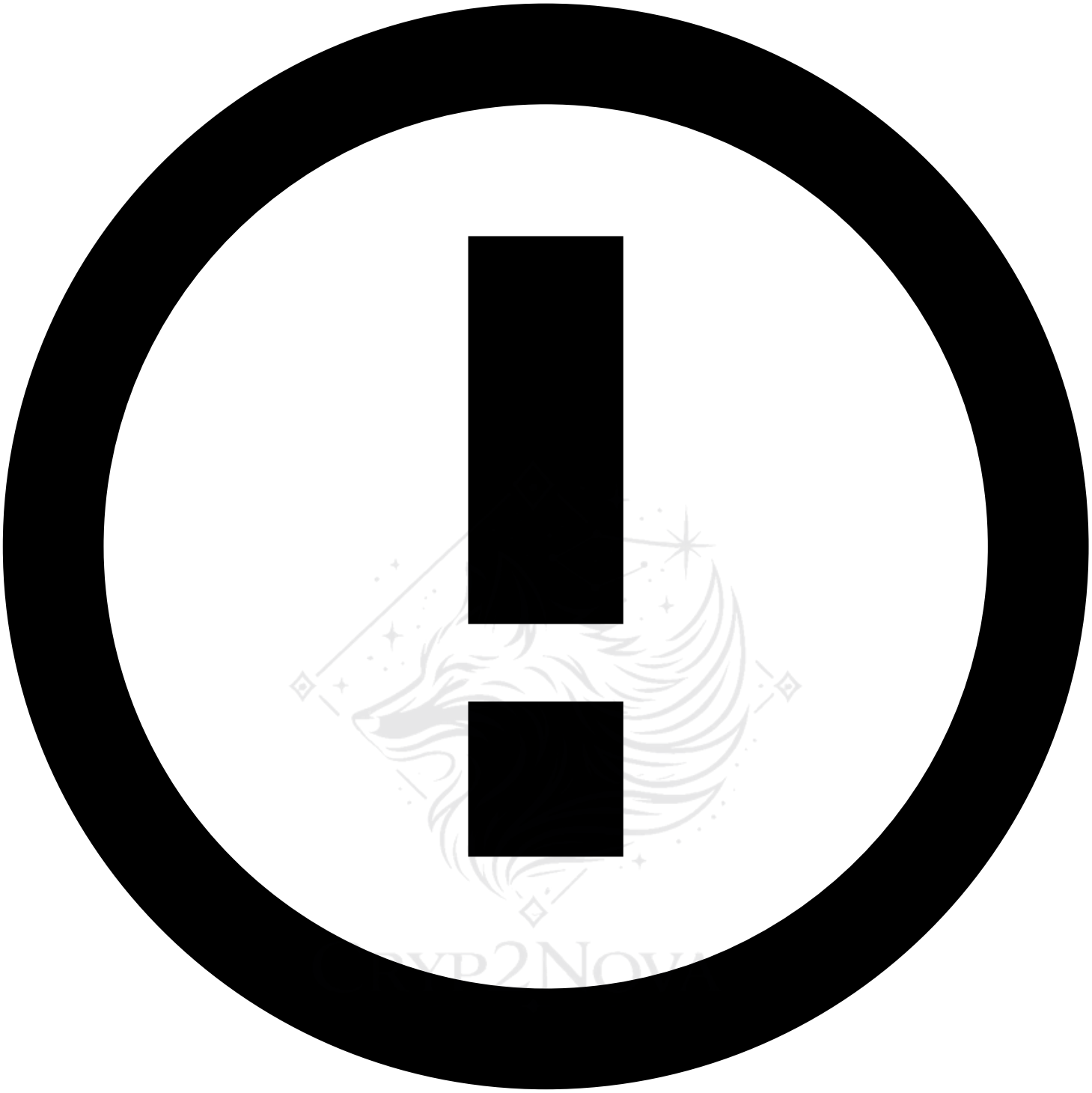
Introduction

Powerloom Protocol is a decentralized data protocol for applications that need reliable blockchain data without operating their own indexing and verification stack. It coordinates snapshotter nodes, validators, data markets, and consumers around peer-validated observations of smart contract state transitions and events.

By composing smaller consensus-reached data units into queryable datasets, Powerloom gives dashboards, bots, aggregators, agents, and analytics systems a verifiable source of blockchain data.



