Trading Precision for Efficiency with Quantization

The Economics of Lower-Precision Inference

Author: Phil Marshall, PhD

Publish Date: June, 2026

Executive Summary

Quantization lowers AI inference costs by reducing the numerical precision used to represent model weights. This decreases model-weight memory requirements and data movement, potentially allowing models to run on fewer accelerators or serve more requests within the same infrastructure.

Relative to the 16-bit baseline used in the simulation, INT8 reduces model-weight memory requirements by approximately 50%, while improving simulated cost per token by around 18–20%. INT4 reduces model-weight memory by approximately 70–75% and improves simulated cost per token by around 31–33%, but introduces a larger and more variable capability penalty, particularly for complex workloads.

The central trade-off is clear: efficiency gains grow as precision falls, while quality degradation becomes more pronounced and increasingly workload-dependent. The optimal precision is the lowest level that continues to meet the application's quality, latency, reliability, and cost requirements.

© Copyright Tolaga Research, 2026

Why Quantization Matters

AI inference is becoming a major infrastructure and operating-cost challenge. As models grow larger and usage expands, the memory required to store and move model weights constrains deployment density, throughput, and cost efficiency.

Quantization addresses this by reducing numerical precision, typically from 16-bit to 8-bit or 4-bit, without changing the underlying model architecture. The economic benefits can be substantial: larger models may fit on fewer accelerators, while the resulting memory headroom can support larger batches, longer contexts, or more concurrent requests.

The trade-off is that aggressive compression can reduce quality, especially on complex reasoning, coding, math, long-context, or subtle instruction-following tasks.

How Modern Quantization Methods Work

Modern post-training quantization methods, including GPTQ and AWQ, reduce precision intelligently rather than treating all weights equally. In this note, INT8 and INT4 primarily refer to low-precision model-weight representations and do not necessarily imply that all inference calculations use integer arithmetic.

GPTQ quantizes a model layer by layer, using second-order information to estimate how weight changes affect layer outputs and selecting quantized values that minimize reconstruction error. This can preserve much of the original model's performance at 4-bit precision and, under suitable configurations, at 3-bit precision.

AWQ uses activation data from a calibration set to identify and protect the most influential weight channels through rescaling, helping preserve accuracy in a hardware-friendly,

uniform low-precision format.

Other methods include SmoothQuant, which supports 8-bit quantization of both weights and activations, and approaches such as SpQR, AQLM, and QuIP, which target more aggressive low-bit compression.

GPTQ and AWQ are primarily associated with lowbit, weight-only quantization, particularly 4-bit deployment.

© Copyright Tolaga Research, 2026

A Healthcare Analogy

Quantization is like using a concise patient summary instead of the full medical record for every consultation. For most cases the summary suffices and dramatically reduces overhead; complex cases may still need the complete record. Moderate quantization preserves most capabilities, while aggressive compression increases the risk of losing critical information for demanding tasks.

