The AI Frontier

Hidden Bottlenecks, Blindspots and the Future of AI Infrastructure

A Healthcare Analogy for Investors and Practitioners

Author: Phil Marshall, PhD

Publish Date: June 2026

Takeaways:

AI is evolving into a layered, coordinated system. The structure of modern healthcare provides a useful analogy for understanding this shift.

Model scale remains important, but AI competition is increasingly shaped by end-to-end infrastructure efficiency.

Moving data is becoming as important as processing it.

Tiered AI solutions are becoming competitively important, and organizations that embrace emerging optimization techniques will be better positioned to match workloads to the right model and infrastructure layer.

Agentic AI is where many commercial AI opportunities are likely to be realized, but capturing that value requires a shift from optimizing computation to coordinating models, tools, data, and workflows.

As infrastructure demands shift, investors risk misvaluing critical AI infrastructure components.

© Copyright Tolaga Research, 2026

History shows that technology pioneers are often not the long-term winners. Successful markets attract waves of investment, innovation, and competition that can overwhelm early leaders. Netscape was eclipsed by Microsoft and later Google, while MySpace gave way to Facebook.

Tech investors recognize these same dynamics in Artificial Intelligence (AI) today. While the opportunity is enormous, the pace of innovation is relentless. Competition is intense, market positions are shifting rapidly, and companies across virtually every industry are racing to secure their place in the emerging AI ecosystem.

This creates substantial uncertainty about how AI will ultimately evolve, where durable value will accrue, and which parts of the ecosystem will prove most strategically important. To help navigate this uncertainty, we use healthcare as a framework for understanding the AI infrastructure stack and its emerging value chain. While no analogy is perfect, healthcare provides a useful lens for examining the roles, dependencies, bottlenecks, and competitive dynamics emerging across the AI ecosystem.

We then investigate several common blind spots relating to compute, connectivity, memory, and orchestration, and consider how these factors may influence the competitive positioning of AI infrastructure and platform providers as the market evolves.

In related research, we use our AI Infrastructure Simulator to examine how key AI innovations affect infrastructure cost, performance, and efficiency.

Vibrant Innovation Across Entire Infrastructure Stack



Source: Tolaga Research, 2026

© Copyright Tolaga Research, 2026

Part One: The Analogy

Modern healthcare manages complexity through a layered system of primary care clinics, regional care centers, major hospitals, specialists, and care coordinators. Expertise is applied selectively, escalating resources only when needed and coordinating specialized capabilities at scale. Frontier AI infrastructure has a similar layered logic, making healthcare a useful framework for understanding how AI systems allocate intelligence, manage cost, and where bottlenecks are emerging.

Primary Care Clinic: Small Language Models

Most healthcare interactions are handled by primary care clinics, with only the more complex cases escalated to specialists or major hospitals. Small Language Models (SLMs) play a similar role in AI, handling everyday workloads efficiently while reserving frontier models for tasks that require greater reasoning capability.

SLMs can be deployed locally, at the edge, or as part of larger AI platforms. Their advantages are lower cost, lower latency, and greater deployment flexibility.

Like primary care clinics, however, SLMs have limits. While increasingly capable, the most complex reasoning and long-horizon tasks may still require escalation to frontier models. In both healthcare and AI, the challenge is determining when escalation is needed and ensuring it occurs efficiently.

The Regional Clinic: Shared Infrastructure for Intermediate Workloads

Not all workloads can be handled efficiently by small models, nor do they all justify dedicated hyperscale infrastructure. Between these extremes, an intermediate layer of AI infrastructure is emerging to balance capability, cost, latency, and scale.

Like regional healthcare networks, this layer provides shared resources that bridge routine and highly specialized AI workloads. Some of these capabilities are built inside frontier AI environments such as OpenAI, Anthropic, Google, Meta, xAI, and DeepSeek, while others are supplied by specialist infrastructure providers, including AI cloud platforms such as CoreWeave, Nebius, and Lambda, and distributed infrastructure from telecommunications providers such as Nokia and Ericsson.

This layer enables organizations to access advanced AI capabilities, including frontier models, without the cost and complexity of building and operating hyperscale infrastructure themselves.

The Major Hospital: Frontier Models

The major hospital is the central coordinating hub of the healthcare system. It integrates across specialties, maintains a comprehensive view of the patient, and provides the highest level of expertise for complex cases. Frontier models, such as the largest variants of GPT, Claude, Gemini, Llama, and Grok, play a similar role. They maintain coherence across massive contexts and provide general-purpose reasoning across a wide range of tasks.

© Copyright Tolaga Research, 2026

The tradeoff is resource consumption. Delivering frontier-scale reasoning is computationally expensive, regardless of the underlying architecture. As with major hospitals, the challenge is not their value but their efficient utilization. Frontier models remain essential for the most demanding workloads, but applying their full capabilities to routine tasks quickly becomes uneconomic at scale.

The Specialist Departments: Mixture of Experts

Hospitals scale expertise through specialization; not every patient needs to see every specialist. Modern Mixture-of-Experts (MoE) architectures apply a similar principle. Systems such as Mixtral, Gemini MoE variants, Grok, and DeepSeek-MoE contain many expert subnetworks. For each inference, they activate only the expert subnetworks most relevant to the task. This allows total model capability to expand without a proportional increase in active compute requirements.

Like hospital specialists, however, MoE experts do not operate in isolation. They share context, representations, and coordination infrastructure. Crucially, specialization introduces a new bottleneck. Just as complex referral networks create scheduling and administrative overhead, MoE systems generate substantial routing, synchronization, and all-to-all network traffic across GPUs.

Care Coordination: Agentic AI

Complex medical cases require a coordination layer to determine which specialists to engage, when to request tests, and how to sequence care over time. Agentic AI systems play a similar role. Rather than merely responding to a single prompt, agents decompose goals, route tasks to specific tools or sub-agents, retrieve external data, and manage multi-step workflows.

