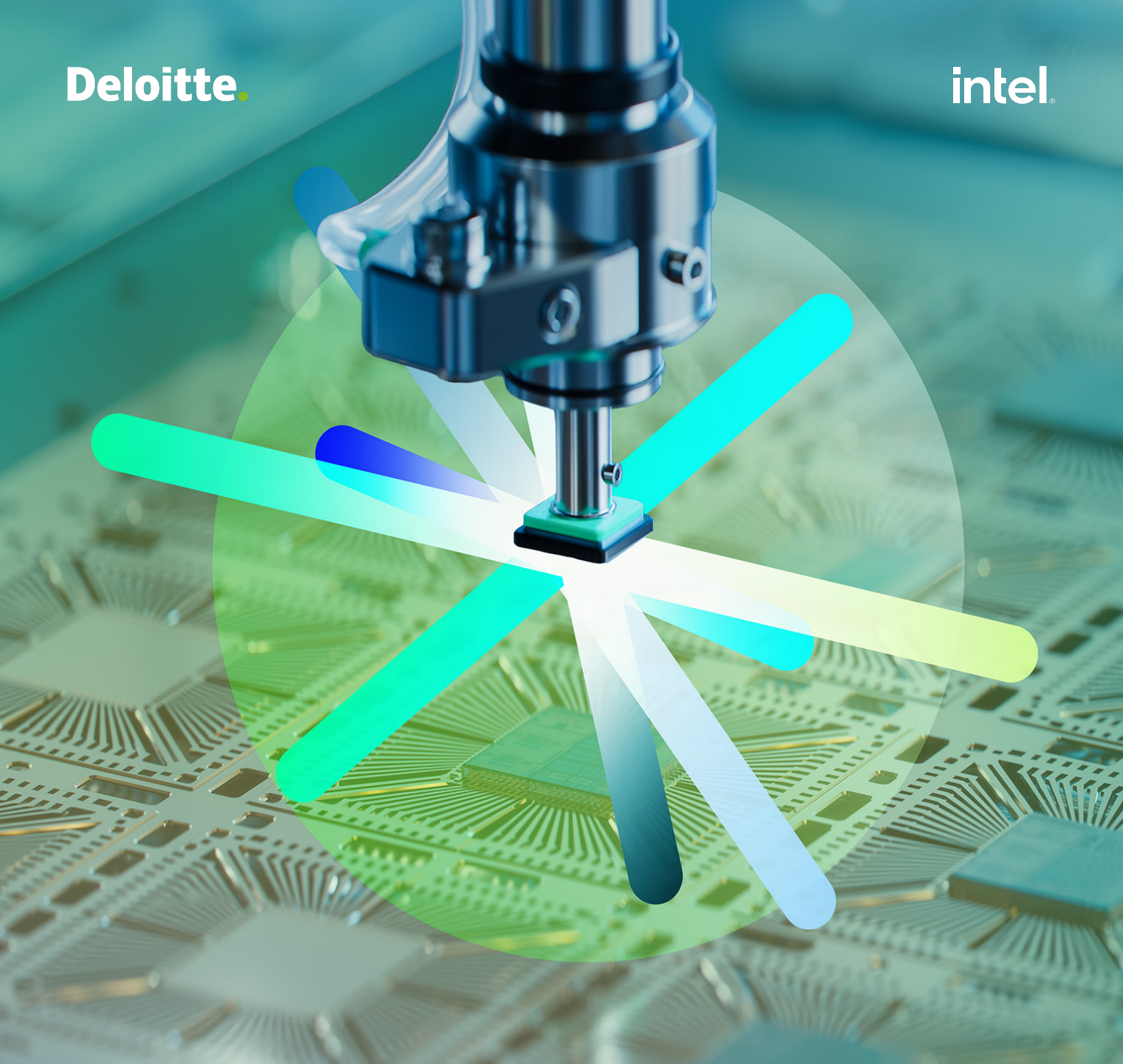


Deloitte.

intel.



