

Investigación y Cultura Académica

**Paradigmas computacionales y eficiencia
arquitectónica en sistemas de alto rendimiento:
una revisión sistemática**

**Computing paradigms and architectural
efficiency in high-performance systems: a
systematic review**

**Paradigmas computacionais e eficiência
arquitetônica em sistemas de alto desempenho:
uma revisão sistemática**



César Andrés Mero Baquerizo
Universidad de Guayaquil
<https://orcid.org/0009-0001-1347-4219>
cesar.merob@ug.edu.ec



Víctor Hugo León Quiñónez
Universidad de Guayaquil
<https://orcid.org/0009-0003-3685-0381>
victor.leonq@ug.edu.ec



Ciencias de la Educación
Artículo de Revisión

Cómo citar
este artículo:

Mero Baquerizo, C. A., & León Quiñónez, V. H. (2026). *Paradigmas computacionales y eficiencia arquitectónica en sistemas de alto rendimiento: una revisión sistemática*. *Investigación y Cultura Académica*, 2(2), 202–240.
<https://doi.org/10.5281/zenodo.20632782>

Paradigmas computacionales y eficiencia arquitectónica en sistemas de alto rendimiento: una revisión sistemática

Resumen

Los paradigmas computacionales han evolucionado como respuesta a restricciones crecientes de procesamiento, memoria, comunicación, almacenamiento, confiabilidad, escalabilidad y volumen de datos. El presente artículo tuvo como objetivo analizar la relación entre paradigmas computacionales y eficiencia arquitectónica en sistemas de alto rendimiento, considerando la evolución epistemológica de la computación, las arquitecturas paralelas y distribuidas, las redes de interconexión, la jerarquía de memoria, la computación en la nube, los modelos post-nube, la gestión de grandes datos y la heterogeneidad del hardware. La investigación se desarrolló bajo un enfoque cualitativo, documental y de revisión sistemática con alcance integrativo, siguiendo la lógica PRISMA para la identificación, cribado, elegibilidad e inclusión de documentos. El corpus final estuvo conformado por 39 documentos científicos y técnicos relacionados con ciencia de la computación, arquitectura de computadores, high-performance computing, cloud computing, edge computing, big data, memoria, caché, DRAM, GPU, APU y redes de interconexión. Los resultados evidencian que la eficiencia arquitectónica no depende únicamente de la velocidad del procesador, sino de la articulación entre cómputo, memoria, comunicación, datos, software, energía y confiabilidad. Se concluye que los paradigmas computacionales contemporáneos avanzan hacia modelos híbridos, distribuidos, heterogéneos, confiables y orientados a datos, capaces de responder a los desafíos de la ciencia intensiva en información, la inteligencia artificial y los sistemas computacionales emergentes.

Palabras clave: paradigmas computacionales; eficiencia arquitectónica; computación de alto rendimiento; memoria; sistemas distribuidos.

Private commercialization of customs data and its relationship with the competitive performance of ecuadorian importers between 2022 and 2025

Abstract

Computing paradigms have evolved in response to increasing constraints related to processing, memory, communication, storage, reliability,

scalability, and data volume. This article aimed to analyze the relationship between computing paradigms and architectural efficiency in high-performance systems, considering the epistemological evolution of computing, parallel and distributed architectures, interconnection networks, memory hierarchy, cloud computing, post-cloud models, big data management, and hardware heterogeneity. The study was conducted under a qualitative, documentary, and systematic review approach with an integrative scope, following the PRISMA logic for the identification, screening, eligibility, and inclusion of documents. The final corpus consisted of 39 scientific and technical documents related to computer science, computer architecture, high-performance computing, cloud computing, edge computing, big data, memory, cache, DRAM, GPU, APU, and interconnection networks. The findings show that architectural efficiency does not depend solely on processor speed, but on the articulation among computation, memory, communication, data, software, energy, and reliability. The study concludes that contemporary computing paradigms are moving toward hybrid, distributed, heterogeneous, reliable, and data-oriented models capable of addressing the challenges of information-intensive science, artificial intelligence, and emerging computational systems.