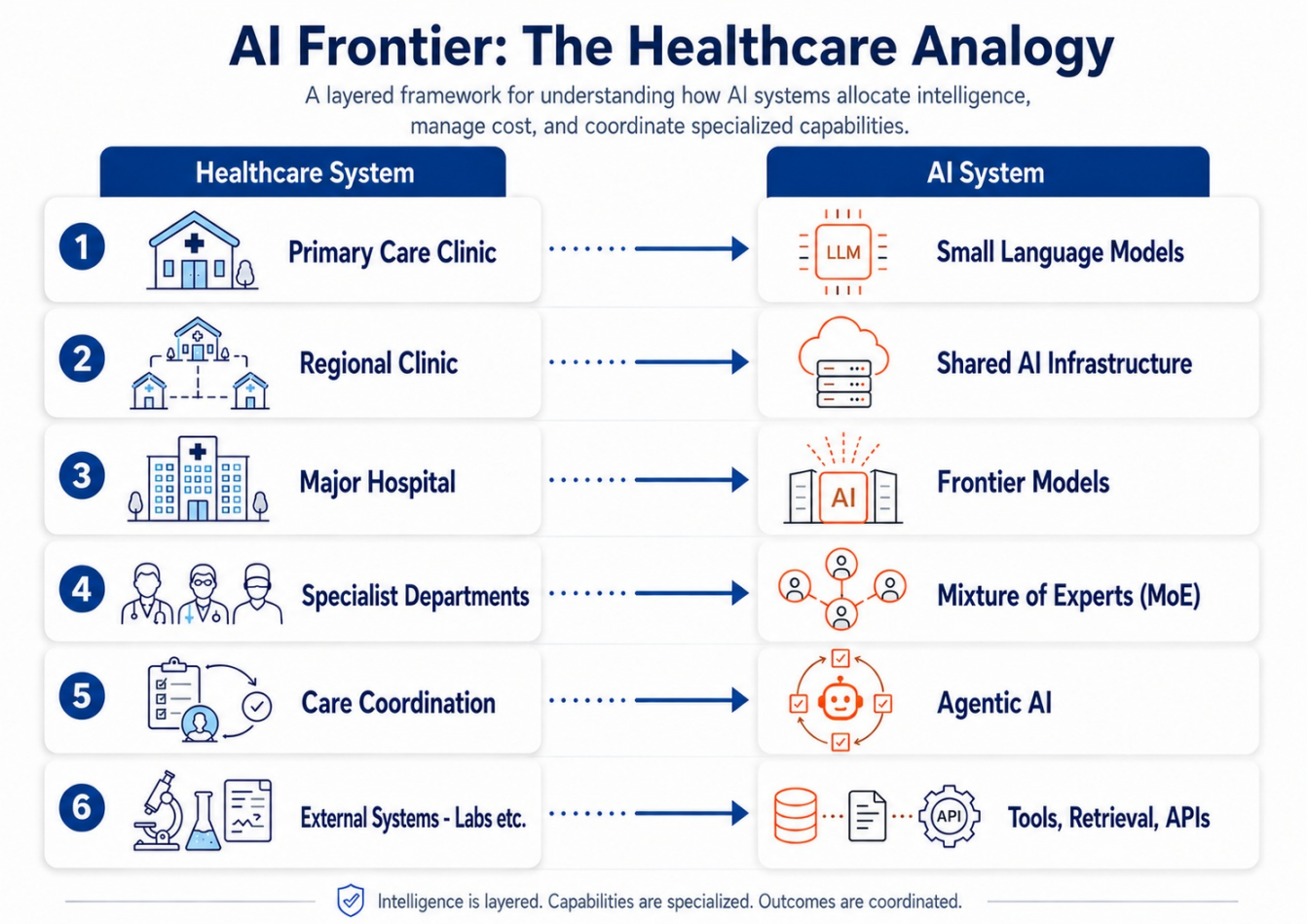
As AI becomes more agentic, the primary challenge shifts from individual model capability to system-wide execution reliability. Success depends on maintaining coherence across long-running workflows, managing dependencies, handling failures, and allocating resources efficiently.

External Systems: Tools, Retrieval, APIs

No hospital operates in isolation; it relies on an interconnected ecosystem of laboratories, imaging centers, pharmacies, and referral networks. AI systems are evolving similarly. Retrieval systems, vector databases, enterprise APIs, and software applications increasingly function as supporting infrastructure around the core model. Platforms such as Pinecone, Weaviate, and Qdrant provide dedicated retrieval and long-term memory capabilities, while orchestration platforms bridge models to enterprise data, tools, and workflows.

As in healthcare, overall system effectiveness increasingly depends on how well its components work together. Frontier AI is no longer defined by model capability alone, but also by how effectively it accesses information, invokes tools, and coordinates resources across a broader ecosystem.

© Copyright Tolaga Research, 2026



Where the Analogy Breaks

Taken together, AI systems are evolving from monolithic models into coordinated networks of specialized components. While structurally useful, the healthcare comparison has several important limitations:

Physical Geography

Healthcare architectures are fundamentally anchored by physical geography; patients and providers must exist in the same physical world. AI architectures are far more flexible. While geography matters under specific conditions such as data sovereignty, data locality, and latency requirements, AI systems are primarily organized around capability, economics, and performance rather than physical proximity.

© Copyright Tolaga Research, 2026

Statefulness vs. Statelessness

Healthcare systems are deeply stateful. Patients accumulate persistent medical records and identities over time. In contrast, most modern AI systems remain largely stateless, operating with restricted context windows and limited long-term continuity between sessions.

Opaque vs. Auditable Specialization

Human specialists have explicit credentials and decision processes. MoE experts do not. Their specializations emerge during training and remain difficult to inspect or explain.

Accountability and Governance

Clinical environments operate within strict frameworks of licensing, liability, and documented referral chains. AI systems largely lack equivalent governance mechanisms. However, this will likely change for AI architectures to operate safely in high-stake domains.

Part Two: The Implications

While the hospital analogy has its limitations, it provides a useful framework for understanding how AI infrastructure is evolving and highlights several common blind spots relating to compute, connectivity, memory, orchestration, and competitive positioning.

Blindspot: Compute Is Not Always The Bottleneck

What many people think

AI performance is primarily about compute. More FLOPS means more performance.