*ENTERPRISE AGENTIC AI ON
GOOGLE CLOUD USING INTEL®
XEON® 6 PROCESSORS*

**PRODUCTION VALIDATION OF CPU-BASED INFERENCE
FOR ENTERPRISE WORKLOADS**

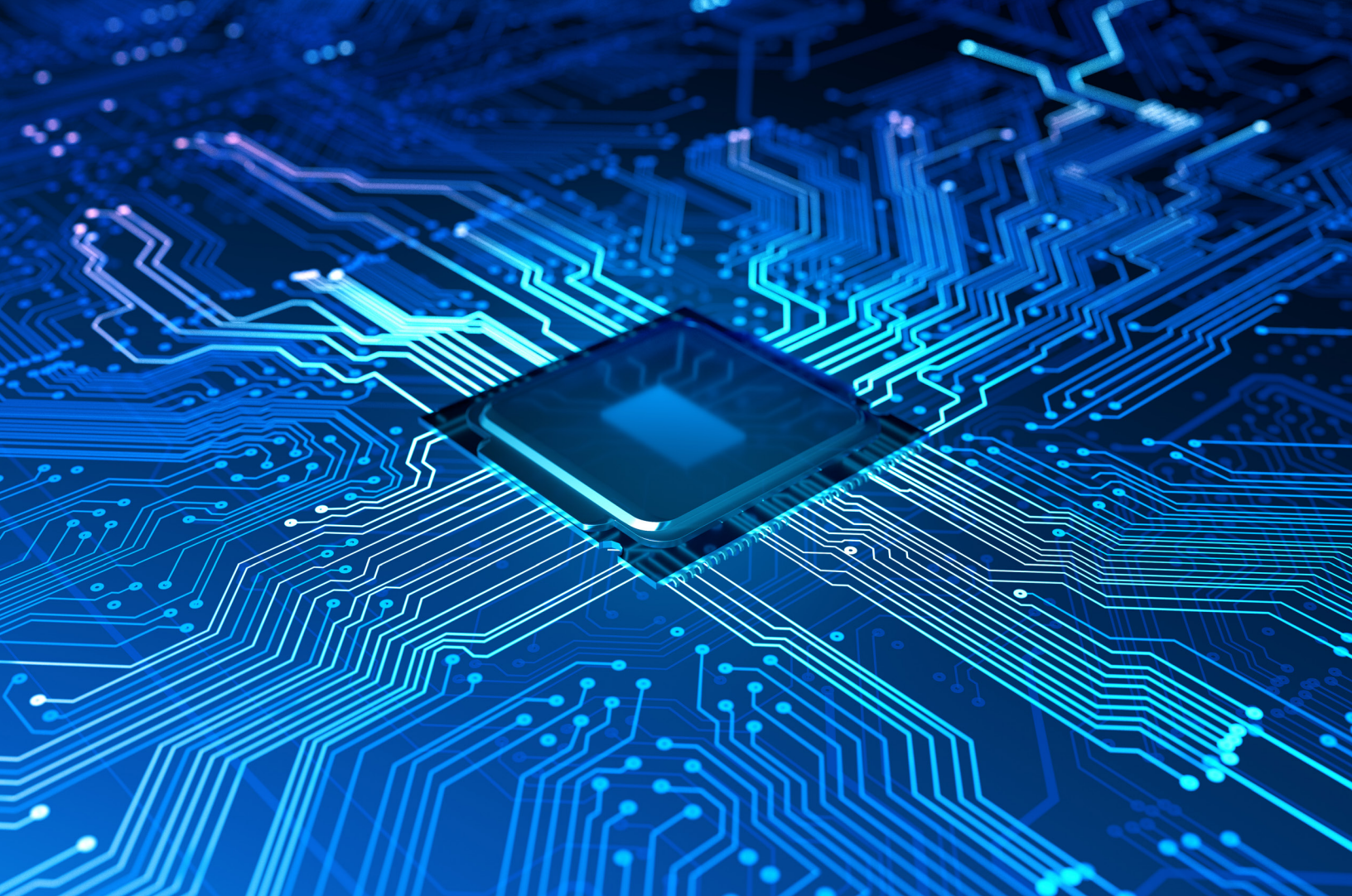


TABLE OF CONTENTS

Executive summary	03
1.0 Why consider CPU for inferencing now	04
2.0 Workloads tested: Deep dives on two use cases	05
3.0 Benchmarking approach (minimum reproducible details)	07
4.0 Results and analysis: Validated performance envelope	09
5.0 Economics model	11
6.0 Unit economics and 3-year TCO scenarios	14
7.0 What to deploy on GCP (recommended specs and operating model)	16
Conclusion	17

EXECUTIVE SUMMARY

As enterprises move GenAI from pilots into production, infrastructure decisions increasingly determine whether AI systems can scale in a controlled, cost-effective manner. While model quality is rarely the limiting factor, many organizations struggle to operationalize AI where data governance, analytics integration, and predictable economics are equally critical.

