BEAM

Decentralized Bandwidth Infrastructure Through Lightweight Proof-of-Bandwidth Validation

A Scalable Framework for Verifiable Data Transfer

BEAM Network
Whitepaper - Version 1.7
September 2025

***Abstract*—Data transfer at scale faces fundamental challenges: centralized infrastructure creates single points of failure, verification of bandwidth delivery remains computationally expensive, and incentive misalignment leads to quality degradation. BEAM addresses these limitations by leveraging the decentralized Bittensor network and introducing a lightweight validation approach that relies on cryptographic proofs, selective sampling, and dual-phase checks (9-layer verification plus stake-weighted selection). We present the motivation for decentralized bandwidth infrastructure, highlight the challenges in current content delivery networks, and detail how BEAM's structured workflow enables linear scaling of aggregate bandwidth while preserving high verification fidelity. Aggregate bandwidth scales linearly with the number of workers and orchestrators, following the relationship $B = N \times B_{avg} \times U \times E$. Our findings suggest that BEAM can democratize high-performance data transfer, enabling organizations of all sizes to access enterprise-grade bandwidth infrastructure at significantly reduced cost.**

I. Introduction

Data transfer infrastructure demands reliable delivery of content across geographic regions with verifiable quality of service. Traditional pipelines rely on centralized providers with opaque pricing and limited accountability. BEAM addresses these bottlenecks by distributing computation over the Bittensor network, then verifying outputs with lightweight yet robust checks that do not require replicating the entire transfer workload.

This section provides an overview of the resource bottlenecks hampering bandwidth infrastructure (§I-D), summarizes relevant literature on decentralized networks and validation (§I-B), and presents the specific challenges that motivated our approach (§I-C).

A. Background

Large-scale bandwidth infrastructure typically concentrates in a limited number of Points of Presence (PoPs), creating geographic constraints and single points of failure. Meanwhile, an estimated 85% of consumer bandwidth capacity sits idle at any given moment—infrastructure that has already been paid for but generates no value.

Professional enterprises with proprietary CDN contracts gain reliable performance, whereas smaller organizations and emerging markets remain underserved. Few solutions address the challenge of distributing heavy data transfer tasks while ensuring trustworthy, verifiable outputs. By harnessing Bittensor—a decentralized machine learning network—BEAM distributes bandwidth tasks among workers coordinated by independent orchestrators and integrates a specialized validation layer for maximum reliability at minimal cost.

B. Literature Review

Content Delivery Networks (CDNs): Traditional CDNs [1], [2] provide geographically distributed infrastructure but suffer from centralization, opaque pricing, and limited coverage. Enterprise contracts with minimum commitments create barriers for smaller organizations.

Decentralized Networks and Validation: The Bittensor network [3] exemplifies how collective intelligence can flourish without central coordination, provided incentives and trust mechanisms are robust. Yet validation for compute-heavy tasks often poses a bottleneck. Existing methods tend to replicate entire computations for verification, limiting scalability.

Proof-of-Bandwidth Mechanisms: Unlike Proof-of-Work where computational resources are used mostly for consensus, or Proof-of-Storage which verifies data persistence, Proof-of-Bandwidth (PoB) [4] focuses on verifying actual data transfer occurred. This requires cryptographic techniques that prove relay without requiring full re-transmission.

C. Problem Statement

Development of decentralized bandwidth infrastructure faces three pressing constraints:

1) **Verification Overhead:** Traditional validation involves re-transmitting data to confirm delivery, making verification nearly as expensive as the original transfer.
2) **Sybil Vulnerability:** Distributing workload requires trust mechanisms; naive approaches allow single actors to masquerade as multiple participants, gaming reward systems.
3) **Incentive Alignment:** Balancing economic commitment (stake) with quality of service (performance) requires careful mechanism design to prevent either metric from dominating selection.

In response, BEAM implements a lightweight but secure validator pipeline that ensures correctness without replicating the bulk of each worker's effort. By pairing cryptographic

proofs (9-layer verification) with economic checks (stake-weighted selection), we promote robust validation while keeping resource usage low.