Keywords: computing paradigms; architectural efficiency; high-performance computing; memory; distributed systems.

Comercialização privada de dados aduaneiros e sua relação com o desempenho competitivo dos importadores equatorianos entre 2022 e 2025

Resumo

Os paradigmas computacionais evoluíram como resposta a restrições crescentes de processamento, memória, comunicação, armazenamento, confiabilidade, escalabilidade e volume de dados. O presente artigo teve como objetivo analisar a relação entre paradigmas computacionais e eficiência arquitetônica em sistemas de alto desempenho, considerando a evolução epistemológica da computação, as arquiteturas paralelas e distribuídas, as redes de interconexão, a hierarquia de memória, a computação em nuvem, os modelos pós-nuvem, a gestão de grandes dados e a heterogeneidade do hardware. A pesquisa foi desenvolvida sob uma abordagem qualitativa, documental e de revisão sistemática

com alcance integrativo, seguindo a lógica PRISMA para a identificação, triagem, elegibilidade e inclusão dos documentos. O corpus final foi composto por 39 documentos científicos e técnicos relacionados à ciência da computação, arquitetura de computadores, computação de alto desempenho, computação em nuvem, edge computing, big data, memória, cache, DRAM, GPU, APU e redes de interconexão. Os resultados evidenciam que a eficiência arquitetônica não depende apenas da velocidade do processador, mas da articulação entre computação, memória,

comunicação, dados, software, energia e confiabilidade. Conclui-se que os paradigmas computacionais contemporâneos avançam para modelos híbridos, distribuídos, heterogêneos, confiáveis e orientados a dados, capazes de responder aos desafios da ciência intensiva em informação, da inteligência artificial e dos sistemas computacionais emergentes.

Palavras-chave: paradigmas computacionais; eficiência arquitetônica; computação de alto desempenho; memória; sistemas distribuídos.

Introducción

La computación contemporánea no puede comprenderse solo como una sucesión de dispositivos, lenguajes o infraestructuras técnicas, sino como un proceso histórico de reorganización del conocimiento, las arquitecturas y las formas de resolver problemas complejos. Desde esta perspectiva, los paradigmas computacionales constituyen marcos de pensamiento y acción que orientan el diseño, ejecución y optimización de procesos de cálculo, procesamiento de información, almacenamiento, comunicación y toma de decisiones en sistemas digitales. Thinh y Tung (2026) sostienen que la ciencia de la computación ha evolucionado desde la computación mecánica hasta la inteligencia algorítmica mediante etapas de acumulación, selección y reestructuración del conocimiento. Esta lectura permite asumir que la computación avanza no solo por innovación tecnológica, sino también por cambios epistemológicos que modifican la comprensión de calcular, representar, automatizar y producir conocimiento.

En este proceso, los fundamentos teóricos han sido decisivos. La formalización lógica y matemática permitió establecer límites y posibilidades de los sistemas computacionales, mientras que las arquitecturas electrónicas dieron paso a formas de procesamiento cada vez más complejas. Vallverdú i Segura (2009) plantea que la e-Science y la epistemología computacional emergen cuando la generación de conocimiento depende de inteligencia artificial, supercomputadores, sistemas expertos, computación distribuida, bases de datos, redes, middleware y tecnologías de visualización. Así, los paradigmas computacionales actuales no solo sostienen aplicaciones técnicas, sino que transforman las prácticas científicas, los métodos de investigación y la gestión de información a gran escala.

El tránsito desde arquitecturas centralizadas hacia sistemas paralelos, distribuidos, heterogéneos y orientados a datos expresa una de las principales transformaciones de la informática moderna. La necesidad de procesar grandes volúmenes de información, reducir latencias, optimizar recursos, mejorar eficiencia energética y sostener aplicaciones científicas e industriales ha impulsado nuevos modelos computacionales. Shawish y Salama (2014) describen la computación en la nube como un paradigma que materializa la computación utilitaria mediante virtualización, servicios bajo demanda, elasticidad, acuerdos de nivel de servicio, costos variables y recursos distribuidos accesibles por Internet. No obstante, su desarrollo también ha generado desafíos

asociados con latencia, seguridad, privacidad, disponibilidad, escalabilidad, desempeño y eficiencia energética.