Why people think this

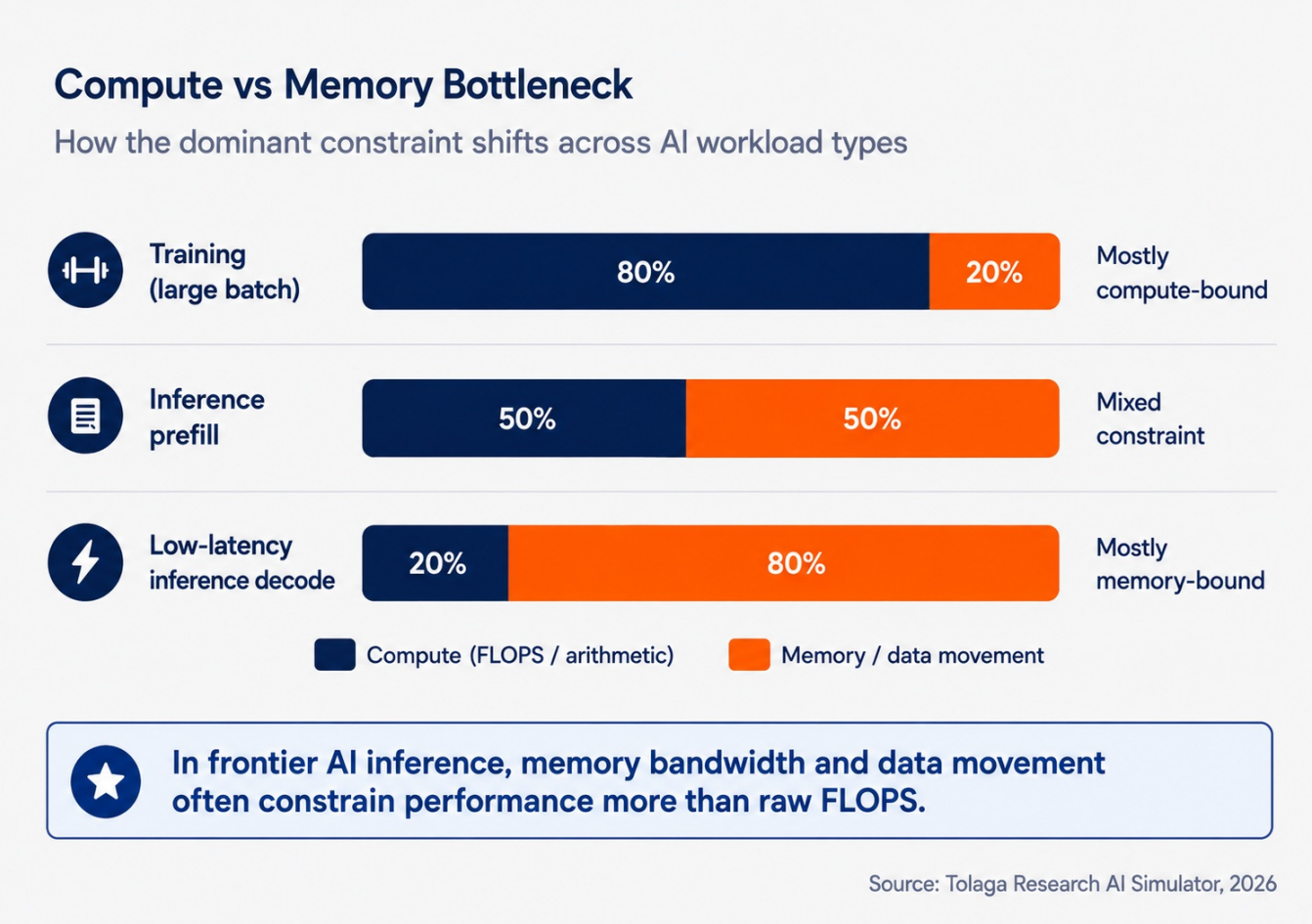
Historically, AI infrastructure was optimized around training workloads, where large batch sizes and massive matrix multiplications made raw compute throughput the dominant constraint.

What is actually happening?

In frontier AI inference, the bottleneck is often not arithmetic throughput but data movement. During token-by-token decoding, GPUs repeatedly access model weights, KV-cache data, and intermediate activations, while tensor-parallel systems also incur synchronization overhead. This can leave expensive compute resources underutilized, making memory bandwidth and interconnect efficiency central to inference performance.

The hospital equivalent is a surgical team waiting for patient records or imaging scans before surgery can begin. The theatre is ready, the staff are present, but the procedure is delayed because critical information has not yet arrived.

© Copyright Tolaga Research, 2026



Implication for Investors and Practitioners

The hardware race is not just about peak FLOPS. In real inference workloads, a chip with significantly higher memory bandwidth can outperform one with greater theoretical compute. This gap between benchmark performance and deployed performance is increasingly where custom silicon vendors compete.

For more detail, see our research note: **When Memory Not Compute Limits AI Inference**

© Copyright Tolaga Research, 2026

Blindspot: MoE Efficiency Comes With A Networking Tax

What many people think

MoE is more efficient than dense models because only a fraction of parameters activate per token. This is correct, as far as it goes.

However

MoE models generate significant communication traffic because tokens must be routed to experts distributed across multiple GPUs. Within a node, and in some systems across a tightly coupled rack-scale GPU domain, technologies such as NVLink and NVSwitch can reduce this overhead. The challenge becomes more difficult as workloads span multiple nodes, racks, or clusters, where all-to-all communication, synchronization, routing overhead, and expert load imbalance can become major sources of latency, sometimes eroding the theoretical efficiency gains that make MoE architectures attractive in the first place.

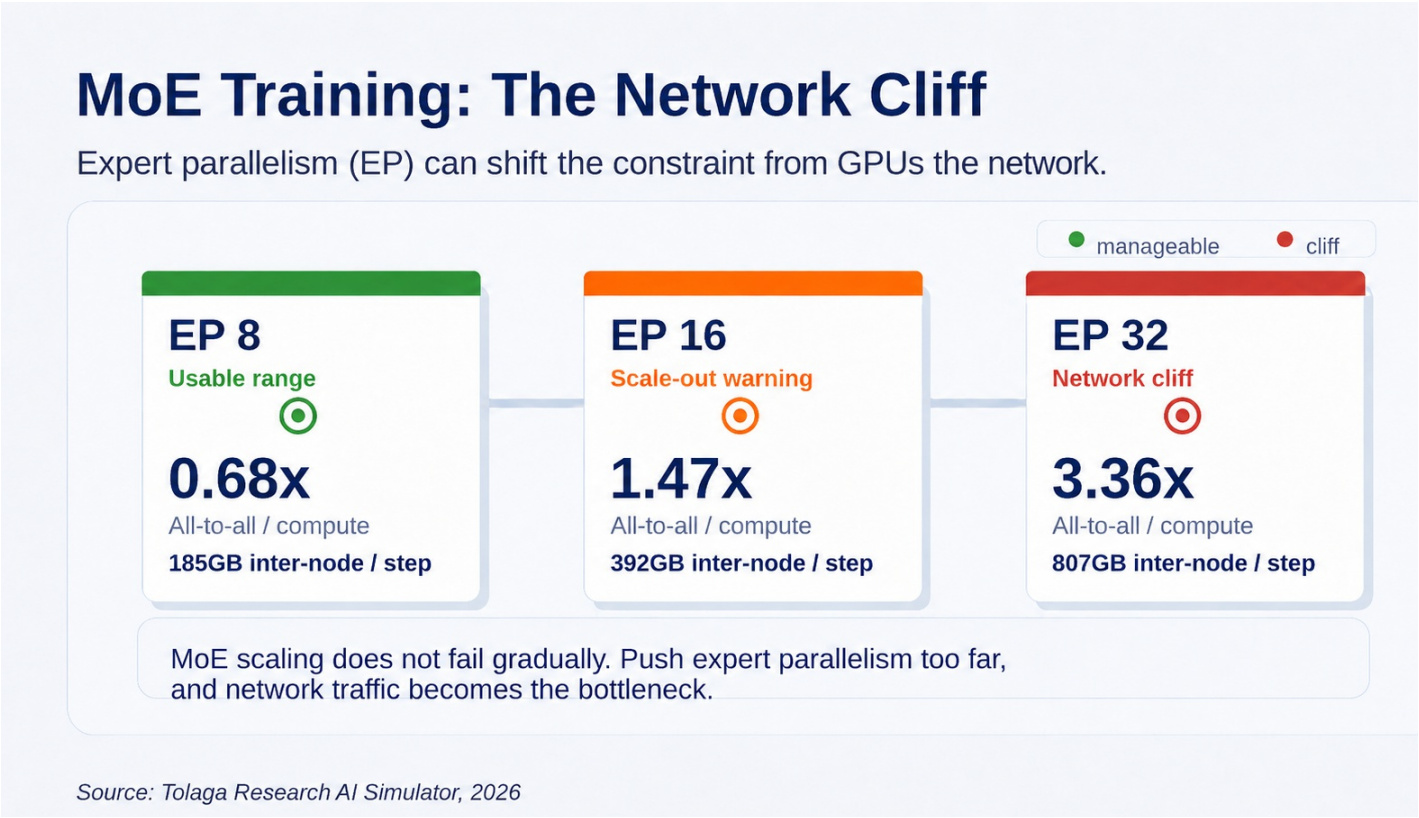
This exposes a broader infrastructure reality: at frontier scale, AI performance depends not only on GPU performance, but also on how efficiently data moves within servers, across racks, between data centres, and ultimately to users.