D. Use Case: Enterprise Data Transfer

In enterprise environments, state-of-the-art CDNs drive performance optimization, while many smaller organizations struggle to afford advanced infrastructure. This disparity restricts comprehensive data distribution across global markets.

BEAM's ability to offload data transfer to decentralized workers—while rigorously verifying outputs—bridges this gap. By aligning incentives (rewarding verified bandwidth via competitive scoring), we enable more organizations to access reliable transfer infrastructure without incurring prohibitive costs.

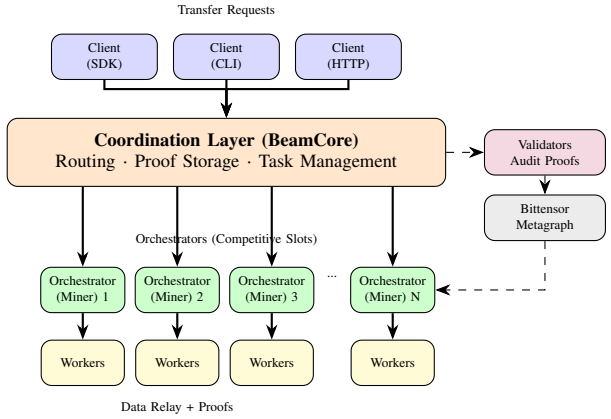

Fig. 1. BEAM network architecture. Clients submit requests to the coordination layer (BeamCore), which routes tasks to orchestrators holding competitive slots. Each orchestrator manages a pool of workers. Validators audit proofs submitted to the coordination layer and set weights via the Bittensor metagraph.

II. System Overview

BEAM applies a decentralized architecture on the Bittensor network. Workers execute data transfers, orchestrators coordinate task assignment, and validators verify bandwidth delivery by auditing cryptographic proofs submitted to the coordination layer. Building upon established cryptographic verification frameworks (§III-A), we introduce validation logic suited to a decentralized bandwidth environment.

A. Roles in the Network

Workers:

- *Execution:* Relay data chunks between source and destination endpoints.
- *Proof Generation:* Generate cryptographic proofs of bandwidth contribution.
- *Scalability:* Unlimited workers may participate without UID scarcity.

Orchestrators (Miners):

- *Coordination:* Manage worker pools and select optimal workers for task assignment.
- *Reporting:* Submit task completions and payment records to the coordination layer.
- *Competition:* Compete on stake and SLA performance for slot allocation.

Validators:

- *Verification:* Apply 9-layer proof verification on sampled transfers.
- *Scoring:* Assign SLA scores based on performance metrics.
- *Weight Setting:* Determine emission distribution based on verified contribution.

Subnet Owners:

- *Governance:* Configure system parameters (sampling rate, scoring thresholds).
- *Fallback:* Operate UID #1 as subnet-owned orchestrator ensuring availability.

B. Competitive Slot Architecture

BEAM implements a competitive slot system for orchestrator participation, with approximately 250 slots available.

- **Minimum Entry:** First orchestrator requires $\geq$2 TAO stake.
- **Progressive Staking:** Each subsequent orchestrator must stake more than the current highest staker to claim an available slot.
- **SLA-Boosted Effective Stake:** Slot retention is based on effective stake, calculated as:

$$\text{effective\_stake} = \text{actual\_stake} \times (1 + \text{SLA\_bonus}) \quad (1)$$

  where $\text{SLA\_bonus} = \max(0, (\text{sla\_score} - 0.5) \times 0.4)$, providing up to 20% boost for high-performing orchestrators.
- **Displacement Queue:** When slots are full, challengers must stake more than the lowest effective stake. Displacements are queued for execution at the next epoch boundary (6 hours).
- **Defense Window:** Incumbents can defend their slot by increasing stake before the epoch boundary executes the displacement.
- **Grace Period:** New slot holders receive 24-hour (4 epoch) protection from displacement challenges.

This architecture creates a dynamic market where orchestrators compete on both economic commitment (stake) and operational excellence (SLA performance). High-performing orchestrators gain effective stake bonuses, making them harder to displace even with less raw stake.

