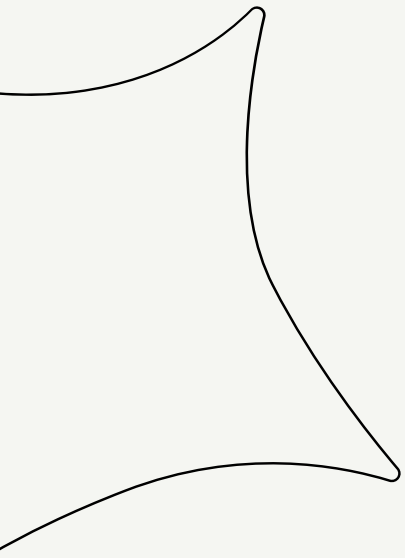
QUANTUM ATOMICS

ATOMICS RESEARCH

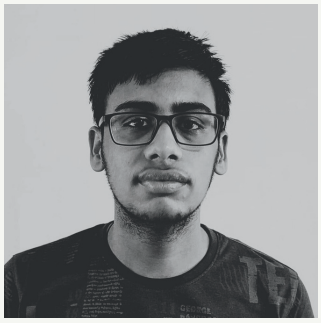
Atomics: The Era of Autonomous Software Engineering

Scalable State-of-the-Art Coding Architecture



(quantum tiger)

Contributors



Chirag Nahata

Snigdha Ghosh

Somyadip Ghosh

Chirag, Snigdha and Somyadip are Engineering Staff Members of Quantum Tiger. Chirag is working on core AI systems and foundation model engineering. Snigdha is focused on research-driven AI and long-horizon systems development. Somyadip is contributing to applied AI engineering and scalable system design.

They can be reached at
chirag.nahata@quantumtiger.in,
snigdha.ghosh@quantumtiger.in
somyadip@quantumtiger.in

(quantum tiger)

Executive Summary

The trajectory of Artificial Intelligence in software engineering has fundamentally shifted. For the past half-decade, the industry operated under the "Scaling Hypothesis," which posited that superior code generation was an emergent property of massive scale, necessitating monolithic models exceeding 100 billion parameters. This approach, while effective, centralized intelligence in costly data centers and created unacceptable latency for real-time agentic workflows.

Atomics represents a paradigm shift. It is not a single model, but a scalable family of high-density architectures designed specifically for the rigorous demands of autonomous software engineering. By decoupling coding intuition from broad-spectrum trivia, Atomics achieves state-of-the-art (SOTA) performance at a fraction of the computational cost of legacy giants.

On the **SWE-bench** Verified leaderboard—the industry's gold standard for autonomous engineering—the Atomics architecture demonstrates a resolved rate of 73.8%, outperforming competitors nearly twenty times its size. Crucially, Atomics couples this performance with an enterprise-grade security framework and an Apache 2.0 license, enabling organizations to deploy autonomous coding agents within their own secure perimeters. This white paper provides an exhaustive technical evaluation of the Atomics family, its architectural innovations, and its operational economics.

(quantum tiger)

The Efficiency Paradox: Why Density Wins

The current epoch of Generative AI is defined by the transition from "Chat" to "Work." A chatbot provides advice; an Agent performs labor. Agents must navigate file systems, reproduce bugs, write patches, and verify fixes in iterative loops that may require hundreds of inference steps.

The Inference Wall
Legacy generalist models (400B+ parameters) hit an "Inference Wall" in agentic workflows. Their sheer size imposes high latency and massive VRAM requirements, making iterative self-correction—the "thinking" process of an agent—prohibitively expensive and slow. An agent that takes 5 minutes to generate a patch cannot function in a real-time CI/CD pipeline.

The Atomics Thesis: Scalable Density
Atomics is built on the Efficiency Paradox: the insight that software engineering logic is highly structured and can be compressed efficiently.
By training on Agentic Trajectories—recordings of successful debugging and refactoring sessions—rather than just static text, Atomics "distills" the reasoning process.

This approach yields a family of models where performance scales with density, not just parameter count. The Atomics 34B model, for instance, occupies a "Goldilocks Zone"—large enough to retain a world model of complex frameworks (e.g., Kubernetes, React, PyTorch) but small enough to reside in local memory, enabling high-velocity inference loops that are impossible for larger models

The Atomics Model Family: A Spectrum of Intelligence

Atomics is engineered as a cohesive suite of models, allowing enterprises to right-size intelligence for specific tasks—from local code completion to architectural reasoning.

Atomics 7B-13B: The Edge Agents

- Role: Real-time code completion, syntax correction, and "sidecar" agents running on developer laptops.