The hospital equivalent is a specialist referral network. Referrals can improve outcomes by directing patients to the most appropriate expertise, but they also introduce coordination, scheduling, and handoff overhead. As these interactions scale, the referral process itself can consume a meaningful share of total care time. At scale, a poorly coordinated specialist network can become slower than a well-run general hospital despite having superior expertise.

MoE efficiency gains are real, but they depend heavily on the surrounding infrastructure. Performance is shaped as much by routing efficiency, synchronization overhead, and interconnect topology as by the architecture itself. Poorly optimized MoE deployments can underperform dense models despite their theoretical compute advantages. In large deployments, the strongest gains increasingly depend on custom networking fabrics and topology-aware orchestration.

For more detail, see our research note: **When Experts Become the Bottleneck**

© Copyright Tolaga Research, 2026



Source: Tolaga Research AI Simulator, 2026

Networking Is Becoming Core AI Infrastructure

The implication extends beyond MoE. Distributed training, large-scale inference, RAG systems, agentic workflows, and edge deployment all depend on fast, reliable movement of data across the AI stack. Networking is no longer a secondary layer beneath compute. It is becoming a primary determinant of latency, throughput, utilization, and cost.

The challenge also extends beyond the data center. User experience depends not only on model capability but also on the path between the inference system and the user. In some cases, smaller models deployed closer to users can deliver a better experience than more capable models running in distant centralized facilities.

For investors, this suggests that networking infrastructure may be materially underappreciated relative to GPUs. As AI systems become more distributed, companies providing high-bandwidth interconnects, low-latency fabrics, and advanced networking architectures are developing durable competitive advantages that are difficult to replicate.

Look out for our upcoming AI Connectivity research series.

© Copyright Tolaga Research, 2026

Blindspot: Inference Is Not a Single Market

Much of the discussion around AI infrastructure focuses on the distinction between training and inference. While the two have different requirements, an equally important shift is occurring within inference itself. The market is fragmenting into workloads with very different economic and architectural needs.

Regime A: High-frequency, low-complexity inference

Consumer chat, coding assistants, API inference, and real-time agentic tasks prioritize throughput, latency, utilization, and low cost per token. These workloads increasingly rely on distilled and quantized models, speculative decoding, custom silicon, and regional or edge deployments closer to users. In healthcare terms, this resembles primary care clinics and regional hospitals: high-volume systems optimized for speed, efficiency, and accessibility.

Regime B: Low-frequency, high-complexity inference

Research synthesis, scientific discovery, legal analysis, and deep reasoning workloads prioritize capability over cost. These systems optimize for reasoning depth, large context retention, reliability, and frontier-scale infrastructure. This is the equivalent of specialist surgery or tertiary care: expensive, centralized, and reserved for the most complex cases.

The implication is that AI infrastructure is no longer converging toward a single architecture. It is fragmenting into layered systems optimized for fundamentally different operational and economic realities.

For investors, Regime A and Regime B may appear similar but increasingly operate under different economic models. Regime A rewards efficiency, latency, and scale economics, while Regime B rewards reasoning depth, capability density, and outcome quality. Treating them as a single market risks mispricing both.

© Copyright Tolaga Research, 2026

Blindspot: AI Economics Are Being Reshaped By Optimization

The economics of AI are often framed around model architectures and GPUs. In practice, however, inference and training costs are increasingly shaped by optimizations across the full stack. These techniques determine how efficiently models use compute, memory, networking, and specialized hardware once they are deployed at scale.

The exhibit below summarizes an illustrative set of optimization techniques. Some, such as hybrid parallelism, mixed precision, fast interconnects, retrieval-augmented generation (RAG), and ZeRO-style sharding, are embedded in Tolaga's baseline AI Infrastructure Simulator. Others, including KV cache optimization, quantization, distillation, speculative decoding, and custom silicon and LPU acceleration, are modeled separately to test their impact across different deployment scenarios.

AI Optimization Across the Stack

Illustrative examples of how inference and training optimize across different operating constraints.

Layer / Category	Inference Optimizations	Training Optimizations
Model Architecture & Design	GQA/MQA/MLA; MoE routing; sparse attention; modular design	MoE/sparse designs; architecture search; scaling choices
Model Compression & Adaptation	Weight, activation and KV quantization; pruning; distilled models	Quantization-Aware Training; pruning; distillation; LoRA/QLoRA
Data Efficiency	Prompt/context compression; retrieval filtering; context selection	Curation; deduplication; curriculum; synthetic data
Algorithmic Acceleration	Speculative decoding; early exit; adaptive / parallel decoding	Advanced optimizers; mixed precision; gradient accumulation
Memory, Attention & Context Management	KV cache paging, quantization, eviction and offload; prefix/radix caching; FlashAttention; long-context	Checkpointing; ZeRO sharding; sequence/context parallelism; offload
Serving / Training System Optimization	Dynamic batching; prefill/decode split; scheduling; routing; tiered inference	Dynamic batching; checkpointing; fault tolerance; DeepSpeed/Megatron
Distributed Infrastructure	TP/PP/DP/EP; distributed KV cache; network-aware routing	Hybrid parallelism; communication compression and overlap
Hardware Architecture	GPUs, TPUs, LPUs, NPUs; custom silicon; energy per token	High-HBM accelerators; fast interconnects; training ASICs; cluster energy optimization

© Copyright Tolaga Research, 2026

KV Cache Optimization: Managing the Memory Burden of Inference

KV cache optimization is an important part of the AI inference optimization stack, especially for long-context, high-concurrency, and memory-constrained workloads. In transformer models, each generated token attends to prior tokens using Key and Value attention-state tensors. Rather than recomputing this attention state for the full prompt and all previously generated tokens at every decoding step, the model stores previously computed Key and Value tensors in a KV cache.

This allows the decode stage to reuse prior computations and focus on generating the next token.