Archived copy · cryp2nova.com



info

Current mainnet read path:

1. **DSV Mainnet** — the decentralized sequencer-validator network that finalizes data, writes CIDs on-chain, and adds economic accountability to the finalization layer.
2. **BDS Data Market** — the first production data market on DSV, serving Uniswap V3 data from Ethereum mainnet through a snapshotter full-node resolver.
3. **Agents & BDS** — agent access through metered routes, API keys, the OpenClaw/ClawHub skill, hosted MCP, and the headless `bds-agent` CLI (`pip install bds-agent` or `uv tool install bds-agent`).

Archived copy · cryp2nova.com



tip

Powerloom in a nutshell: Powerloom turns raw blockchain activity into decentralized, consensus-backed data products.

Archived copy · cryp2nova.com

Snapshotter nodes observe contracts and produce snapshots. Validators finalize canonical outputs. Consumers and agents use the resulting data through APIs, tooling, and market-specific products.

What Powerloom Provides

Decentralized data: Applications can build on peer-validated, consensus-backed datasets rather than relying only on a single hosted indexer.

Real-time data products: Snapshotter networks can track contract state and event activity as markets evolve, enabling data feeds for dashboards, bots, alerts, and analytics.

Time-series queries: Datasets are captured across epochs, making them useful for historical analysis, monitoring, and trend detection.

Flexible data markets: Curators can define new data products around protocol-specific requirements, while the network allocates participants and incentives around those markets.

Modular architecture: Snapshotting, validation, finalization, data serving, metering, and consumption tooling are separate layers that can evolve independently.

Roles in the Powerloom Ecosystem

The Powerloom ecosystem has several roles that work together to produce, validate, serve, and consume decentralized data.

Snapshotter

Snapshotter nodes observe smart contracts and produce data snapshots for assigned data markets. They capture state transitions, event emissions, and derived outputs required by a market's compute logic.

Validator

Validators participate in the finalization layer. In the current DSV path, validator nodes collect submissions, validate them, deduplicate payloads, aggregate consensus outputs, upload finalized artifacts to IPFS, and anchor canonical references on-chain.

CRYP2NOVA

Curator

Curators define and maintain data markets. They are responsible for market-specific logic, supported datasets, and the usefulness of the resulting data product.

Signaller

Signallers help direct network attention toward valuable data markets. Their role is to influence where snapshotter resources and market participation should concentrate.

Consumer

Consumers are the applications, dashboards, agents, analysts, and services that use Powerloom data. In the current BDS market, consumers can access Uniswap V3 data through metered HTTP routes and agent-oriented tooling.

Watchmen

Archived copy · cryp2nova.com

Watchmen are accuracy-focused participants that can challenge incorrect data. Their role adds another accountability path for data correctness when disputes arise.

Current Production Path: BDS on DSV

The current production path starts with **BDS**, a live Uniswap V3 data market on Ethereum mainnet. Snapshotter nodes produce market snapshots, the decentralized sequencer-validator network finalizes canonical CIDs, and a snapshotter full-node resolver exposes the data through metered HTTP routes.

Supported BDS responses include a `verification` object containing the finalized CID, epoch, project ID, protocol state contract, and data market contract. Agents and applications can verify provenance by checking:

```
ProtocolState.maxSnapshotsCid(dataMarket, projectId, epochId)
```

That means alerts, reports, and automated workflows can carry a proof path back to DSV-finalized state instead of trusting the API operator alone.

Start With the Current Mainnet Guides

- **Consume verified data:** start with Agents & BDS.
- **Understand the live data market:** read What Is BDS?.
- **Review the endpoint surface:** see the Endpoint Catalog.
- **Verify provenance:** use Verification in Agent Workflows.
- **Understand DSV finalization:** begin with Why DSV Exists.
- **Review production signals:** read Stability and Scale.

See Powerloom in Action

The current mainnet access path begins with the BDS Data Market, where DSV-finalized Uniswap V3 data is served through metered HTTP routes with verification metadata.

Powerloom BDS Dashboard

Building With Powerloom

- **Run a BDS snapshotter slot:** follow Snapshotter Lite V2 Setup.
- **Operate resolver/full-node infrastructure:** review Snapshotter Core Edge.
- **Use OpenClaw and hosted MCP:** follow OpenClaw & Hosted MCP.
- **Run a headless agent:** `pip install bds-agent` or `uv tool install bds-agent`, then follow `Headless CLI for BDS`.