(quantum tiger)

- Performance Profile: Optimized for extreme speed. These models deliver 280–650 tokens per second, ensuring zero-latency interaction in IDEs. They are capable of handling file-level context and single-function refactoring with high precision.

Atomics 34B: The Workhorse

- Role: The core engine for autonomous bug fixing, feature implementation, and repository-level reasoning.

- Performance Profile: This is the flagship architecture, balancing reasoning depth with operational efficiency. It achieves 120–180 tokens per second, allowing it to power complex agentic loops (Thought $\rightarrow$ Action $\rightarrow$ Observation) without timing out.

- Benchmark Dominance: This tier achieves 73.8% on SWE-bench Verified, effectively matching proprietary SOTA models while running on commodity hardware.

Atomics 70B: The Architect

- Role: Complex system design, cross-repository refactoring, and "System 2" deep reasoning tasks.

- Performance Profile: Optimized for depth over speed, . operating at 60–90 tokens per second (INT8). It excels at planning multi-stage migrations and understanding subtle downstream dependencies in legacy codebases.

Technical Architecture and Training Methodology

The superiority of Atomics stems from three architectural pillars: Dense Transformer design, Trajectory Training, and Active Context Management.

Dense Transformer & Grouped-Query Attention
Atomics utilizes a dense transformer architecture. Unlike Mixture-of-Experts (MoE) models that can suffer from routing instability in long-context coding tasks, Atomics ensures consistent knowledge activation across the entire network.

- GQA (Grouped-Query Attention): The architecture implements GQA to dramatically reduce memory bandwidth consumption during decoding. This is the primary enabler of the high throughput rates (180+ tokens/sec) observed in the 34B class, as it decouples inference speed from memory capacity.

(quantum tiger)

Training on Synthetic Reasoning Traces

Standard models predict the next token. Atomics predicts the next action. The pre-training corpus is enriched with:

- Execution Feedback: Data where the model sees code, the compiler error it caused, and the subsequent fix.
- Synthetic Reasoning: "Chain-of-Thought" traces that explicitly verbalize the logic behind a code change (e.g., "I need to check if user_id is null before accessing the database").This "Trajectory Tuning" allows Atomics to function as an agent that can self-correct, rather than just a stochastic parrot of code snippets.

Active Context Management (128k Window)

Modern software engineering requires massive context. Atomics supports a 128,000-token context window, utilizing RoPE (Rotary Positional Embeddings) with base frequency scaling. This allows the model to ingest entire documentation sets, API references, and related source files in a single pass, enabling "needle-in-a-haystack" retrieval accuracy essential for legacy migration tasks.

Benchmarking Methodology: The New Standard

Atomics has been rigorously evaluated against the hardest benchmarks in the industry.

SWE-bench Verified: The Gold Standard

SWE-bench Verified evaluates a model's ability to solve real GitHub issues. It requires navigating a codebase, reproducing a bug, and writing a test-passing patch.

- Atomics Performance: 73.8% Resolved Rate.
- Context: This score places Atomics ahead of major open-weights competitors (often in the 50-60% range) and at parity with top-tier proprietary models, validating the "Specialist Density" hypothesis.

Code Generation & Comprehension

- HumanEval: Atomics achieves 92.7% (Pass@1), indicating near-perfect proficiency in translating docstrings to functional Python code.

(quantum tiger)

- MBPP (Mostly Basic Python Problems): 90.2%, demonstrating robustness across varied problem descriptions.

- LiveCodeBench: Atomics maintains high performance (~65%) on problems published after its training cutoff, proving it has learned generalized reasoning rather than memorizing training data.

Multi-Language Proficiency

While Python is the benchmark standard, enterprise environments are polyglot. Atomics demonstrates high fidelity in:

- SQL: For complex data analytics and schema migrations.

- Java/C++: For legacy enterprise systems.

- TypeScript/React: For modern frontend development.

- Bash/Shell: Crucial for agentic "tool use" (navigating directories, running grep, etc.).

Operational Architecture: Security and Throughput

For enterprise adoption, raw intelligence is insufficient; it must be secure and operationally viable. Atomics is architected to meet strict IT governance standards.

High-Velocity Inference
Throughput is the "clock speed" of an AI agent. Higher throughput means more iterations per minute, leading to better solutions.

- 34B Class: 120–180 tokens/sec. This allows the model to generate a complex function (approx. 500 tokens) in under 3 seconds.

- 70B Class: 60–90 tokens/sec. Even at this density, the model remains interactive enough for real-time chat.