Google Cloud is increasingly used for **analytics-led agentic AI**, where intelligence is triggered by insights generated inside governed data platforms rather than by real-time user interaction alone. In these environments, AI agents are often embedded directly into analytical workflows, processing large volumes of structured and semi-structured data, orchestrating multi-step analysis, and producing decisions that must remain explainable, auditable, and cost-predictable.

This paper validates that **CPU-first small language model (SLM) inference on Intel® Xeon® 6 processor ("Xeon 6") is well suited to analytics-centric workloads on Google Cloud**. Deloitte and Intel conducted production-representative benchmarking on Google Cloud C4 instances using two enterprise workloads - **Financial Intelligent Document Processing** and **Predictive Turnover Risk** - to evaluate what CPU-based inference can handle, under what conditions, and how its economics behave when sized to a validated concurrency envelope.

The results show that modern Xeon 6 CPUs can support **high-volume, sequential-input enterprise AI workloads** in cases where concurrency, data proximity, and predictable throughput matter more than ultra-low response times. By translating measured performance into monthly processing capacity, unit cost, and three-year total cost of ownership, this paper provides decision-grade clarity for CIOs, CTOs, and CFOs evaluating how to scale AI within governed data environments.

For analytics-driven agentic workflows and document-centric processing on Google Cloud, **CPU-based SLM inference on Intel Xeon 6 is fit-for-purpose**. It offers fixed, capacity-based economics and on-demand availability that simplify deployment and budgeting as AI moves beyond pilots.

WHY CONSIDER CPU FOR INFERENCE NOW

■ Enterprise AI infrastructure is becoming increasingly heterogeneous.

Production AI systems are increasingly decomposed into many narrowly scoped tasks: **classification, extraction, routing, summarization, and agentic tool-using steps** that do not necessarily require large, general-purpose models or sub-second response times.

As organizations scale these systems, **infrastructure economics** and **operational predictability** overtake **real-time responsiveness** and **burst throughput** as the primary, dominant enterprise constraints. Many enterprise workloads benefit from predictable and fixed-cost capacity that can be planned, shared, and scaled linearly.

Two trends expand the viable CPU footprint for inference:

Aa

The rise of small language models (SLMs)

Enterprise AI workloads increasingly rely on models in the ~4B-8B parameter range that are task-fit for automation and analytical workflows.



Advances in quantization and CPU architecture

INT4 and INT8 quantization significantly reduce memory footprint and increase effective throughput for inference workloads without materially degrading task accuracy. At the same time, Intel Xeon 6 introduces architectural capabilities that directly benefit transformer inference. Intel® Advanced Matrix Extensions (Intel® AMX) provide hardware acceleration for the matrix operations that dominate inference. Higher memory bandwidth and larger cache capacity reduce bottlenecks during response generation when the model produces output token by token and repeatedly accesses model weights and context.

As a result of the combination of these two trends, inference workloads that were previously impractical on CPUs due to cost or throughput constraints are now viable when properly sized to concurrency and workload shape.

Against this backdrop, the question for enterprises becomes:

■ Can CPU-based inference deliver production-grade performance with predictable economics for common enterprise workloads?

This paper answers that question for a specific, well-defined set of workloads. Using production-representative testing on GCP, we benchmark Intel Xeon 6 processors on C4 instances across **two enterprise use cases** and translate measured performance into capacity, unit economics, and three-year TCO. *The goal of the paper is to provide actionable guidance for organizations deciding how to deploy AI sustainably at scale.*

WORKLOADS TESTED: DEEP DIVES ON TWO USE CASES

The following two use cases were selected for full end-to-end performance and cost validation, representing CPU-appropriate workload shapes.

2.1 DEEP DIVE A: FINANCIAL INTELLIGENT DOCUMENT PROCESSING

Critical deal intelligence is often trapped in unstructured financial documents. Analysts spend significant time manually extracting information from filings that include dense tables, footnotes, and regulatory formatting.



**Business
problem**

An AI document processing pipeline that ingests financial filings, extracts structured data from both narrative sections and tables, and outputs normalized fields for downstream analysis.