La emergencia de paradigmas post-nube evidencia que la centralización computacional no siempre responde a las exigencias de aplicaciones contemporáneas. Zhou, Zhang y Xiong (2017) señalan que la nube presenta limitaciones frente a dispositivos inteligentes, redes ubicuas, Internet de las cosas, realidad aumentada, vehículos no tripulados y servicios sensibles a la latencia. Por ello, modelos como fog computing, edge computing, mobile edge computing y dew computing trasladan parte de la infraestructura hacia entornos cercanos al usuario, a los sensores o a las fuentes de datos. Esta transformación no elimina la nube, sino que la complementa mediante estructuras distribuidas orientadas a reducir cuellos de botella comunicacionales.

En esta misma línea, Li, Xue, Wang, Zhang y Li (2018) explican que la computación orientada al borde responde a problemas propios de la nube, como limitaciones de ancho de banda, sobrecarga de comunicación y baja sensibilidad geográfica. Los sistemas edge y fog extienden los servicios desde los centros de datos hacia la periferia de la red para atender dispositivos inteligentes, aplicaciones ubicuas y procesamiento local. En consecuencia, la eficiencia arquitectónica ya no depende solo de la potencia de los centros de datos, sino de la capacidad para distribuir funciones computacionales entre nube, borde, dispositivos finales, redes y sistemas intermedios.

Arquitecturas de alto rendimiento y eficiencia computacional

La eficiencia arquitectónica constituye una preocupación transversal en los sistemas de alto rendimiento, porque el incremento de capacidad computacional no siempre se traduce en mejoras proporcionales de desempeño. Los sistemas modernos enfrentan restricciones asociadas con comunicación, memoria, consumo energético, transferencia de datos, confiabilidad y escalabilidad. En este sentido, la computación de alto rendimiento requiere analizar de forma integrada la arquitectura del procesador, la jerarquía de memoria, las redes de interconexión, los aceleradores, los modelos de programación y los patrones de acceso a datos. Van der Steen (2008) clasifica las arquitecturas de supercomputadores según modelos macroarquitectónicos como SIMD, MIMD, memoria compartida, memoria distribuida, ccNUMA, clústeres y

aceleradores computacionales, lo cual evidencia la diversidad de soluciones desarrolladas para responder a distintas necesidades de cálculo científico.

La computación paralela aparece como uno de los pilares de la eficiencia en sistemas de alto rendimiento. Alam y Varshney (2015) sostienen que el procesamiento paralelo se basa en la manipulación concurrente de elementos de datos y que las redes de interconexión cumplen un papel central en el desempeño de los sistemas multiprocesador. Según estos autores, la topología de red influye en parámetros como diámetro, distancia promedio entre nodos, ancho de bisección, costo y densidad de tráfico. Esta visión permite comprender que la eficiencia arquitectónica no depende solamente de la cantidad de procesadores, sino también de la forma en que estos se comunican, acceden a memoria y distribuyen el trabajo.

La necesidad de reducir latencia ha impulsado nuevas propuestas de optimización topológica. Nakao, Sakai, Hanada, Murai y Sato (2021) plantean que, en sistemas paralelos, supercomputadores y centros de datos, la latencia de las redes de interconexión afecta significativamente el rendimiento de las aplicaciones. Su propuesta basada en optimización de grafos busca reducir diámetro y longitud promedio de caminos, superando topologías convencionales como k-ary n-cube. Este enfoque muestra que la eficiencia arquitectónica puede beneficiarse de la teoría de grafos, la optimización y el diseño de redes con propiedades de mundo pequeño, especialmente cuando se busca mejorar desempeño sin incrementar excesivamente costos de hardware.