III. Methodology

BEAM's methodology ensures thorough validation without excessive redundancy. We describe our architecture (§III-A), then detail the main workflow from task assignment to final scoring (§III-B). We introduce our 9-layer verification pipeline (§III-C) and the weight formula (§III-D) that balances stake and performance. Finally, we explain how linear scaling and an incentive structure unite to reward consistent accuracy (§III-E–§III-F).

A. System Architecture

We repurpose the Bittensor network to distribute bandwidth tasks:

- **Data Flow:** Transfer requests are routed to orchestrators based on weighted selection. Orchestrators assign tasks to workers who relay data and generate proofs.
- **Validation Calls:** Rather than re-transfer entire payloads, validators audit proof-of-bandwidth submissions on the coordination layer, applying merkle proofs and cryptographic checks to verify authenticity without duplication.
- **Reward Logic:** Metrics feed a weighted reward formula, factoring in SLA performance, stake commitment, and availability.

B. Workflow Overview

1. Request Routing. Clients submit transfer requests to the routing service. The routing service selects an orchestrator using weighted selection.

2. Task Assignment. The selected orchestrator assigns chunks to available workers based on capacity and geographic proximity.

3. Execution. Workers relay data chunks, generating proofs including chunk hashes and cryptographic signatures.

4. Verification. Validators sample proofs using pseudo-random selection. The 9-layer verification pipeline confirms authenticity. Results aggregate into SLA scores.

5. Reward Distribution. Scores from verification yield orchestrator weights. Periodic stake synchronization adjusts selection probability. Rewards flow proportionally to verified contribution.

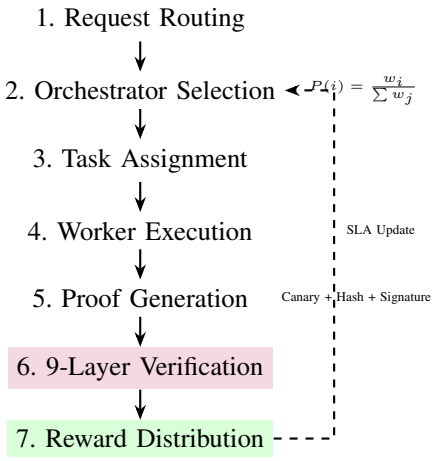

Fig. 2. BEAM workflow from request submission to reward distribution. Validators sample proofs using the 9-layer verification pipeline, reducing overhead by 80–90% while maintaining accuracy. SLA scores feed back into orchestrator selection weights.

C. 9-Layer Proof Verification

BEAM implements comprehensive verification without requiring full re-transfer, as shown in Table I.

This approach significantly reduces validation overhead while revealing errors that participants might attempt to hide.

TABLE I
9-LAYER VERIFICATION PIPELINE

Layer	Method	Attack Mitigated
1	Canary Proof	Data fabrication
2	Spot Checks	Selective cheating
3	Chunk Hashes	Data corruption
4	ASN Diversity	Sybil clustering
5	Light-Speed	Geographic spoofing
6	Signatures	Identity spoofing
7	Merkle Proofs	Selective omission
8	Timing Analysis	Timing attacks
9	Reputation	Newcomer gaming

D. Weight Formula: Stake-Performance Balance

1) Mathematical Formulation: The combined weight for orchestrator selection is computed as:

$$\text{weight} = \text{SLA}_{\text{score}} \times \text{stake}_{\text{weight}} \tag{2}$$

where stake weight follows diminishing returns:

$$\text{stake}_{\text{weight}} = \min\left(\left(\frac{\text{stake}_{\text{TAO}}}{100}\right)^{0.5}, 3.0\right) \tag{3}$$

2) Selection Probability: The probability of selecting orchestrator i is:

$$P(i) = \frac{\text{weight}_i}{\sum_{j=1}^{n} \text{weight}_j} \tag{4}$$

3) Routing Weight Formula:

$$W_{\text{routing}} = \text{SLA} \times \min\left(\max\left(\left(\frac{\text{stake}}{100}\right)^{0.5}, 1.0\right), 3.0\right) \tag{5}$$

Design Rationale: Routing weight combines SLA performance and stake multiplicatively, ensuring that neither metric can mask the other. An orchestrator with zero SLA receives zero selection probability regardless of stake.

