ARIA

A Peer-to-Peer Efficient AI Inference Protocol

Autonomous Responsible Intelligence Architecture

Version 2.0

Anthony MURGO

anthony.murgo@outlook.com

February 2026

github.com/spmfrance-cloud/aria-protocol

Abstract

ARIA (Autonomous Responsible Intelligence Architecture) is an open protocol for distributed AI inference on consumer CPUs. By combining 1-bit quantized large language models (LLMs) with a peer-to-peer network governed by explicit consent contracts, ARIA enables low-cost, energy-efficient, and privacy-preserving AI inference without specialized hardware.

The protocol introduces a reputation-based contribution system where nodes earn quality scores through useful work, replacing traditional token-based incentives. A provenance ledger provides cryptographic verification of all inference operations. Multi-architecture validation across AMD and Intel CPUs confirms that ARIA achieves 36-120 tokens/second on consumer hardware while consuming approximately 11-66 mJ per token, representing a 99%+ energy reduction compared to datacenter GPU inference.

ARIA aims to democratize AI by turning billions of idle CPUs worldwide into a distributed intelligence network, where participation is governed by consent, quality is ensured by reputation, and provenance is guaranteed by cryptography.

1. Introduction

Artificial intelligence has become the defining technology of this decade. Large language models (LLMs) demonstrate remarkable capabilities across text generation, reasoning, code synthesis, and multimodal understanding. Yet this power is concentrated: a small number of companies control AI infrastructure through expensive GPU clusters, creating dependency, surveillance risk, and exclusion for billions of users.

Meanwhile, there are an estimated 2-3 billion personal computers worldwide, the vast majority sitting idle for 90%+ of their operational time. These consumer CPUs represent an enormous untapped computational resource. The breakthrough of 1-bit quantization (ternary weights: {-1, 0, +1}) makes it possible for the first time to run meaningful LLMs on standard CPUs with no GPU requirement.

ARIA Protocol combines three key innovations:

1. CPU-native 1-bit inference using ternary lookup tables, eliminating floating-point multiplication entirely.
2. Peer-to-peer networking with explicit consent contracts, where every node declares exactly what it is willing to contribute.
3. Provenance tracking and reputation scoring, providing cryptographic verification of all inference operations and quality-based routing.

Multi-architecture validation (v0.5.5) demonstrates that this approach is hardware-agnostic: ARIA runs efficiently on both AMD Zen 4 and Intel Tiger Lake architectures, with performance characteristics that differ by ISA implementation rather than raw core count.

2. The Problem

Current AI inference infrastructure suffers from three fundamental problems: centralization, cost, and opacity.

2.1 Centralization

Over 95% of AI inference runs through a handful of providers (OpenAI, Google, Anthropic, Meta). Users send sensitive prompts to remote servers with no control over data handling, model behavior, or availability. Service outages, policy changes, or censorship decisions affect millions of users simultaneously.

2.2 Cost

GPU-based inference is expensive. A single NVIDIA H100 costs ~$30,000 and consumes 700W. Cloud API pricing ranges from $0.90 to $60 per million tokens. For organizations processing millions of tokens daily, this represents a significant operational cost.

2.3 Existing Approaches

Project	Hardware	Incentive	Consent	Privacy	Energy Tracking
ARIA	CPU (1-bit)	Reputation	Granular	Local-first	Per-inference
Bittensor v2	GPU	TAO tokens	None	Cloud	None
Gensyn	GPU	Tokens	None	Cloud	None
Render	GPU	RNDR tokens	None	Cloud	None
Petals	GPU/CPU	None	None	Partial	None

ARIA is unique in combining CPU-first execution, consent-based governance, and per-inference energy tracking. The Falcon-Edge ecosystem (TII, 2024) validates the 1-bit approach for edge deployment, while ARIA extends it to a distributed P2P network.

3. Protocol Architecture

ARIA is organized into five distinct layers, each responsible for a specific aspect of the distributed inference pipeline.