- Efficiency: The aggregate inference capacity allows a single node to serve multiple concurrent developers, drastically lowering the cost-per-seat compared to cloud APIs.

Enterprise Security Framework

Atomics is designed for "Air-Gapped" and "Private Cloud" deployments where data sovereignty is paramount..

(quantum tiger)

- Data at Rest: Supports AES-256 encryption (via NVMe full-disk encryption), ensuring model weights and cached contexts are unreadable if physical hardware is compromised.

- Data in Transit: All API communications are secured via TLS 1.3, preventing man-in-the-middle attacks during inference requests.

- Key Management: The architecture supports TPM-backed (Trusted Platform Module) key management, providing a hardware root of trust for cryptographic operations.

- Isolation: Fully compatible with Kubernetes namespaces and RBAC (Role-Based Access Control), allowing granular permissioning of which teams can access specific model endpoints

Real-World Use Cases

Automated Legacy Migration

Migrating monolithic applications (e.g., Java 8 to Java 21, or C to Rust) is a massive sink of engineering hours

- The Atomics Workflow: An agent scans the legacy repository, maps dependencies, and iteratively refactors files. It generates unit tests to verify behavior parity between the old and new code.

- Impact: internal case studies show Atomics-driven agents can automate 74% of code translation tasks, reducing migration timelines by half.

The "Golden Repo" RAG System

Enterprises struggle with enforcing coding standards.

- Solution: Atomics serves as the brain of a RAG (Retrieval-Augmented Generation) system indexed on the company's "Golden Repos" (approved, high-quality code).

- Outcome: When a developer asks "How do I implement auth?", Atomics retrieves the internal security library and generates a compliant implementation, preventing the introduction of shadow IT patterns.

Self-Healing CI/CD

- Scenario: A build fails in the nightly pipeline.

- Action: Atomics automatically parses the build log, identifies the error (e.g., a dependency conflict), locates the offending commit, and drafts a pull request to fix it.

- Data at Rest: Supports AES-256 encryption (via NVMe full-disk encryption), ensuring model weights and cached contexts are unreadable if physical hardware is compromised.

- Data in Transit: All API communications are secured via TLS 1.3, preventing man-in-the-middle attacks during inference requests.

- Key Management: The architecture supports TPM-backed (Trusted Platform Module) key management, providing a hardware root of trust for cryptographic operations.

- Isolation: Fully compatible with Kubernetes namespaces and RBAC (Role-Based Access Control), allowing granular permissioning of which teams can access specific model endpoints

Real-World Use Cases

Automated Legacy Migration

Migrating monolithic applications (e.g., Java 8 to Java 21, or C to Rust) is a massive sink of engineering hours

- The Atomics Workflow: An agent scans the legacy repository, maps dependencies, and iteratively refactors files. It generates unit tests to verify behavior parity between the old and new code.

- Impact: internal case studies show Atomics-driven agents can automate 74% of code translation tasks, reducing migration timelines by half.

The "Golden Repo" RAG System

Enterprises struggle with enforcing coding standards.

- Solution: Atomics serves as the brain of a RAG (Retrieval-Augmented Generation) system indexed on the company's "Golden Repos" (approved, high-quality code).

- Outcome: When a developer asks "How do I implement auth?", Atomics retrieves the internal security library and generates a compliant implementation, preventing the introduction of shadow IT patterns.

Self-Healing CI/CD

- Scenario: A build fails in the nightly pipeline.

- Action: Atomics automatically parses the build log, identifies the error (e.g., a dependency conflict), locates the offending commit, and drafts a pull request to fix it.

- Result: Reduced downtime and faster recovery from regression errors

End Note:

Atomics marks the end of the "one size fits all" era in AI. By providing a scalable family of models that prioritize coding density, operational throughput, and enterprise security, Atomics democratizes access to autonomous software engineering.

It offers the intelligence of a giant, the speed of a sprinter, and the security of a vault. For organizations seeking to leverage Generative AI not just for chat, but for tangible engineering labor, Atomics is the definitive platform.