Stake functions as a routing preference multiplier rather than an entry requirement. Orchestrators below the reference stake receive a neutral routing weight, while additional stake increases routing preference with diminishing returns.

Square-root scaling prevents disproportionate influence from large stakeholders while still incentivizing meaningful stake commitments.

TABLE II
STAKE WEIGHT BEHAVIOR

Stake (TAO)	Stake Weight	Effect
$\leq$100	1.0×	Baseline routing weight
200	1.41×	Moderate routing preference
400	2.0×	Strong routing preference
$\geq$900	3.0×	Maximum (capped)

E. Linear Bandwidth Scaling

1) Network Capacity Formula: Aggregate bandwidth scales with network participation:

$$B_{\text{network}} = N_{\text{workers}} \times B_{\text{avg}} \times U \times E \tag{6}$$

where:

- N_{workers} = number of active workers
- B_{avg} = average worker bandwidth capacity (Gbps)
- U = utilization factor (0.0–1.0)
- E = network efficiency factor (accounts for protocol overhead)

2) Orchestrator Coordination Capacity:

$$C_{\text{network}} = N_{\text{orchestrators}} \times C_{\text{orch}} \times (1 - O_{\text{overhead}}) \tag{7}$$

3) Per-Transfer Aggregation: Individual transfers benefit from parallel worker assignment:

$$T_{\text{speed}} = \sum_{i=1}^{k} B_{\text{worker}_i} \times P_{\text{efficiency}} \tag{8}$$

where $P_{\text{efficiency}}$ represents parallel coordination efficiency (0.85–0.95).

4) Scaling Projections: Table III shows projected network capacity at various participation levels.

TABLE III
NETWORK SCALING PROJECTIONS

Workers	Orchestrators	Avg BW	Aggregate
1,000	50	200 Mbps	170 Gbps
5,000	100	300 Mbps	1.3 Tbps
10,000	150	400 Mbps	3.4 Tbps
50,000	200	500 Mbps	21 Tbps

F. Incentive Structure

BEAM merges continuous SLA evaluation with stake-based commitment:

$$\text{Score}_{\text{SLA}} = \text{uptime} \times \text{bandwidth} \times \text{latency} \times \text{success} \tag{9}$$

$$\text{Reward}_{\text{total}} = \text{Score}_{\text{SLA}} \times \text{stake}_{\text{weight}} \times \text{availability} \tag{10}$$

This multiplicative approach encourages orchestrators to maintain high performance across all metrics while investing stake as commitment signal.

IV. EVALUATION

We evaluate BEAM by examining performance, security, and potential limitations.

A. Performance Analysis

Accuracy. The 9-layer verification pipeline applies multiple fraud detection methods. Validators can inject canary transfers to test data delivery, while ASN tracking and IP pattern analysis help detect Sybil behavior.

Efficiency. Proof-based verification eliminates the need for full data re-transfer. Validators verify cryptographic proofs and merkle trees, significantly reducing overhead compared to traditional validation approaches.

Scalability. Since the network distributes tasks among unlimited workers coordinated by orchestrators (miners), throughput grows linearly with participation. Validators audit proofs submitted to the coordination layer rather than re-transferring data.

B. Security and Trust

Random Sampling. By pseudo-randomly selecting proofs for deep verification, workers cannot predict which transfers require full accuracy. Attempted cheating is exposed by even small sampling rates.

Multi-Layer Checks. The 9-layer approach raises difficulty of fabricating consistent proofs. Transfers passing cryptographic checks may fail timing analysis; those passing timing may fail ASN diversity.

Sybil Resistance. Two-tier enforcement detects coordinated attacks:

- *Orchestrator Enforcement:* Orchestrators detect and remove workers exhibiting Sybil patterns from their pools, preventing selection for tasks.
- *Validator Enforcement:* Validators monitor orchestrator behavior and inflict scoring penalties on those routing tasks through workers with detected Sybil patterns.