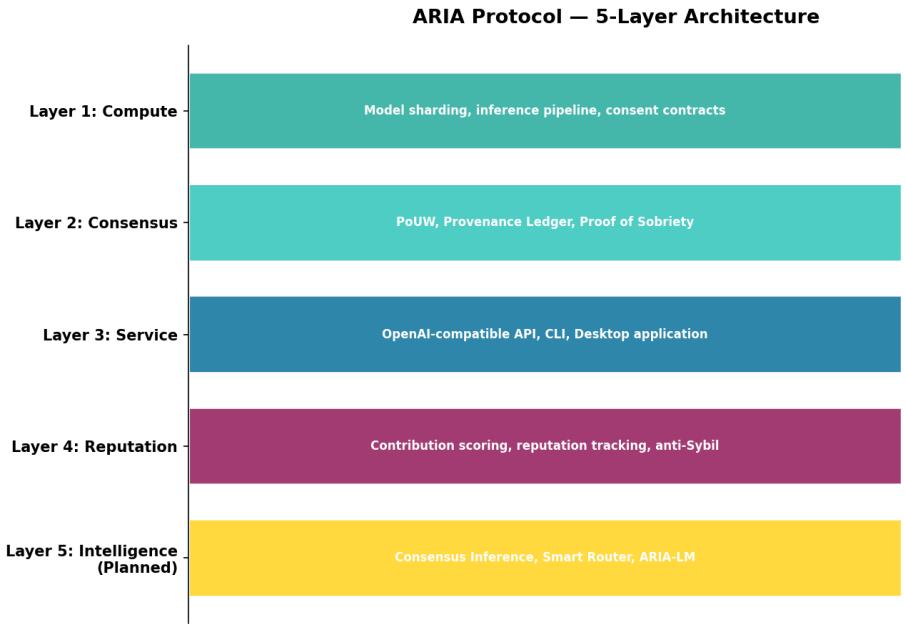

Figure 1: ARIA 5-layer architecture

3.1 Layer 1 - Compute

The compute layer handles model sharding, inference execution, and consent enforcement. Models are split into shards distributed across nodes. Each node declares its capabilities through a consent contract specifying CPU allocation, RAM limits, schedule availability, accepted task types, and contribution score thresholds.

Consent contracts are the ethical backbone of the protocol. No work is assigned to a node unless it explicitly matches the node's declared consent parameters. This ensures that every participant has full control over their contribution.

3.2 Layer 2 - Consensus

Proof of Useful Work (PoUW): Every inference generates a cryptographic proof binding the input hash, output hash, computation time, energy consumed, and node identity. Unlike blockchain proof-of-work, the computation is the useful inference itself, not a waste puzzle.

Provenance Ledger: An append-only chain of blocks records all inference operations. Each record contains the inference hash, node ID, model ID, timestamp, and proof. Users can independently verify that their query was processed correctly.

Proof of Sobriety: Nodes report energy consumption per inference. The protocol tracks energy efficiency ratings (A+ through F) and provides network-wide savings estimates compared to datacenter baselines.

3.3 Layer 3 - Service

ARIA exposes an OpenAI-compatible REST API, enabling zero-code integration with existing applications. A command-line interface provides developer tools for node management, benchmarking, and network

monitoring. ARIA Desktop, built with Tauri 2.0 and React, provides a consumer-friendly GUI with 12-language support, allowing non-developers to contribute to the network in under 60 seconds.

3.4 Layer 4 - Reputation

The reputation layer replaces traditional token-based incentives with a quality-focused contribution scoring system. Nodes earn contribution scores based on useful work:

```
Score(n) = base_rate x inferences_completed x quality_score x efficiency_bonus
```

Where quality_score = f(uptime, latency, verification_pass_rate) in [0, 1] and efficiency_bonus = g(energy_per_inference / network_average) in [0.5, 2.0]. Nodes that consume less energy per inference earn up to 2x bonus, directly incentivizing energy efficiency.