Solution

Each document involves **long input content and short, structured output**, with workflows typically triggering multiple inference steps per document (for example, extraction, validation, normalization, and summarization). This pattern reflects an **input-heavy, prefill-dominant workflow** and a **tolerance for multi-second processing times**, as documents are submitted asynchronously and results are consumed downstream. Together, these characteristics align well with CPU-based inference, where throughput, predictable latency, and cost efficiency matter more than real-time responsiveness.



**Workload
profile**

Document processing is typically batch-oriented and tolerant of multi-second latency because users submit documents asynchronously and retrieve results downstream. The workload is dominated by large document ingestion and analysis rather than real-time response, which aligns well with CPU-based inference.



**Why this
use-case is
a CPU fit**

2.2 DEEP DIVE B: PREDICTIVE TURNOVER RISK

HR leaders need actionable insights into workforce retention, but relevant data is distributed across multiple enterprise systems. HR teams often spend significant time gathering and reconciling information before meaningful analysis can begin, and in some cases, it becomes too late to take the necessary action to retain an employee.

An agentic system that accepts natural language queries about retention risk, retrieves relevant information from enterprise data sources, and synthesizes findings with supporting evidence.

Each user query has a **short input context** and produces **substantive output**. Each interaction fans out into multiple inference steps, such as routing, retrieval, synthesis, and formatting as the agent orchestrates analysis across systems. This pattern reflects the **step-by-step nature of analytical agent workflows** and a **moderate tolerance for multi-second processing times**, which are acceptable in decision-support scenarios where users consume results as generated text. Together, these characteristics align well with CPU-based inference, where predictable per-step latency and concurrency matter more than single-shot response speed.

Agentic workflows generate many discrete inference calls and emphasize predictable first-token latency and smooth token streaming over ultra-low single-shot response times. Users tolerate brief initial delay (approximately 1 to 4 seconds) and consume responses as text streams at human reading speed.

2.3 TECHNICAL STACK

To illustrate the architectural consistency across both workloads and the points where they diverge, the table below summarizes the full technical stack used in both use cases.

Table 1: Technical stack overview for validated workloads

Component	Financial IDP	Predictive turnover risk
Compute	C4-standard-96-Issd (General Purpose Machine Family: Intel Xeon 6)	C4-standard-96-Issd (General Purpose Machine Family: Intel Xeon 6)
Inference	vLLM serving Llama 3.1-8B, Qwen 3-4B both AWQ models	vLLM serving Llama 3.1-8B, Qwen 3-4B both AWQ models
Orchestration	Google Kubernetes Engine (GKE)	Google Kubernetes Engine (GKE)
Data sources	Github Dataset 1,147 question-answer examples for numerical reasoning over financial reports	Source-1 1,470 records and 35 feature columns capturing demographic, job, and performance attributes
Storage	BigQuery, DocumentAI (temporary document storage)	BigQuery, Vertex AI Store

BENCHMARKING APPROACH (MINIMUM REPRODUCIBLE DETAILS)

3.1 TEST CONFIGURATION

All tests used Ubuntu 24.04.3 LTS with vLLM 0.11.0, with Hyper-Threading enabled to maximize concurrent request handling. The models tested were Llama 3.1-8B and Qwen 3-4B, which were both INT4-quantized (AWQ). The load was generated by running many long-lived POST requests that each execute the full SLM inference cycle, keeping CPU, memory, and KV-cache under constant contention and exposing queuing and scheduling behavior without needing high queries per second. Concurrency was then scaled from 1 to 128 concurrent users to characterize behavior as concurrency increases, as well as to identify predictable operating ranges and tail-latency breakpoints. The study also compared C4 (Xeon 6) against the prior-generation C3 under identical software conditions to isolate generational hardware effects.