Memoria, datos y cuellos de botella en sistemas computacionales modernos

Otro eje central de la eficiencia arquitectónica se encuentra en la gestión de memoria y datos. A medida que los procesadores incrementan su capacidad de cálculo, la diferencia entre velocidad de procesamiento y velocidad de acceso a memoria se convierte en un problema crítico. Vaithianathan (2025) señala que la optimización de la jerarquía de memoria es fundamental para mejorar el desempeño en arquitecturas HPC, debido a que la brecha entre velocidad del procesador y tiempo de acceso a memoria genera ineficiencias conocidas como memory wall. Esta problemática exige estrategias como reconfiguración dinámica de caché, integración de memorias

emergentes, jerarquías sensibles al comportamiento de aplicaciones y reducción de penalizaciones por fallos de caché.

Los sistemas de gestión de grandes datos también han transformado el papel de la memoria en la arquitectura computacional. Zhang, Chen, Ooi, Tan y Zhang (2015) sostienen que el crecimiento de la capacidad de memoria principal ha impulsado el desarrollo de sistemas in-memory para gestión y procesamiento de big data, al eliminar el cuello de botella de entrada y salida en disco y permitir analítica interactiva. No obstante, estos sistemas se vuelven más sensibles a otros costos, como utilización de caché, paralelismo, concurrencia, tolerancia a fallos y consistencia. Así, la eficiencia arquitectónica depende no solo de almacenar datos en memoria, sino de diseñar estructuras capaces de aprovechar jerarquías modernas de CPU, memoria y almacenamiento.

Mazumdar, Seybold, Kritikos y Verginadis (2019) amplían esta discusión desde el ecosistema Cloud-Big Data, señalando que el crecimiento exponencial de datos presiona las tecnologías existentes y exige metodologías eficientes de almacenamiento, ubicación y migración de información entre centros de datos. Su revisión evidencia que la gestión de big data requiere considerar propiedades no funcionales como costo, rendimiento, latencia, privacidad, disponibilidad y ubicación óptima de datos. Por ello, la eficiencia arquitectónica en sistemas contemporáneos depende de la coordinación entre almacenamiento, procesamiento, redes, políticas de colocación y patrones de acceso.

En el nivel de memoria caché, los problemas de coherencia también condicionan el rendimiento de sistemas multiprocesador y distribuidos. Aguilar y Leiss (2004) proponen un esquema adaptativo de coherencia y reemplazo para sistemas de caché proxy web, orientado a reducir carga de servidores, latencia de solicitudes y tráfico de red. En otro ámbito, Alkhamisi (2022) revisa los problemas de coherencia de caché en sistemas multiprocesador, destacando que cuando varios procesadores comparten memoria, las copias de datos deben mantenerse consistentes para evitar errores de ejecución. Estos aportes permiten observar que la eficiencia arquitectónica exige coherencia, consistencia y políticas inteligentes de reemplazo tanto en sistemas web distribuidos como en arquitecturas de memoria compartida.

La confiabilidad de la memoria se ha convertido en otro desafío crítico. Patel, Kim, Hassan y Mutlu (2019) muestran que las tecnologías DRAM modernas incorporan códigos de corrección de errores en el propio chip, lo cual mejora la confiabilidad, pero también dificulta la caracterización experimental de errores al ocultar distribuciones reales. Criss et al. (2020) analizan mejoras de confiabilidad en DDR5 mediante mecanismos para acotar fallos de DRAM y reducir la sobrecarga requerida por ECC. Estos estudios permiten incorporar al análisis una dimensión adicional: los sistemas de alto rendimiento no solo deben ser rápidos y escalables, sino también confiables frente a errores físicos, fallos de memoria y limitaciones de fabricación.

Heterogeneidad, aceleración y nuevos modelos de procesamiento

La heterogeneidad arquitectónica representa una respuesta a la necesidad de mejorar rendimiento y eficiencia energética. Wilkinson y Skylaris (2013) muestran que las arquitecturas basadas en GPU pueden acelerar operaciones intensivas como transformadas rápidas de Fourier en cálculos cuánticos de primeros principios; sin embargo, también advierten que la transferencia de datos entre CPU y GPU puede convertirse en un cuello de botella significativo. Esta observación resulta relevante porque evidencia que la aceleración no depende solo del poder de cómputo del dispositivo, sino de la relación entre cálculo, memoria, transferencia y adaptación algorítmica.