Contribution scores are NOT transferable, NOT tradeable, and have NO monetary value. They serve exclusively as a quality metric for network routing and task assignment. This eliminates regulatory complexity and aligns incentives purely with network health.

3.5 Layer 5 - Intelligence (Planned)

Consensus Inference: A multi-agent debate protocol where multiple nodes independently process the same query, then reach consensus through structured argumentation. Research from Nature (2025) and the SLM-MATRIX framework validates this approach, achieving 92.85% accuracy with 7B models through multi-agent debate.

Smart Router: Confidence-based routing (inspired by SLM-MUX) that directs queries to the most appropriate model/node combination based on task complexity, required quality, and node capabilities.

ARIA-LM: A community-fine-tuned model evolved through LoRA adapters and SAPO (Self-play Alignment with Principle Optimization), allowing the network to continuously improve its own model through decentralized training.

Knowledge Network: Distributed retrieval-augmented generation (RAG) via Kademlia DHT, enabling nodes to share and query a collective knowledge base.

4. CPU-Native 1-Bit Inference

The fundamental insight enabling ARIA is that 1-bit (ternary) quantization eliminates floating-point multiplication entirely. In a standard neural network, the most expensive operation is matrix multiplication: Y = W x X, where W contains billions of floating-point weights. With ternary weights ({-1, 0, +1}), this becomes pure addition and subtraction, implementable as lookup tables (LUTs) that execute efficiently on standard CPU instruction sets.

Microsoft's BitNet b1.58 architecture demonstrates that 1-bit models achieve competitive quality with standard FP16 models while requiring 10-20x less memory and dramatically less energy. ARIA leverages the bitnet.cpp inference engine which compiles optimized kernels targeting AVX-512, AVX2, and ARM NEON instruction sets.

4.1 Performance Results

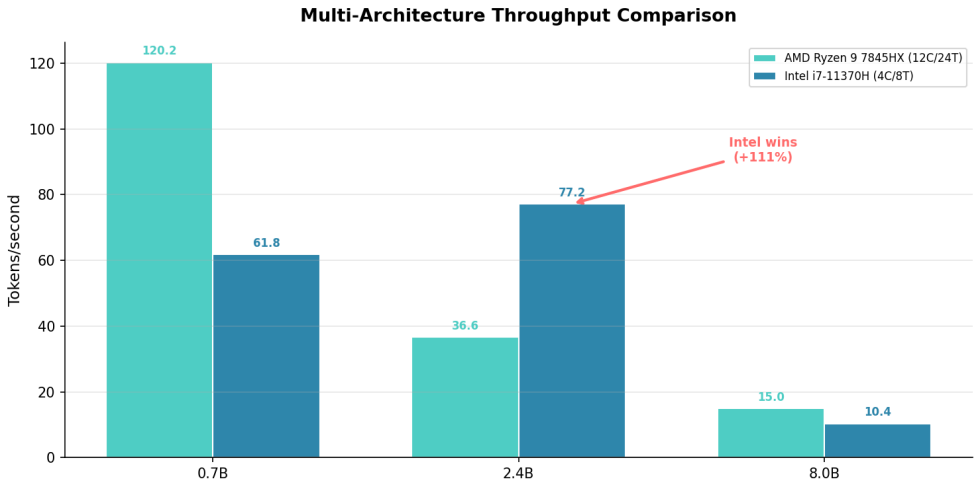

Figure 2: Multi-architecture throughput comparison

Metric	AMD Ryzen 9 7845HX	Intel Core i7-11370H
Architecture	Zen 4 (12C/24T)	Tiger Lake (4C/8T)
0.7B throughput	120.25 t/s	61.81 t/s

2.4B throughput	36.62 t/s	77.21 t/s
8.0B throughput	~15.03 t/s	10.36 t/s
Memory (2B)	~0.4 GB	~0.4 GB
Energy (2.4B)	~28 mJ/token	~28 mJ/token

Multi-architecture validation reveals that 1-bit inference performance is ISA-sensitive. Intel Tiger Lake with native 512-bit AVX-512 execution units outperforms AMD Zen 4 (double-pumped 2x256-bit) on the 2.4B model by 111%. This validates ARIA's hardware-agnostic design: the protocol adapts to the strengths of each architecture.