Table 2: Minimal reproducible setup
(pricing and scope included)

Item	Value used in validation
 Cloud instance	Google Cloud c4-standard-96-lssd (Intel Xeon 6), Linux
 On-demand price basis	\$6.06 per hour (Oregon US-west1, Dec 25)
 Inference engine	vLLM 0.11.0 (CPU)
 Models (deep dives)	Llama 3.1-8B and Qwen 3-4B, INT4 AWQ quantized
 Context length setting	32,768 tokens (Financial IDP); 8,192 tokens (Predictive Turnover Risk)
 Concurrency sweep	1 to 128 concurrent users; characterization at 32 and 64
 Metrics captured	Throughput (req/s, tok/s), TTFT (P99), inter-token latency (median/P50), CPU and memory utilization
 Economics scope	Inference-node costs only; excludes orchestration, storage, data access, monitoring, preprocessing

3.2 METHODOLOGY NOTES

Each concurrency level was executed long enough to reach steady operating conditions after warm-up, reflecting how inference runs in production environments. Identical software stacks and serving configurations were used across all runs so that observed differences reflect underlying hardware behavior rather than tuning or configuration variance.

Concurrency testing was used to identify a practical operating threshold where latency begins to increase more noticeably as load rises. Reported operating points are intended to reflect realistic enterprise usage patterns rather than peak or stress-test performance. Public datasets were used to support reproducibility, including SEC EDGAR 10-K filings

for the Financial Intelligent Document Processing use case and a synthetic employee dataset from the IBM AIF360 repository for the Predictive Turnover Risk use case. All cost calculations assume a fixed-cost, always-on inference node model, with unit cost varying based on achieved utilization as summarized in Table 5.



RESULTS AND ANALYSIS: VALIDATED PERFORMANCE ENVELOPE

CPU inference behavior under load is best understood **by how latency and throughput evolve** as concurrency increases. Across the tested workloads, C4 instances based on Intel Xeon 6 showed predictable performance at moderate concurrency, followed by rising latency as system pressure increases.

4.1 FINANCIAL INTELLIGENT DOCUMENT PROCESSING: PERFORMANCE RESULTS (C4)

Table 3 shows latency and output throughput for the financial document processing workload at planning-relevant concurrency levels. As concurrency increased, aggregate throughput scales while time to first token increased, reflecting increased system contention under sustained load.

Table 3: Financial intelligent document processing: Performance results

Model	Concurrency	TTFT median (ms)	TTFT P99 (ms)	ITL median (ms)	ITL P99 (ms)	Throughput (requests/s)
Qwen 3-4B (INT4 AWQ)	32	3,130	4,285	255	294	0.85
Qwen 3-4B (INT4 AWQ)	64	5,642	8,040	400	442	1.1
Llama 3.1-8B (INT4 AWQ)	32	3,388	6,202	301	350	0.72
Llama 3.1-8B (INT4 AWQ)	64	7,691	12,475	498	1,697	0.85

4.2 PREDICTIVE TURNOVER RISK: PERFORMANCE RESULTS (C4)

Table 4 shows performance metrics for the predictive turnover workload, which consists of multiple sequential inference steps. At moderate concurrency, first-token latency and token streaming remained consistent. As concurrency increased, latency rose while aggregate output throughput continued to scale.

Table 4: Predictive turnover risk: Performance results

Model	Concurrency	TTFT median (ms)	TTFT P99 (ms)	ITL median (ms)	ITL P99 (ms)	Throughput (requests/s)
Qwen 3-4B (INT4 AWQ)	32	3,196	3,894	61	112	2.01
Qwen 3-4B (INT4 AWQ)	64	4,199	4,598	95	132	2.74
Llama 3.1-8B (INT4 AWQ)	32	1,027	1,129	60	91	18.64
Llama 3.1-8B (INT4 AWQ)	64	2,506	2,514	100	323	18.58

4.3 PERFORMANCE CHARACTERISTICS

Across workloads, CPU-based inference on Intel Xeon 6 demonstrates a **well-defined and predictable performance envelope** as concurrency increases. Throughput scales smoothly at moderate concurrency levels, while latency rises gradually as system contention increases under sustained load.

Model choice influences performance efficiency within this envelope.	For the document processing workload, Qwen 3-4B consistently delivers lower first-token latency and higher aggregate token throughput than Llama 3.1-8B at comparable concurrency levels. In agentic, multi-step workflows, both models exhibit stable token streaming at moderate concurrency, with latency increasing in a controlled manner as concurrency rises.
Generational improvements materially shift the operating range.	When evaluated under identical software configurations, C4 instances deliver meaningfully higher throughput and lower latency than prior-generation C3, reflecting architectural gains from AMX acceleration, increased memory bandwidth, and overall platform efficiency. These improvements expand the usable concurrency range before latency degradation becomes material.
Concurrency defines practical operating boundaries.	Performance remains most stable and predictable up to approximately 32 concurrent users per instance , where throughput scaling is efficient and latency variability remains limited. At 64 concurrent users , instances continue to scale throughput but enter a stretch regime characterized by higher and more variable latency. This boundary provides a clear basis for defining capacity tiers and informs the cost and scaling analysis in Section 5.