Kalidas, Daga, Krommydas y Feng (2015) profundizan este problema al comparar métodos de acceso a datos en sistemas heterogéneos con GPU y APU. Su estudio muestra que los beneficios de aceleración pueden reducirse cuando las transferencias de datos por PCIe no se amortizan adecuadamente, mientras que las APU mitigan parte de este problema al integrar CPU y GPU en el mismo chip. Desde esta perspectiva, la eficiencia arquitectónica requiere analizar los caminos de acceso a datos, la ubicación de la memoria y los patrones de comunicación entre procesadores heterogéneos.

En el ámbito de bases de datos, Kozar et al. (2025) sostienen que la era de procesadores homogéneos ha cedido espacio a arquitecturas heterogéneas con GPU, MIC y FPGA, lo cual exige nuevas metodologías para ejecutar consultas de manera eficiente en distintos procesadores. Su análisis sobre procesamiento de consultas en hardware heterogéneo muestra que las diferencias arquitectónicas generan resultados variables

y requieren compilación adaptativa, distribución de cargas, optimización en tiempo de ejecución y reducción de cuellos de botella de transferencia. Este enfoque permite vincular la eficiencia arquitectónica con sistemas de información, big data y procesamiento analítico.

En conjunto, la literatura revisada permite afirmar que los paradigmas computacionales contemporáneos evolucionan hacia modelos híbridos, distribuidos, heterogéneos, orientados a datos y sensibles a restricciones físicas. La eficiencia arquitectónica ya no puede explicarse únicamente por la velocidad del procesador, sino por la interacción entre cómputo, memoria, red, almacenamiento, energía, confiabilidad, ubicación de datos y modelo de aplicación. Por ello, una revisión sistemática sobre paradigmas computacionales y eficiencia arquitectónica resulta pertinente para organizar la evidencia disponible, identificar tendencias, reconocer vacíos y proponer una lectura integradora de los sistemas de alto rendimiento en el contexto de la transformación digital, la ciencia intensiva en datos y la inteligencia algorítmica.

Dimensiones críticas de la eficiencia arquitectónica en los paradigmas computacionales

La revisión de los documentos permite reconocer que la eficiencia arquitectónica es un concepto multidimensional, pues no se limita al aumento de velocidad de procesamiento ni a la capacidad de cómputo. En los sistemas contemporáneos, depende de la interacción entre arquitectura, memoria, redes de interconexión, almacenamiento, distribución de datos, confiabilidad, aceleración heterogénea, consumo energético y modelos de aplicación. Esta complejidad explica la evolución desde arquitecturas centralizadas hacia configuraciones paralelas, distribuidas, híbridas y adaptativas, capaces de responder a problemas intensivos en datos, comunicación y procesamiento.

Desde las arquitecturas de alto rendimiento, la eficiencia está condicionada por la forma en que los procesadores se conectan, comunican y coordinan. Dally (1990) plantea que las redes de interconexión en sistemas VLSI están limitadas por la densidad de cableado, mientras que Agarwal (1991) señala que la latencia depende de retardos de conmutación, cableado, tamaño de paquetes, localidad de comunicación y

restricciones de diseño. Estos aportes muestran que la eficiencia arquitectónica articula diseño lógico, limitaciones físicas y patrones reales de comunicación.

La memoria constituye otra dimensión crítica, debido a que la velocidad de los procesadores ha crecido más rápido que el acceso eficiente a datos. Vaithianathan (2025) identifica el memory wall como una limitación central en arquitecturas HPC. De forma complementaria, Zhang, Chen, Ooi, Tan y Zhang (2015) sostienen que los sistemas in-memory reducen la dependencia del disco y permiten analítica interactiva, aunque introducen desafíos de concurrencia, consistencia, tolerancia a fallos, paralelismo y uso eficiente de caché.