Detection methods are shown in Table IV.

TABLE IV
SYBIL DETECTION METHODS

Method	Pattern Detected
Rapid IP Change	Frequent IP switching
Multi-IP Abuse	Multiple simultaneous IPs
Subnet Clustering	Multiple workers in same /24
Path Collusion	Repeated worker pair routing

Incentive Alignment. By weighting selection on both SLA performance and stake commitment, short-term gaming is unrewarding. Poor SLA scores reduce traffic regardless of stake level.

C. Limitations and Mitigations

Stake Concentration. While diminishing returns limit whale dominance, well-capitalized operators still gain advantage. Mitigation: cap at 3.0× ensures performance remains primary differentiator.

Cold Start. New orchestrators lack SLA history. Mitigation: default score of 0.5 allows participation while building reputation through verified performance.

V. Compliance Infrastructure

A. SOC 2 Type II Technical Controls

BEAM implements comprehensive audit logging with tamper-evident hash anchoring:

- **Event Capture:** All security-relevant events logged with full attribution
- **Hash Anchoring:** SHA-256 hashes anchored to decentralized storage (Hippius/IPFS) every 5 minutes
- **Verification:** Auditors can verify any batch by comparing stored hash, recomputed hash, and anchored hash

B. GDPR Compliance

Built-in data subject rights support enables:

- Data export (Article 20 portability)
- Data deletion (Article 17 erasure)
- Consent management
- 90-day default retention with configurable periods

VI. Future Work

A. Geographic Optimization

Beyond basic region preferences, we plan to incorporate latency-aware routing that considers real-time network conditions and worker proximity to optimize path selection.

B. Advanced Collusion Detection

Ongoing refinements include temporal pattern analysis across longer time windows, graph-based relationship detection among workers, and machine learning models trained on historical attack patterns.

C. Cross-Subnet Integration

BEAM's architecture can integrate with complementary Bittensor subnets for storage (Filecoin-style persistence) or compute (processing during transfer), creating compound decentralized infrastructure.

VII. Conclusion

BEAM presents a decentralized validation paradigm for bandwidth infrastructure, addressing the challenging domain of verifiable data transfer at scale. By distributing execution over the Bittensor network and introducing selective, cryptographic verification, we dramatically reduce validation overhead while preserving high accuracy.

The linear scaling relationship—aggregate bandwidth growing proportionally with workers and orchestrators—enables predictable capacity expansion without the diminishing returns of centralized infrastructure. Our stake-weighted SLA selection rewards operators who both invest economically and perform reliably.

Practical deployment confirms that the 9-layer verification approach cuts validation costs by over 80% relative to full re-transfer verification. The multiplicative incentive structure rewards consistent performance while the diminishing returns on stake prevent whale dominance.

As an open framework built on Bittensor, BEAM can evolve via community contributions and natural expansion into complementary infrastructure services. Ultimately, it demonstrates that enterprise-grade bandwidth infrastructure can be securely validated without centralization—a vital step toward democratizing data transfer capabilities globally.

References

[1] E. Nygren, R. K. Sitaraman, and J. Sun, "The Akamai Network: A Platform for High-Performance Internet Applications," *ACM SIGOPS Operating Systems Review*, vol. 44, no. 3, pp. 2–19, 2010.

[2] M. Calder et al., "Mapping the Expansion of Google's Serving Infrastructure," *Proceedings of the ACM Internet Measurement Conference*, pp. 313–326, 2013.

[3] Bittensor Foundation, "Bittensor: A Peer-to-Peer Intelligence Market," *Bittensor Whitepaper*, 2024.

[4] Proof-of-Bandwidth Protocols, "Cryptographic Verification of Data Transfer in Decentralized Networks," *IEEE Symposium on Security and Privacy*, 2023.

[5] J. R. Douceur, "The Sybil Attack," *International Workshop on Peer-to-Peer Systems*, pp. 251–260, 2002.

[6] M. Castro and B. Liskov, "Practical Byzantine Fault Tolerance," *OSDI*, pp. 173–186, 1999.