Simulation Setup and Quantization Results

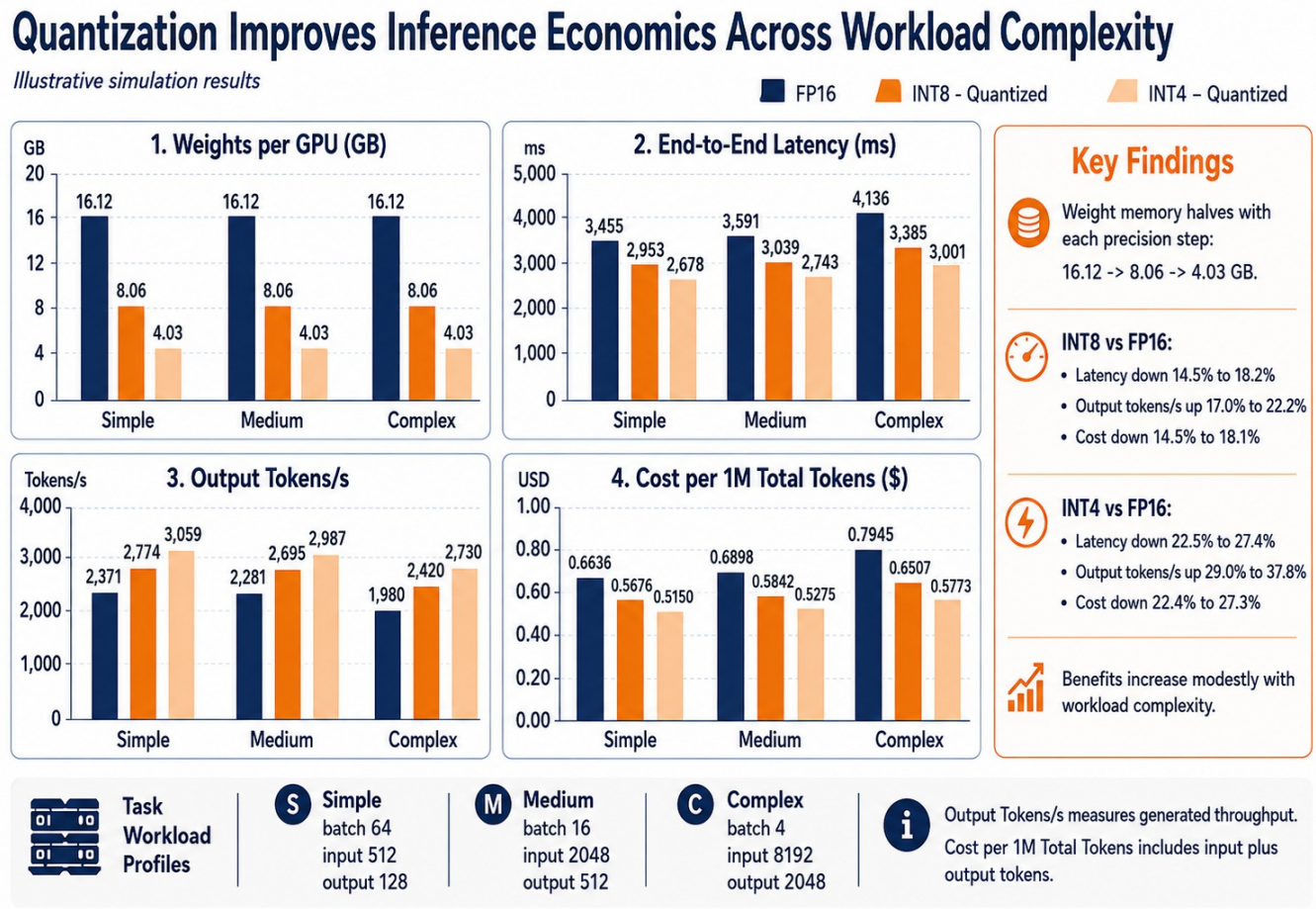
A 70B parameter inference workload was simulated with three task profiles: simple, medium, and complex. The simple profile used batch 64 with 512 input tokens and 128 output tokens; the medium profile used batch 16 with 2,048 input tokens and 512 output tokens; and the complex profile used batch 4 with 8,192 input tokens and 2,048 output tokens. Each workload was evaluated using a 16-bit baseline and two lower-precision model-weight formats: INT8 and INT4.

The results show that quantization materially improves inference economics by reducing model-weight memory and associated data movement. Weights per GPU fall from 16.12 GB in FP16 to 8.06 GB with INT8 and 4.03 GB with INT4. This represents an approximate 50% reduction at INT8 and 75% reduction at INT4.

Across the simulated workloads, INT8 reduces end-to-end latency by roughly 14.5% to 18.2%, increases output throughput by approximately 17.0% to 22.2%, and lowers cost per 1M total tokens by about 14.5% to 18.1%. INT4 delivers larger gains, reducing latency by roughly 22.5% to 27.4%, increasing output throughput by about 29.0% to 37.8%, and lowering cost per 1M total tokens by about 22.4% to 27.3%.

The benefits increase modestly with workload complexity because larger prompts and longer generations place greater pressure on memory movement and serving efficiency. However, the simulation focuses on infrastructure economics and does not directly model quality loss. In production, the economic gain from INT4 must be weighed against the higher risk of capability degradation, especially for complex reasoning, coding, math, long-context, and subtle instruction-following tasks.

© Copyright Tolaga Research, 2026



The Trade-Off is Nonlinear

The infrastructure benefits of quantization are relatively predictable, especially for model-weight memory, which generally falls in proportion to the reduction in precision. The realized latency, throughput, and cost gains are less exact because they depend on hardware support, memory bandwidth, kernel efficiency, batching, and serving-stack overhead.

Capability degradation is less predictable. It can increase nonlinearly at lower precision and varies significantly by task, model, context length, language, and quantization method. Moving from 16-bit to 8-bit typically delivers strong savings with limited downside. Moving from 8-bit to 4-bit provides further savings, but can introduce meaningfully greater quality risk, especially for long-context, multilingual, or complex reasoning workloads.