El almacenamiento y la ubicación de datos también inciden en el rendimiento. Mazumdar, Seybold, Kritikos y Verginadis (2019) muestran que el ecosistema Cloud-Big Data requiere metodologías de almacenamiento, colocación y migración capaces de responder a grandes volúmenes de datos, costos, latencias y propiedades no funcionales. En sistemas distribuidos, la eficiencia no depende solo de procesar datos rápidamente, sino de ubicarlos estratégicamente, reducir transferencias innecesarias y mantener disponibilidad, privacidad y rendimiento.

La confiabilidad de la memoria y la corrección de errores son igualmente relevantes en sistemas de alto rendimiento. Patel, Kim, Hassan y Mutlu (2019) muestran que las tecnologías DRAM modernas incorporan corrección de errores en el chip, aunque pueden ocultar distribuciones reales de fallos. Criss et al. (2020) proponen estrategias para acotar fallos en DDR5, reducir sobrecargas de ECC y mejorar confiabilidad. Así, la eficiencia arquitectónica debe evaluarse también por la capacidad del sistema para sostener integridad de datos en contextos de escalamiento tecnológico.

La heterogeneidad computacional amplía este panorama al incorporar procesadores especializados, aceleradores y arquitecturas mixtas. Wilkinson y Skylaris (2013) muestran que las GPU pueden acelerar operaciones intensivas, pero la transferencia de datos entre CPU y GPU puede convertirse en cuello de botella. Kalidas, Daga, Krommydas y Feng (2015) refuerzan esta idea al evidenciar que la ubicación de los datos y los caminos de transferencia en GPU y APU influyen en rendimiento, energía y potencia. Por ello, las arquitecturas heterogéneas requieren integrar cálculo, movimiento de datos, memoria compartida y adaptación algorítmica.

La eficiencia arquitectónica también se vincula con la evolución hacia sistemas ubicuos y distribuidos. Li, Xue, Wang, Zhang y Li (2018) sostienen que la computación de borde responde a limitaciones de la nube, como cuellos de botella de ancho de banda, sobrecarga comunicacional y baja sensibilidad a la ubicación. Zhou, Zhang y Xiong (2017) describen paradigmas post-nube como fog, mobile edge y dew computing, orientados a acercar la infraestructura computacional al usuario, los dispositivos o la fuente de datos. Esta transición demuestra que la eficiencia debe distribuirse en múltiples niveles de infraestructura.

En este marco, los paradigmas computacionales pueden entenderse como respuestas históricas y técnicas a restricciones sucesivas. La computación mecánica automatizó operaciones; la computación formal definió límites del cálculo; la computación paralela respondió al agotamiento secuencial; la nube trasladó recursos a infraestructuras virtualizadas; el borde corrigió limitaciones de latencia y ubicación; y la computación heterogénea aprovechó procesadores especializados para mejorar rendimiento y eficiencia energética. Por tanto, su revisión sistemática permite comprender no solo la evolución tecnológica, sino también la transformación de los criterios de eficiencia en sistemas computacionales.

Tabla 1

Dimensiones de análisis de los paradigmas computacionales y la eficiencia arquitectónica

Dimensión	Alcance analítico	Contribución a la eficiencia arquitectónica
Evolución epistemológica de la computación	Analiza el tránsito desde la computación mecánica, lógica y teórica hacia la inteligencia algorítmica y la e-Science.	Permite comprender la computación como sistema de conocimiento y no solo como infraestructura técnica.
Redes de interconexión	Examina topologías, latencia, bisección, comunicación entre nodos y restricciones físicas de diseño.	Mejora la coordinación entre procesadores y reduce cuellos de botella comunicacionales.
Jerarquía de memoria	Considera caché, DRAM, memoria emergente, coherencia, consistencia y optimización de accesos.	Reduce latencias, mejora el throughput y disminuye penalizaciones asociadas al memory wall.
Cloud, Big Data y almacenamiento distribuido	Integra ubicación de datos, migración, almacenamiento, procesamiento distribuido y propiedades no funcionales.	Optimiza costos, disponibilidad, escalabilidad, rendimiento y acceso a grandes volúmenes de datos.
Computación de borde y post-nube	Estudia fog, edge, mobile edge y dew computing como extensiones distribuidas de la nube.	Reduce latencia, mejora sensibilidad geográfica y acerca procesamiento a usuarios y dispositivos.