The tradeoff is memory. KV cache memory grows with context length, output length, batch size, number of layers, and the attention representation. For long-context models, RAG, multi-turn conversations, and agentic workflows, the KV cache can become a major constraint on GPU memory capacity and memory bandwidth. In these cases, inference performance may be limited less by theoretical FLOPS than by how efficiently memory is allocated, reused, compressed, and moved.

Several techniques are used to reduce this burden. **Paged KV cache management** stores cache data in smaller memory blocks, reducing fragmentation and allowing serving systems to support more concurrent requests. **Prefix caching** reuses cached states for repeated system prompts, templates, shared instructions, or common document prefixes. **KV cache quantization** reduces the memory footprint of the cache by storing Key and Value tensors at lower precision. **Cache eviction and compression** reduce the amount of historical context retained when full attention over all prior tokens is no longer necessary. **Multi-query attention** and **grouped-query attention** reduce the number of Key and Value heads that must be stored, lowering cache memory requirements at the model architecture level.

These techniques are complementary rather than mutually exclusive. A production inference system may combine paged cache allocation, prefix caching, continuous batching, and KV cache quantization in the serving stack, while also using model-level techniques such as grouped-query attention. The result is better memory utilization, higher concurrency, lower latency, and improved throughput, particularly where workloads reuse context or operate near GPU memory limits.

© Copyright Tolaga Research, 2026

KV Cache Optimization Techniques

Serving-level and model-level methods used to improve inference memory efficiency.

Technique	Where it applies	Examples	Brief note
Paged KV cache	Serving engine	vLLM PagedAttention, TensorRT-LLM, SGLang, LMDeploy	Reduces memory fragmentation.
Prefix caching / KV reuse	Serving engine	vLLM, SGLang RadixAttention, TensorRT-LLM, Modular MAX	Reuses repeated prompt prefixes.
KV cache quantization	Serving engine / model runtime	vLLM, TensorRT-LLM, SGLang, LMDeploy, TGI	Compresses cached Key/Value tensors.
GQA / MQA	Model architecture	Llama, Mistral, Qwen, Gemma, Granite	Reduces Key/Value heads.
Eviction / compression / offloading	Serving / memory tiering	LMCache, NVIDIA Dynamo, SGLang, custom stacks	Frees or moves cache memory.

KV cache optimization should not be viewed as a replacement for other inference optimization methods. In addition, it illustrates a broader point: AI inference is increasingly a systems-engineering problem. As context windows expand and production workloads become more interactive, KV cache optimization matters most where latency, throughput, concurrency, and cost are constrained by memory rather than raw compute. Its value remains workload-dependent, but for long-context and high-volume AI services, it is becoming a first-order determinant of inference economics.

Look out for our upcoming study on KV cache optimization

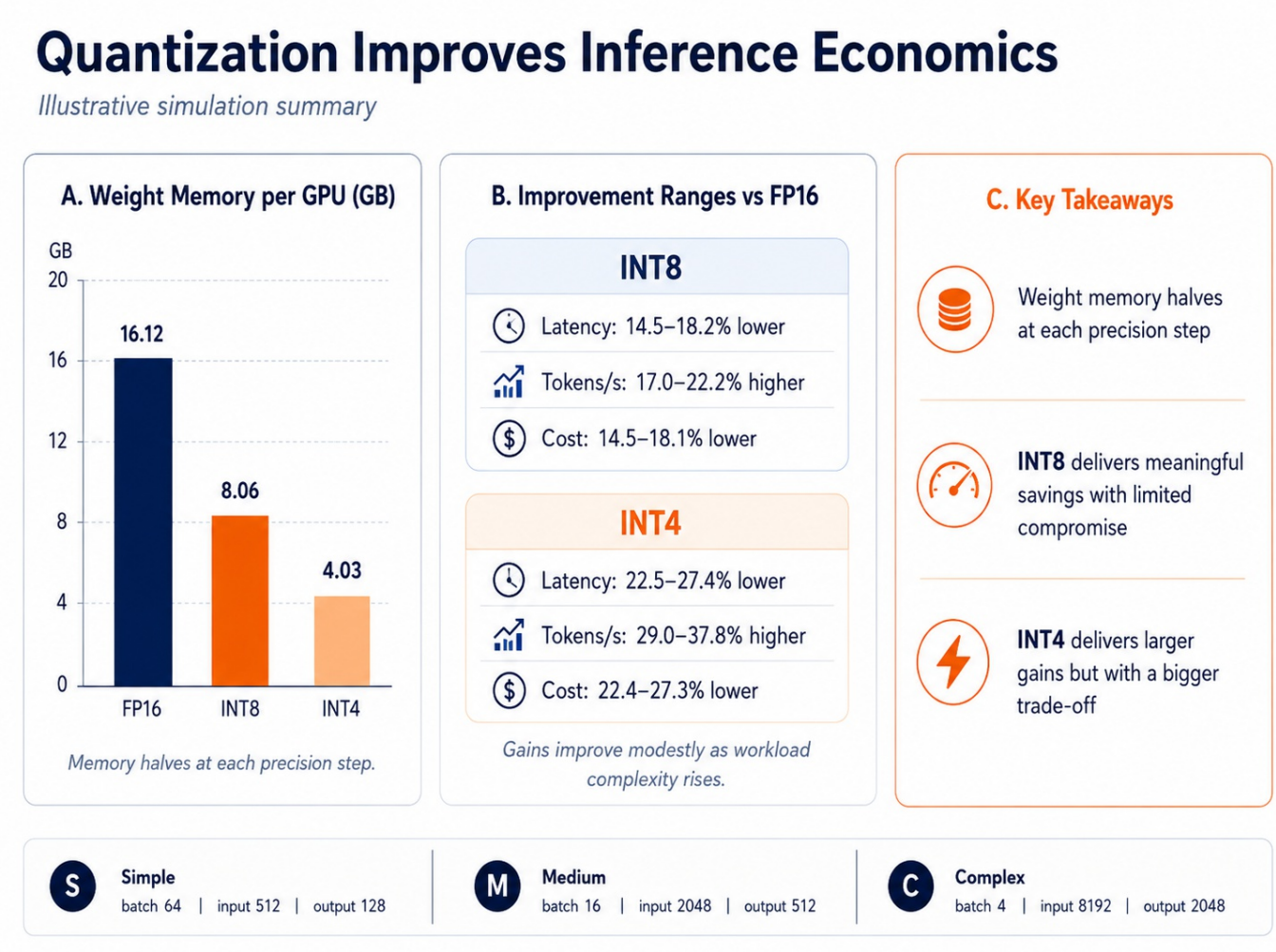
© Copyright Tolaga Research, 2026

Quantization: Compressing Model Weights to Lower Inference Costs

Quantization reduces the precision of model weights, typically from 16-bit to 8-bit or 4-bit formats. By reducing the amount of data that must be stored and moved through the system, it lowers memory requirements and can improve inference latency, throughput, and cost. Modern quantization techniques, including GPTQ and AWQ, can preserve much of the original model quality while delivering meaningful efficiency gains.