ECONOMICS MODEL

Our economics model is intentionally simple and aligns with how most enterprises budget infrastructure. We treat an inference node as an always-on resource with a fixed hourly price.

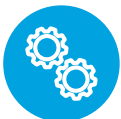
Because the infrastructure cost is fixed, the unit cost depends on utilization. As more validated throughput is delivered on the same instance, the cost per 1,000 inferences declines.

To apply the model, we begin with measured throughput at a validated concurrency level and convert it into monthly capacity. We then calculate

cost per 1,000 inferences from that capacity. Next, we extend unit costs to a three-year total cost of ownership and use the resulting economics to inform instance sizing.

All calculations are grounded in the validation environment and use a single set of explicit and consistent assumptions.

5.1 PRICING AND TIME BASE ASSUMPTIONS



Instance

Google Cloud c4-standard-96-lssd (Intel Xeon 6)



Monthly timebase

30 days × 24 hours = 720 hours



Pricing basis

\$6.06 per hour (on-demand, Oregon us-west1, January 2026)



Evaluation period

36 months (three years)



Operating model

Always-on (24×7)



Scope

Inference-node compute costs only; orchestration, storage, data access, and downstream services are excluded

Under these assumptions, the fixed monthly cost of a single inference node is:

$$\mathbf{\$6.06 \times 720 = \$4,363 \text{ per month (rounded)}}$$

The corresponding three-year total cost of ownership for one always-on node is:

$$\mathbf{\$4,363 \times 36 = \$157,075 \text{ (rounded)}}$$



5.2 HOW PERFORMANCE TRANSLATES INTO CAPACITY AND COST

Benchmark results report inference throughput in requests per second at validated concurrency points. Translating this into business capacity and unit cost requires two additional factors:



First, a workflow fan-out:

the number of inference requests required to complete one business outcome (for example, a document or an agentic query)



Second, a timebase conversion:

Converting per-second throughput into monthly volume using a 30-day operating month (2,592,000 seconds)

The same formulas are applied consistently across workloads: Monthly capacity = (requests/sec ÷ workflow fan-out) × 2,592,000, then the cost per 1,000 units = (monthly instance cost ÷ monthly units delivered) × 1,000.

This walkthrough shows how the Financial IDP use case economics in Table 5 are computed from the validated throughput point:

Step



2

3

4

5

This walkthrough shows how the Predictive Turnover Risk use case economics in Table 5 are computed from the validated throughput point. Because agentic workflows involve multiple inference calls per interaction, workflow fan-out is the key translation from requests/sec to queries/month.

13

UNIT ECONOMICS AND 3-YEAR TCO SCENARIOS

Table 5 consolidates unit economics at two representative utilization points and at full validated capacity. This is the table to use for first-order budgeting; it connects performance to capacity to unit cost, then to 3-year TCO.

Table 5: Unit economics at representative demand levels

Workload	Demand (units/month)	Utilization vs. validated capacity	\$/1K units	Monthly node cost	3-year node TCO
Financial IDP	150,000 documents	34.0%	\$29.08	\$4,363	\$157,075
Financial IDP	300,000 documents	68.0%	\$14.54	\$4,363	\$157,075
Financial IDP (max validated)	441,000 documents	100%	\$9.90	\$4,363	\$157,075
Predictive Turnover Risk	250,000 queries	33.6%	\$17.45	\$4,363	\$157,075
Predictive Turnover Risk	500,000 queries	67.1%	\$8.73	\$4,363	\$157,075
Predictive Turnover Risk (max validated)	745,000 queries	100%	\$5.87	\$4,363	\$157,075

Request volume note: total inference requests per month = demand × workflow fan-out (fan-out is five requests per document for Financial IDP and seven requests per query for Predictive Turnover Risk).

6.1 HOW TO TRANSLATE UNIT ECONOMICS INTO NODE COUNT

The unit economics above are based on a single, always-on inference node. In production deployments, total node count is determined by peak concurrent demand rather than average volume.