© Copyright Tolaga Research, 2026

Implications for AI Infrastructure

Quantization reduces the memory capacity and bandwidth required per model instance. This enables:

- Fewer accelerators needed
- Higher deployment density and better utilization (more room for KV cache and batching)
- Improved latency/throughput in memory-bound workloads
- Economic viability for larger models

Implications for Investors and Infrastructure Providers

Quantization weakens the direct link between parameter count and hardware demand. Demand forecasts based solely on model size or peak FLOPs may overstate accelerator needs. Platforms that deliver strong low-precision performance, memory bandwidth, and software support are better positioned than those offering raw compute alone.

Operational Considerations

Quantization should be validated on representative production workloads, especially difficult, high-risk, or low-frequency cases, rather than relying solely on aggregate benchmarks. Deployment should include monitoring for output quality, consistency, and failure modes, together with periodic reassessment as models, workloads, and serving software change.

There is no universal precision sweet spot. Although INT8 offers the strongest overall balance in this simulation, the optimal choice depends on the model, workload, quantization method, hardware support, and consequences of error. A tiered strategy may be preferable, routing routine requests to a more aggressively quantized model and complex or high-stakes requests to a higher-precision version.

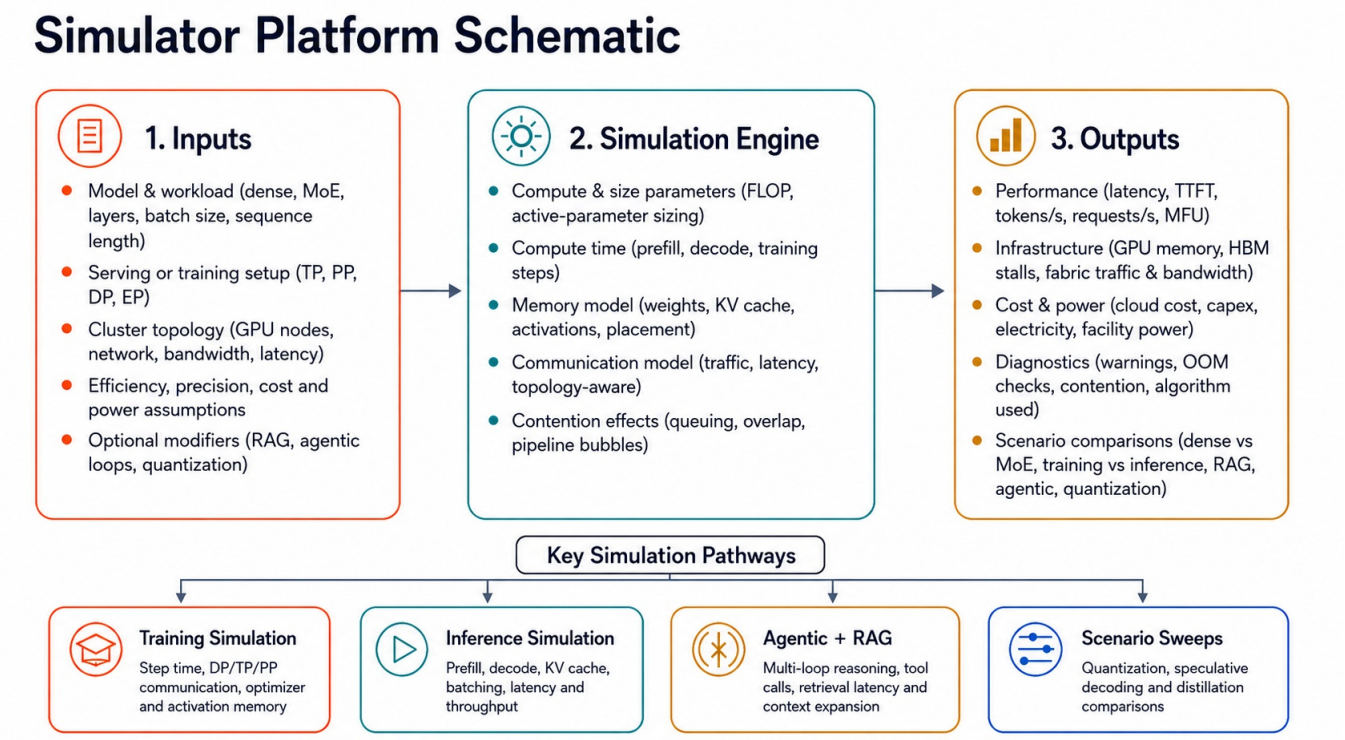
Conclusion

Quantization is one of the most practical and immediately available levers for improving AI inference economics. INT8 can deliver substantial savings with limited quality risk, while INT4 offers deeper reductions at higher (and more variable) capability cost.

The efficiency gains are relatively predictable; the capability effects are not. Quantization should therefore be treated as a workload-specific optimization. The optimal precision is the lowest level that continues to meet the application's quality, latency, reliability, and cost requirements.

© 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