More aggressive compression, such as 4-bit quantization, can deliver larger infrastructure benefits, but it may increase quality risk, especially for complex reasoning tasks.

The hospital equivalent is using a concise patient summary rather than storing a complete medical record in every room. For most consultations, the summary contains enough information while significantly reducing storage and retrieval overhead.



© Copyright Tolaga Research, 2026

The modeled results above illustrate the basic trade-off. Moving from FP16 to INT8 halves modeled weight memory per GPU shard from 16.12 GB to 8.06 GB. Across the simulated workloads, INT8 lowers latency by approximately 14.5% to 18.2%, increases output tokens per second by approximately 17.0% to 22.2%, and lowers cost per token by approximately 14.5% to 18.1%.

Moving from FP16 to INT4 reduces modeled weight memory further to 4.03 GB per GPU shard, a 75% reduction relative to FP16. In the simulation, INT4 lowers latency by approximately 22.5% to 27.4%, increases output tokens per second by approximately 29.0% to 37.8%, and reduces cost per token by approximately 22.4% to 27.3%.

These results suggest that INT8 can provide meaningful efficiency gains with limited compromise, while INT4 offers larger savings but with a higher risk of compression-related quality degradation. Quantization is therefore most attractive when memory efficiency, latency, or serving cost are binding constraints, and where the target workload can tolerate some precision-related quality risk.

For more detail, see our research note: **Trading Precision for Efficiency with Quantization**

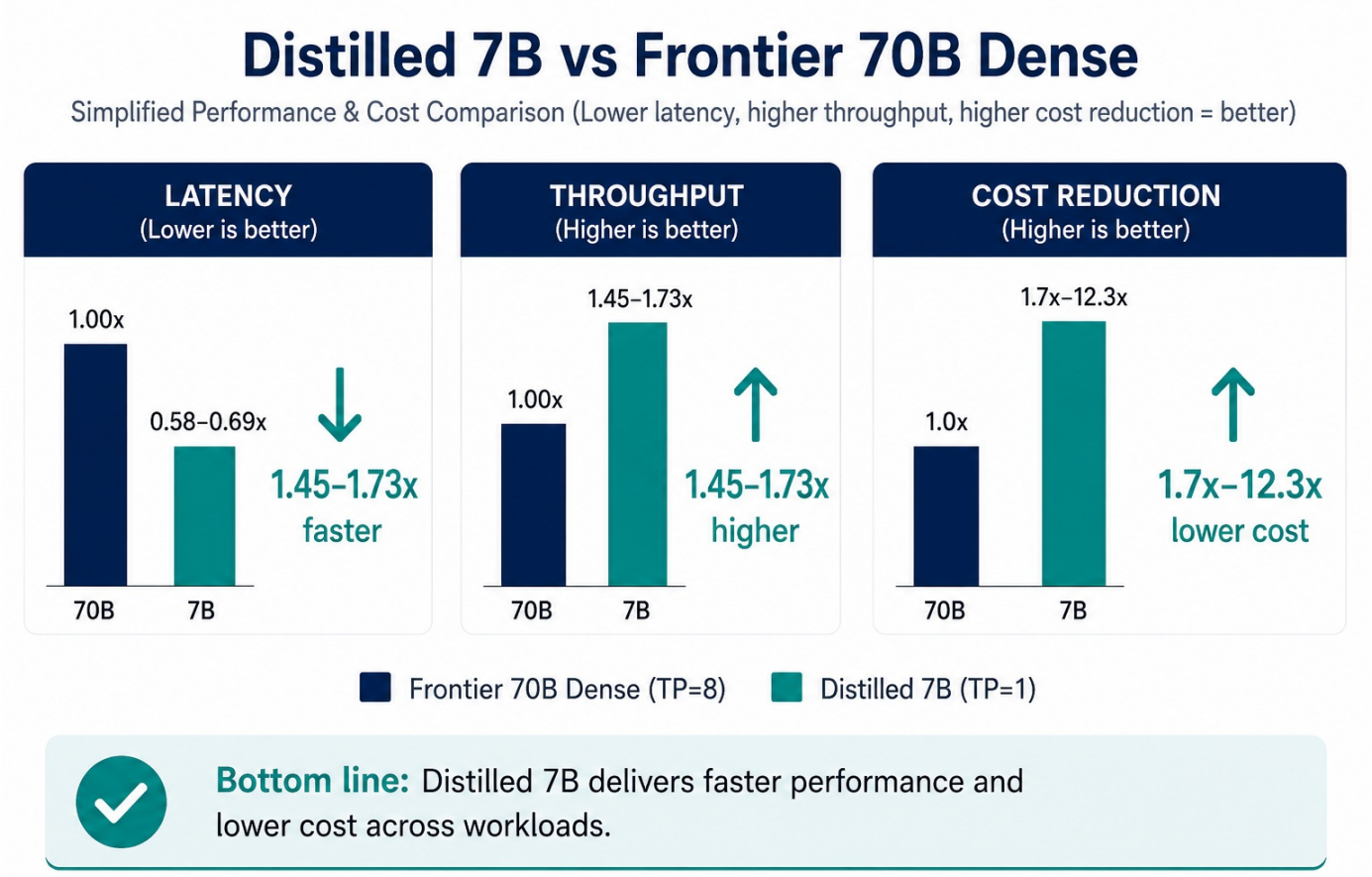
Distillation: Moving Routine AI Workloads to Smaller Models

Model distillation transfers selected capabilities from larger frontier models into smaller, cheaper deployment models. It is not a general replacement for frontier AI, but it can materially improve inference economics when the workload is narrow, repeatable, and measurable.

The strongest opportunity is in tiered inference. Routine and well-understood requests can be served by smaller distilled models, while complex, ambiguous, high-risk, or high-value cases are routed to larger frontier models. In this architecture, the value comes not just from the smaller model, but from the routing, evaluation, monitoring, and escalation system around it.

The economics can be significant, but they depend on routing quality. Based on illustrative results from our AI Infrastructure Simulator, a distilled 7B model reduces latency to roughly 0.58 to 0.69x of a 70B baseline, implying about 1.45 to 1.73x faster response times. Throughput improves by a similar 1.45 to 1.73x, while modeled serving cost falls by a factor of approximately 1.7 to 12.3x across the workloads tested. However, these savings depend on how much traffic can safely remain on the smaller model. If a meaningful share of requests still needs to escalate to a 70B or frontier model, the effective cost advantage narrows.

© Copyright Tolaga Research, 2026



Public benchmark results also show the limits of distillation. Smaller distilled models can retain strong performance on selected tasks, but performance usually weakens as tasks become broader, more complex, or less similar to the distillation data. This is especially important for coding, scientific reasoning, multi-step reasoning, and unfamiliar edge cases. As a result, workload-specific evaluation is essential before production deployment.