A practical sizing approach is to start from observed peak concurrent workflows and divide by the sustainable concurrency supported per node. As demand grows beyond this range, capacity is added by deploying additional nodes.

Initial sizing should be validated through a pilot using production-representative data at two to three times expected peak concurrency, with

attention to tail latency under sustained load. When sustained demand exceeds the validated operating range of a single node, performance remains most predictable when capacity is added horizontally rather than by pushing additional load onto an individual instance.



WHAT TO DEPLOY ON GCP (RECOMMENDED SPECS AND OPERATING MODEL)

To deploy on GCP, the recommended instance family is C4 (Intel Xeon 6). The benchmarks and economics in this paper are anchored on the c4-standard-96-lssd, which serves as a practical starting point for enterprise teams standardizing a CPU inference node type.

7.1 SIZING HEURISTIC (BASED ON SUSTAINABLE CONCURRENCY)

CPU-based inference is well suited to workloads with **moderate latency tolerance and predictable concurrency**, where efficiency and cost control matter more than ultra-low response times. In practice, this includes workloads that can accept **3-10 seconds of initial response**, rely on **quantized models up to ~10B parameters**, and operate on high-volume classification, extraction, or structured-output patterns.

These workloads typically use context windows up to 32K tokens (or can be effectively chunked), exhibit predictable concurrency below ~128 users per instance, and scale cleanly through **horizontal expansion** rather than vertical specialization. Within this envelope, CPU-based inference delivers stable performance and repeatable, capacity-based economics.



CONCLUSION

CPU-based inference is a **production-viable option** for a defined class of enterprise AI workloads. For small, quantized language models supporting sequential processing and agentic orchestration, Intel Xeon 6 delivers consistent performance with predictable, node-based economics.

Across evaluated workloads, throughput remains stable within a defined concurrency range, allowing capacity, unit cost, and multi-year TCO to be planned deterministically. This shifts inference infrastructure from a variable, usage-priced model to a **capacity-planned operational asset**.

As organizations scale AI from pilots to operational systems, these results establish a clear boundary where modern CPUs are the economically rational choice for a defined class of enterprise workloads. It also

compels enterprises to ask **which workloads belong where**. For analytics-driven agentic workflows and document processing, the results of this paper provide a concrete answer.

Intel Xeon 6 on GCP is available today, and it provides an **immediately available foundation for deploying production AI** with controlled cost and transparent scaling behavior.

AUTHORS



Shakir Rizvi

Senior Manager, Strategy, Growth, and Transformation
Deloitte Consulting LLP
shrivzi@deloitte.com



Osazee Paul

Manager, AI & Engineering
Deloitte Consulting LLP
opaul@deloitte.com



Jose Medina

Manager, Strategy, Growth, and Transformation
Deloitte Consulting LLP
jomedina@deloitte.com



Pratyush Kumar

Senior Consultant, Strategy, Growth, and Transformation
Deloitte Consulting LLP
pratyushkumar3@deloitte.com

CONTRIBUTORS

Notices & Disclaimers

Intel, the Intel logo, Intel Xeon, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Other names and brands may be claimed as the property of others.

This document is provided for informational purposes only. It summarizes test results and an economics model for CPU inference nodes. It does not include costs for storage, networking, orchestration, monitoring, or data services. All testing, workload design, benchmarking, and economic analysis presented in this paper were conducted using the configurations and methodologies described herein. Results may vary.

About Deloitte

Deloitte refers to one or more of Deloitte Touche Tohmatsu Limited, a UK private company limited by guarantee, and its network of member firms, each of which is a legally separate and independent entity. Please see www.deloitte.com/about for a detailed description of the legal structure of Deloitte Touche Tohmatsu Limited and its member firms. Please see www.deloitte.com/us/about for a detailed description of the legal structure of Deloitte LLP and its subsidiaries. Certain services may not be available to attest clients under the rules and regulations of public accounting.

This publication contains general information only and Deloitte is not, by means of this publication, rendering accounting, business, financial, investment, legal, tax, or other professional advice or services. This publication is not a substitute for such professional advice or services, nor should it be used as a basis for any decision or action that may affect your business. Before making any decision or taking any action that may affect your business, you should consult a qualified professional advisor. Deloitte shall not be responsible for any loss sustained by any person who relies on this publication.