CRYP2NOVA

Archived copy · cryp2nova.com

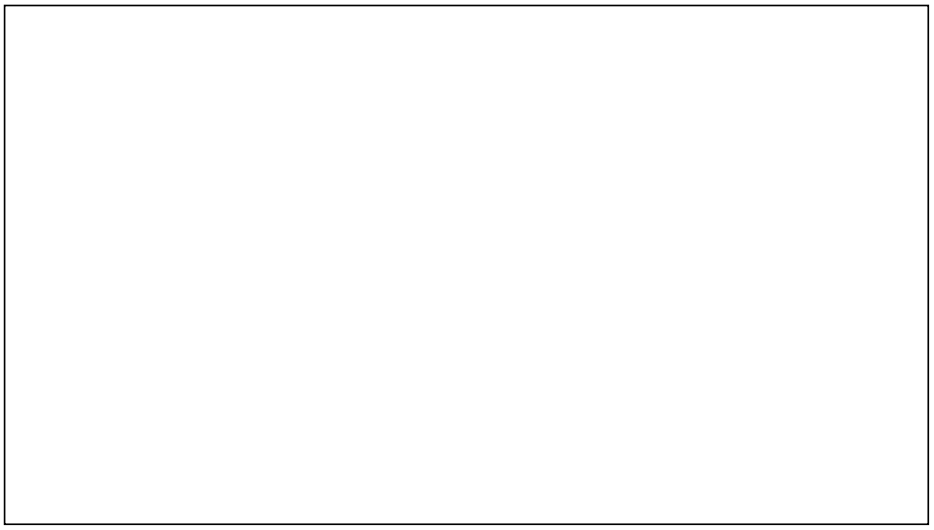
Implementation Repositories

- Powerloom GitHub: github.com/powerloom
- Snapshotter full-node resolver: `powerloom/snapshotter-core-edge`
- Snapshotter lite node: `powerloom/snapshotter-lite-v2`
- DSV validator network: `powerloom/snapshot-sequencer-validator`
- OpenClaw skill: `powerloom/powerloom-bds-univ3`
- Headless agent CLI: `bds-agent` on PyPI · `powerloom/bds-agent-py`

Walkthrough

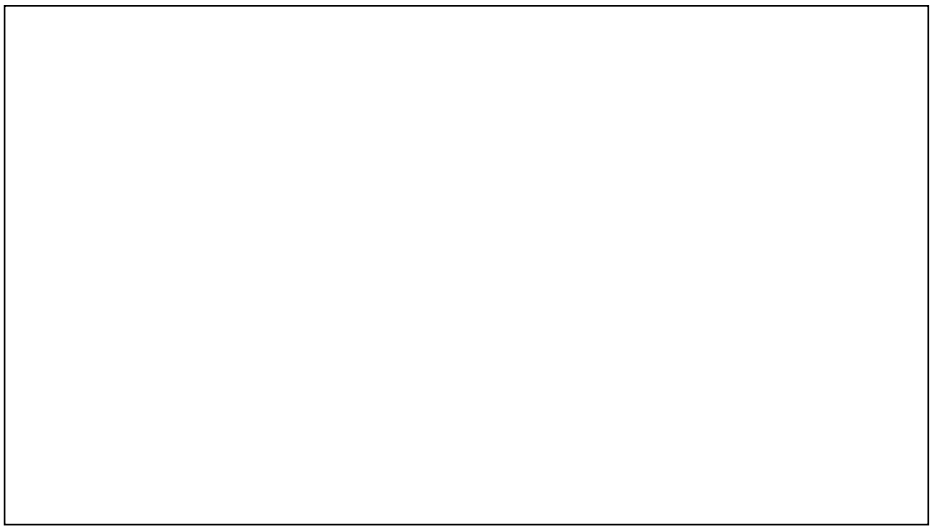
Powerloom has evolved through multiple protocol versions and production paths. These technical community calls provide additional background on the network, how it functions, and how developers can build with Powerloom data.

Technical Community Call #1



Technical Community Call #2

Archived copy · cryp2nova.com



Community

Stay updated with the Powerloom team and community on Discord, X, and GitHub.

Something Missing?

If you find issues with the documentation or have suggestions on how to improve the documentation or the project, please file an issue or email .

Edit this page
Next
Protocol Overview

- What Powerloom Provides
- Roles in the Powerloom Ecosystem
  - Snapshotter
  - Validator
  - Curator
  - Signaller
  - Consumer
  - Watchmen
- Current Production Path: BDS on DSV
- Start With the Current Mainnet Guides
- See Powerloom in Action
- Building With Powerloom
- Implementation Repositories
- Walkthrough
  - Technical Community Call #1
  - Technical Community Call #2
- Community
- Something Missing?

Copyright © 2026 Powerloom Network (BVI) Ltd.

Archived copy · cryp2nova.com