The broader implication is that AI deployment is becoming more layered. Frontier models remain important for creating new capabilities, generating synthetic data, handling difficult reasoning, and validating lower-tier outputs. Distilled models create value when defined, repeatable workloads can be handled at lower latency and cost without unacceptable quality loss. Competitive advantage increasingly shifts toward organizations that can match each task to the right model tier while maintaining quality, governance, and cost control.

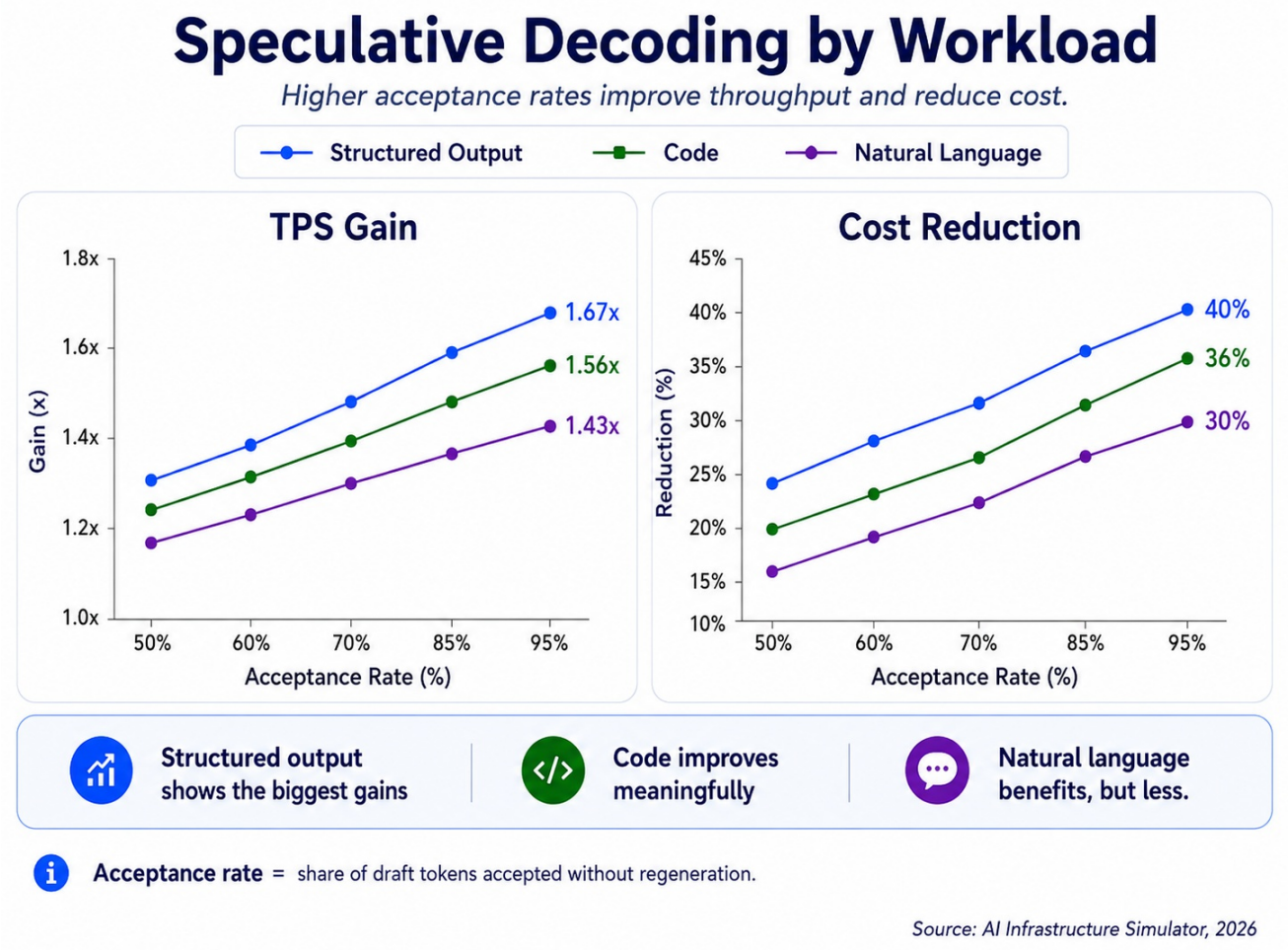
For more detail, see our research report: **Model Distillation and the Diffusion of AI Capability**

© Copyright Tolaga Research, 2026

Speculative Decoding and Inference Throughput

Speculative decoding improves inference throughput by using a smaller draft model, or another speculation mechanism, to propose tokens ahead of the larger target model. The target model then verifies those proposed tokens and accepts or rejects them. The effectiveness of speculative decoding depends mainly on the acceptance rate: higher acceptance reduces sequential decoding work, increasing throughput and lowering cost. The gains are strongest for predictable workloads such as structured outputs, JSON, extraction, function calls, code completion, and templated generation, while open-ended natural language benefits less.

Simulation results show end-to-end tokens-per-second (TPS) gains of up to about 1.67x. These end-to-end gains are lower than the decode-only speedups sometimes reported in the literature because speculative decoding primarily accelerates the token decoding phase, not the full inference pipeline. Prefill, memory movement, batching, verification, and serving-stack overheads are not reduced to the same extent, so total TPS improvement is smaller than the decode-only gain.



For more detail, see our research report: **Speculative Decoding and Inference Throughput**

© Copyright Tolaga Research, 2026

Custom Silicon for Efficient Inference

Custom silicon increasingly targets data movement and memory efficiency. Architectures such as TPUs and LPUs trade flexibility for lower latency, higher efficiency, and more predictable performance.

As a result, the same AI model can produce very different latency, throughput, and cost profiles depending on the underlying architecture, even when theoretical FLOPS are similar. Increasingly, inference performance depends as much on moving data efficiently as on performing computation.

Look out for our upcoming LPU research.

The common thread is that optimization techniques increase capability per dollar of infrastructure investment. In production systems, these techniques are often combined across the model, serving stack, memory system, and underlying hardware. As AI matures, competitive advantage is likely to depend as much on optimization, orchestration, and operational efficiency as on model scale itself.

Blindspot: Agentic AI Means More than Compute

Agentic AI changes infrastructure economics because execution time is not driven by model computation alone. Tool calls, retrieval, orchestration logic, API responses, and subagent coordination can all introduce waiting time while GPU resources remain reserved.

This makes effective utilization, or GPU reservation efficiency, a critical metric. It measures the share of reserved infrastructure time spent on useful model computation rather than waiting, coordination, or orchestration overhead.

The hospital equivalent is an operating theatre booked for six hours but performing surgery for only three. The room remains occupied and unavailable for other procedures even while the team is waiting on lab results, patient transfers, or staff availability.