Heterogeneidad y aceleración	Analiza GPU, APU, FPGA, coprocesadores y métodos de acceso a datos.	Incrementa rendimiento y eficiencia energética cuando existe adecuada adaptación algorítmica.
Confiabilidad y tolerancia a fallos	Considera ECC, fallos de memoria, DRAM, DDR5 y mecanismos de corrección.	Protege la integridad de los datos y sostiene desempeño en sistemas escalados.
Optimización y modelos algorítmicos	Incorpora teoría de grafos, algoritmos de optimización, paralelismo y procesamiento adaptativo.	Permite diseñar arquitecturas más eficientes según restricciones físicas, lógicas y de aplicación.

Nota. La tabla sintetiza las dimensiones principales identificadas en el acervo documental para estudiar la relación entre paradigmas computacionales y eficiencia arquitectónica en sistemas de alto rendimiento.

Pertinencia de una revisión sistemática sobre paradigmas computacionales y eficiencia arquitectónica

La diversidad del acervo evidencia que el estudio de los paradigmas computacionales exige una aproximación sistemática e integradora. Los documentos revisados abordan objetos aparentemente distintos: epistemología computacional, ciencia de la computación, redes de interconexión, arquitecturas HPC, computación en la nube, computación de borde, big data, memoria, caché, confiabilidad, GPU, APU, procesamiento de consultas y logística computacional. Sin embargo, todos convergen en una preocupación común: cómo diseñar, organizar y optimizar sistemas capaces de procesar información de manera eficiente bajo restricciones crecientes de escala, latencia, energía, comunicación, confiabilidad y volumen de datos.

La revisión sistemática se justifica porque permite ordenar esta diversidad en categorías analíticas coherentes. En lugar de estudiar cada paradigma como una innovación aislada, el enfoque integrativo permite observar continuidades, rupturas y desplazamientos. La nube no reemplaza totalmente al cómputo local, sino que se complementa con edge y fog computing; las GPU no sustituyen a las CPU, sino que se integran en sistemas heterogéneos; la memoria principal no elimina todos los problemas de almacenamiento, sino que desplaza la atención hacia caché, consistencia y tolerancia a fallos; las redes de interconexión no solo conectan nodos, sino que condicionan el rendimiento global del sistema. Esta lectura permite superar enfoques fragmentados y construir una interpretación relacional de la eficiencia arquitectónica.

El artículo se propone, por tanto, analizar sistemáticamente cómo los paradigmas computacionales han respondido a los desafíos de rendimiento, escalabilidad,

distribución, memoria, comunicación, almacenamiento y confiabilidad. La pregunta que orienta la revisión puede formularse de la siguiente manera: ¿cómo han evolucionado los paradigmas computacionales y qué dimensiones arquitectónicas explican su contribución a la eficiencia de los sistemas de alto rendimiento? Esta pregunta permite integrar aportes teóricos, históricos y técnicos, conectando los fundamentos epistemológicos de la computación con las arquitecturas contemporáneas orientadas a datos, paralelismo y procesamiento distribuido.

La contribución esperada del artículo es doble. En primer lugar, ofrece una síntesis sistemática de los principales paradigmas computacionales y de las dimensiones arquitectónicas que condicionan su eficiencia. En segundo lugar, propone una interpretación integradora que permite comprender la eficiencia no como una propiedad aislada del hardware, sino como el resultado de interacciones entre cómputo, memoria, comunicación, datos, energía, confiabilidad, software y modelos de aplicación. De esta manera, el estudio busca aportar una base conceptual útil para futuras investigaciones sobre computación de alto rendimiento, arquitecturas distribuidas, ciencia intensiva en datos y sistemas computacionales emergentes.