5. Peer-to-Peer Network Design

5.1 Node Lifecycle

Phase	Actions	Outcome
Join	Generate key pair, download shards, publish consent, build initial reputation	Node visible on network
Contribute	Receive tasks, process inference, submit proofs, earn contribution score	Active network participant
Grow	Accumulate reputation, receive higher-priority routing	Trusted high-value node
Leave	Graceful disconnect, shards redistributed, reputation preserved	Can return with history

5.2 Fault Tolerance

ARIA handles node failures through shard replication and pipeline fallback. Each model shard is held by multiple nodes. When a pipeline stage times out (default: 5 seconds), the network automatically routes to a replica node. Dead peers are detected via heartbeat (30-second interval) and pruned from the routing table.

5.3 Security

ARIA implements a defense-in-depth security model with five layers: transport security (TLS 1.3), protocol security (message authentication, replay protection), consensus security (PoUW, PoSobriety), reputation security (reputation-based registration with contribution history, reputation penalties for fraud), and privacy (consent contracts, data minimization). A comprehensive threat model documents nine attack vectors with current and planned mitigations.

6. Provenance and Verification

6.1 On-Chain Records

Every inference operation is recorded in the provenance ledger as an InferenceRecord containing: node_id, model_id, input_hash (SHA-256), output_hash, tokens_generated, latency_ms, energy_mj, and a timestamp. Records are grouped into blocks with Merkle-style chaining.

6.2 Protocol Contracts

Contract	Purpose
ConsentRegistry	Stores and validates node consent descriptors
InferenceMarket	Matches inference requests with available nodes
ProvenanceLedger	Maintains the immutable inference history chain

ContributionTracker	Calculates and distributes contribution scores based on useful work metrics

7. Reputation and Contribution System

ARIA's contribution system is designed to be simple, fair, and non-financial. Nodes earn contribution points for useful work. The scoring formula balances quantity, quality, and efficiency:

```
Score(n) = base_rate x inferences_completed x quality_score x efficiency_bonus
```

Where:
- quality_score = f(uptime, latency, verification_pass_rate) in [0, 1]
- efficiency_bonus = g(energy_per_inference / network_average) in [0.5, 2.0]

Nodes that consume less energy per inference earn up to 2x bonus, directly incentivizing energy efficiency and rewarding efficient CPU architectures.

Unlike token-based systems, contribution scores are NOT transferable, NOT tradeable, and have NO monetary value. They serve exclusively as a quality metric for network routing and task assignment. This eliminates regulatory complexity and aligns incentives purely with network health.

Reputation Properties

Property	Description
Slow to build	Consistent quality work over time
Fast to lose	A single fraud incident has lasting consequences
Non-transferable	Cannot be bought, sold, or delegated
Temporal	Decays if node is inactive (encourages sustained contribution)

Anti-Sybil protection: creating a new node means starting with zero reputation. High-value tasks require minimum reputation thresholds, making Sybil attacks economically impractical without token deposits. The cost of building reputation through legitimate contribution creates a natural barrier against identity farming.

8. Reference Implementation

The reference implementation is open-source (MIT License) and comprises approximately 2,800 lines of Python across 11 modules, with 196 tests passing. The codebase is designed for readability and extensibility.