For agentic AI, the economic unit shifts from cost per token toward cost per completed workflow. In tool-heavy environments, low GPU reservation efficiency can make long-running workflows materially more expensive than token-based metrics suggest.

This creates both risk and opportunity. Organizations that rely on overly simple token economics may underestimate infrastructure cost, while those that improve orchestration efficiency, reduce tool latency, and increase active GPU utilization can gain a meaningful operating advantage.

The simulation results below show how agentic workflows can convert GPU capacity into waiting time. As tool call latency increases from 50 ms to 600 ms and reasoning loops expand from 2x to 6x, requests become longer and less efficient. Effective GPU utilization falls from roughly 38% to about 25%, reflecting more time spent on orchestration, retrieval, routing, and external calls rather than token generation.

Look out for our research study: Agentic AI and the Economics of Waiting

© Copyright Tolaga Research, 2026

Part Three: What This Means for Investors and Practitioners

For Investors

The AI infrastructure race is no longer just about building larger models. It is increasingly about delivering useful capability efficiently, reliably, and at scale.

The blind spots discussed throughout this paper suggest that infrastructure constraints are becoming central to AI economics. Memory bandwidth, networking, orchestration, utilization, and systems engineering can matter as much as raw compute. The companies that appear strongest at the model layer may not always capture the most durable value if the underlying infrastructure stack becomes the primary source of cost advantage.

AI inference is also fragmenting into distinct markets with different economics. High-volume workloads reward efficiency, latency, and scale economics, while frontier reasoning workloads reward capability density and outcome quality. Agentic AI adds another layer of complexity by introducing workflow, orchestration, and utilization constraints that cannot be solved simply by deploying more GPUs.

Increasingly, durable competitive advantage may reside in the less visible layers of the stack: memory systems, networking fabrics, orchestration platforms, software infrastructure, and operating discipline. For investors, this means looking beyond headline model performance to the infrastructure and execution capabilities that determine cost, scalability, and defensibility.

For Practitioners

The most important practical shift is to stop optimizing solely for benchmark performance and start optimizing for workload economics.

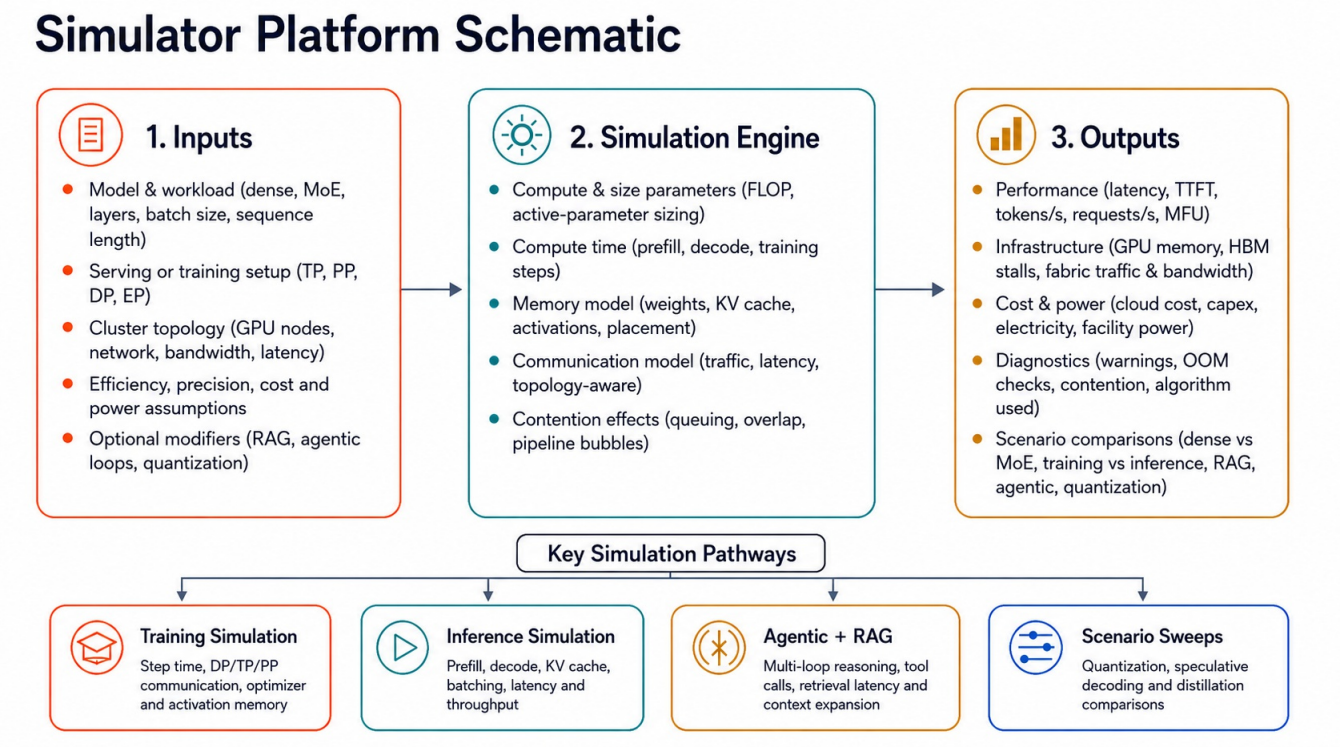
For many production deployments, the optimal solution is not simply the largest available model. It is a workload-specific architecture that delivers sufficient capability at acceptable latency, reliability, governance, and cost. The right design depends on the task, the quality threshold, the risk profile, and the economics of the workflow.

This requires a more disciplined approach to deployment. Teams need to understand where time and cost are being consumed across the full inference pipeline, including model execution, memory movement, coordination, external dependencies, and operational overhead. In agentic systems, workflow efficiency can become as important as model performance.

Architectural decisions made today may be difficult to reverse tomorrow. Systems that deliver impressive capability can also introduce operational complexity, utilization challenges, and governance risk. Increasingly, infrastructure design, workload routing, observability, and operational control are becoming as important as the model itself.

© Copyright Tolaga Research, 2026

Tolaga AI Infrastructure Simulator



LEARN MORE

About the Author: Phil Marshall, PhD

Phil Marshall's work in AI began more than 30 years ago with Kohonen maps, discrete neural networks, machine learning, and fuzzy logic. More recently, his work has targeted AI-enabled solutions for the automotive, healthcare, manufacturing, and smart city sectors. Over the past decade, he has applied natural language processing and large language models to analyze online content and generate distinctive, data-driven industry research insights.

Marshall also brings deep experience in telecom, wireless, mobile, and networking technologies, fields he has worked in since the early 1990s. More recently, he has combined this infrastructure background with advanced software and modeling capabilities to develop Tolaga's AI Infrastructure Simulator, which is used to analyze the performance, cost, and scalability of emerging AI architectures.

Marshall is a dual citizen of the United States and New Zealand and currently resides in Auckland, New Zealand.

Contact: philip.marshall@tolaga.com

© Copyright Tolaga Research, 2026