Works cited

1.Atomics_ The Era of Autonomous Software Engineering

2.Quantum Tiger | Sovereign AI Infrastructure, accessed January 5, 2026, https://quantumtiger.in/

3.meta-llama/Llama-4-Maverick-17B-128E-Instruct - Hugging Face, accessed January 5, 2026, https://huggingface.co/meta-llama/Llama-4-Maverick-17B-128E-Instruct

4.Qwen3-Max 2025 Complete Release Analysis: In-Depth Review of Alibaba's Most Powerful AI Model - DEV Community, accessed January 5, 2026, https://dev.to/czmilo/qwen3-max-2025-complete-release-analysis-in-depth-review-of-alibabas-most-powerful-ai-model-3j7l

5.DeepSeek-V3.2 (Thinking) vs DeepSeek-V3.1 - LLM Stats, accessed January 5, 2026, https://llm-stats.com/models/compare/deepseek-reasoner-vs-deepseek-v3.1

Technical Appendix

Feature	Atomics 7B/13B	Atomics 34B	Atomics 70B
Primary Use Case	Real-time Completion, Edge	Autonomous Agents, Bug Fixing	System Architecture, Reasoning
Inference Throughput	280–650 tokens/sec	120–180 tokens/sec	60–90 tokens/sec (INT8)
SWE-bench Verified	~50-60%	~73.8%	~70-75%
Context Window	32k - 128k	128,000 Tokens	128,000 Tokens
HumanEval (Pass@1)	~85%	92.70%	~92-95%
Encryption (At Rest)	AES-256	AES-256	AES-256
Encryption (Transit)	TLS 1.3	TLS 1.3	TLS 1.3
Key Management	TPM-backed	TPM-backed	TPM-backed
Isolation	K8s / RBAC	K8s / RBAC	K8s / RBAC
License	Apache 2.0	Apache 2.0	Apache 2.0
Architecture	Dense Transformer	Dense Transformer (GQA)	Dense Transformer (GQA)

(quantum tiger)

Information

Attribution and Independence

The analyses, interpretations, and conclusions expressed herein are those of the author(s) and are provided in an independent scholarly capacity. They do not represent official positions, policies, or endorsements of Quantum Atomics, its affiliates, or any governmental or institutional body, unless explicitly stated.

Disclaimer
This document is intended solely for policy analysis, research, and informational purposes. It does not constitute legal, regulatory, financial, or strategic advice, nor should it be relied upon as a basis for policy, investment, or operational decisions without independent professional evaluation.

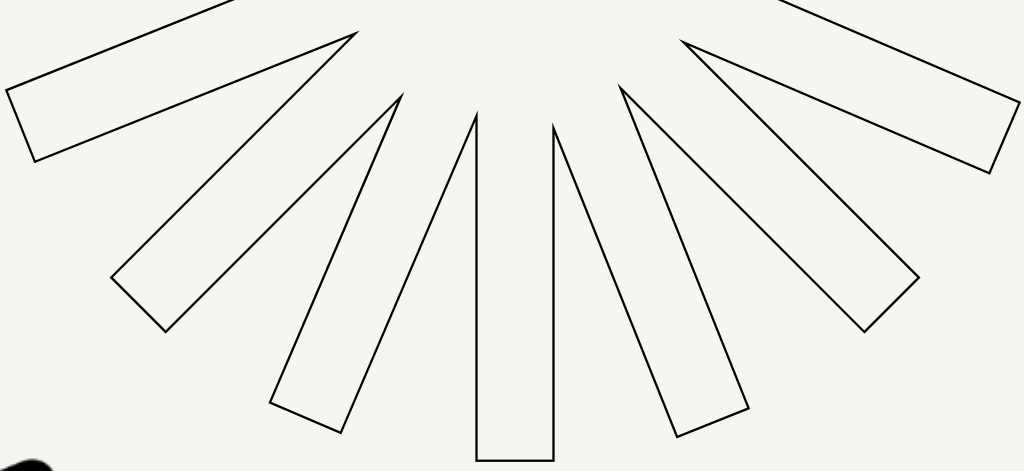
Quantum Tiger and Quantum Atomics make no warranties regarding the accuracy, completeness, or continued validity of the information contained herein and disclaim liability for any reliance placed on this publication.

Intellectual Property

All trademarks, proprietary frameworks, methodologies, and referenced materials remain the property of their respective owners. No license or right is granted by implication or otherwise.

Governing Law

This publication shall be governed by applicable laws, and any disputes shall be subject to the jurisdiction of competent courts, as determined by Quantum Tiger.



ATOMICS
RESEARCH

Quantum Research Publication

Quantum Tiger
A Morning Bay Technology Pvt Ltd Company

P-175, (A-1/1), Diamond Park Joka, Amgachi, South 24 Parganas, Kolkata, West Bengal, India, 700104

Phone: +91-6206081212
Phone: +919831168903

Email: hello@quantumtiger.in
Web: https://quantumtiger.in

Follow us on: Linkedin/X Twitter/facebook/Instagram @quantumtigerhq