Module	Lines	Purpose
consent.py	~150	Consent contracts and matching
network.py	~1,250	P2P WebSocket networking with TLS
node.py	~250	High-level node orchestration
proof.py	~350	PoUW and Proof of Sobriety
ledger.py	~300	Provenance ledger chain
inference.py	~200	BitNet inference engine bindings
api_server.py	~150	OpenAI-compatible REST API

cli.py	~150	Command-line interface

8.1 Desktop Application

ARIA Desktop provides a consumer-friendly interface built with Tauri 2.0 and React. It supports 12 languages and allows one-click node contribution. Design principle: a non-developer should be able to become a network contributor in under 60 seconds. The desktop application includes a model manager, system tray integration, real-time inference statistics, and automatic update support.

9. Future Work

Version	Feature	Description
v0.6.0	Testnet Alpha	Kademlia DHT, NAT traversal, bootstrap nodes
v0.7.0	Smart Layer	Consensus Inference, Smart Router, reputation system
v0.8.0	Extended Context	KV-Cache NVMe paging (500K+ tokens on 8GB)
v0.8.0	Mobile Inference	iOS/Android companion app (1B-3B models)
v0.9.0	ARIA-LM	Community LoRA fine-tune via SAPO
v0.9.0	Knowledge Network	Distributed RAG via Kademlia DHT
v1.0.0	Mainnet	Production-ready, third-party audited

Additional research directions:

- Post-training 1-bit quantization (ternarize existing models)

- Hardware optimization: RISC-V, NPU, DSP targets

- Zero-knowledge proofs for private inference

- Mixture-of-Experts + 1-bit: 100B+ parameters in ~1 GB memory

10. Conclusion

Just as Linux decentralized operating systems and BitTorrent decentralized file sharing, ARIA proposes to decentralize AI inference itself. The convergence of 1-bit quantization, peer-to-peer networking, and consent-based governance creates an opportunity to transform billions of idle CPUs into a global intelligence network.

Our benchmarks demonstrate that this is not theoretical: consumer CPUs today achieve 36-120 tokens per second on 1-bit models, with energy consumption 99% lower than datacenter alternatives. Multi-architecture validation confirms hardware-agnostic operation across AMD and Intel platforms.

ARIA's contribution system, based on reputation rather than financial tokens, eliminates speculative dynamics and aligns participation incentives with network quality. The protocol's consent framework ensures that every contributor maintains full agency over their resources.

The reference implementation is open-source, fully tested, and includes a desktop application for non-technical users. ARIA is ready for community contribution and testnet deployment.

References

[1] S. Ma et al. "The Era of 1-bit LLMs: All Large Language Models are in 1.58 Bits." arXiv:2402.17764, 2024.

[2] S. Ma et al. "BitNet: Scaling 1-bit Transformers for Large Language Models." arXiv:2310.11453, 2023.

[3] Microsoft Research. "bitnet.cpp: Fast and Lossless Inference of 1.58-bit LLMs on CPUs." GitHub, 2024.

[4] Y. Wang et al. "Mixture-of-Experts Meets 1-Bit Quantization." arXiv, 2024.

[5] P. Maymounkov, D. Mazieres. "Kademlia: A Peer-to-peer Information System Based on the XOR Metric." IPTPS, 2002.

[6] S. Nakamoto. "Bitcoin: A Peer-to-Peer Electronic Cash System." 2008.

[7] A. Tang et al. "SAPO: Self-play Alignment with Principle Optimization." arXiv, 2024.

[8] Bittensor. "Decentralized Machine Intelligence Network." bittensor.com, 2024.

[9] Gensyn. "Decentralized GPU Training Network." gensyn.ai, 2024.

[10] Petals. "Collaborative Inference of Large Language Models." petals.dev, 2023.

[11] TII. "Falcon3 Family of Open Models." arXiv, 2024.

[12] SLM-MATRIX. "Achieving GPT-4 Level Performance with 7B Models." Nature npj, 2025.

[13] SLM-MUX. "Confidence-based Routing for Multi-agent Systems." arXiv:2510.05077, 2025.

Document Information

Version: 2.0 | Date: February 2026 | Author: Anthony MURGO

Repository: github.com/spmfrance-cloud/aria-protocol | License: MIT