Metodología

El presente artículo se desarrolló bajo un enfoque cualitativo, documental y de revisión sistemática con alcance integrativo, orientado a analizar la relación entre paradigmas computacionales y eficiencia arquitectónica en sistemas de alto rendimiento. La elección de este diseño responde a la necesidad de organizar, comparar e interpretar evidencia científica proveniente de estudios teóricos, revisiones sistemáticas, artículos técnicos, capítulos académicos, tesis doctorales y documentos especializados relacionados con arquitectura de computadores, computación de alto rendimiento, redes de interconexión, jerarquía de memoria, computación en la nube, edge computing, big data, aceleración heterogénea y confiabilidad de sistemas.

Fuentes documentales y estrategia de búsqueda

La estrategia de búsqueda se organizó a partir de términos relacionados con las variables centrales del artículo y con las áreas técnicas representadas en el acervo. Se utilizaron expresiones en español e inglés, tales como “paradigmas computacionales”, “arquitectura de computadores”, “eficiencia arquitectónica”, “computación de alto rendimiento”, “redes de interconexión”, “jerarquía de memoria”, “computación en la

nube”, “edge computing”, “fog computing”, “big data”, “computación heterogénea”, “GPU”, “APU”, “DRAM”, “cache coherence”, “computing paradigms”, “high-performance computing”, “computer architecture”, “interconnection networks”, “memory hierarchy”, “cloud computing”, “post-cloud computing”, “in-memory processing”, “heterogeneous computing” y “architectural efficiency”.

El acervo documental final estuvo conformado por documentos científicos y técnicos vinculados con el tema de investigación. Se incluyeron trabajos sobre evolución epistemológica de la computación, filosofía de la ciencia computacional, simulación computacional, computación molecular, computación paralela, clasificación de Flynn, supercomputadores, redes de interconexión, topologías de baja latencia, cloud computing, edge computing, big data, gestión de datos en memoria, sistemas distribuidos, optimización de memoria, confiabilidad de DRAM, corrección de errores, GPU, APU, procesamiento de consultas en hardware heterogéneo y arquitecturas ópticas de interconexión.

Criterios de inclusión

Para la conformación del corpus se aplicaron criterios de inclusión orientados a garantizar pertinencia temática, calidad académica y utilidad analítica. Se incluyeron documentos que abordaran al menos una de las siguientes dimensiones: paradigmas computacionales, arquitectura de computadores, computación paralela, computación de alto rendimiento, redes de interconexión, jerarquía de memoria, computación en la nube, edge computing, fog computing, big data, sistemas distribuidos, GPU, APU, hardware heterogéneo, procesamiento de consultas, cache coherence, confiabilidad de memoria, DRAM, ECC, simulación computacional o epistemología computacional.

También se consideraron artículos científicos, revisiones sistemáticas, capítulos de libro, tesis doctorales, ponencias académicas y documentos técnicos publicados en revistas, editoriales o conferencias especializadas. Se incluyeron textos clásicos o fundacionales cuando su aporte resultó necesario para comprender la evolución de los paradigmas computacionales, así como investigaciones recientes orientadas a tecnologías emergentes. La selección priorizó documentos con resumen, objetivos, desarrollo técnico o conceptual, metodología identificable y aporte explícito al análisis de la eficiencia arquitectónica.

Criterios de exclusión

Se excluyeron documentos que no presentaban relación directa con computación, arquitectura de sistemas, eficiencia computacional, procesamiento de datos o paradigmas tecnológicos. También fueron descartados textos cuya orientación fuera exclusivamente comercial, divulgativa o promocional, sin aporte técnico, metodológico o conceptual verificable. Asimismo, se excluyeron documentos duplicados, incompletos o con información insuficiente para identificar su contribución al objetivo del artículo.

No se consideraron documentos centrados en aplicaciones informáticas sin relación con arquitectura, rendimiento, memoria, interconexión, procesamiento distribuido o eficiencia computacional. También fueron excluidos estudios que abordaban tecnologías digitales desde una perspectiva puramente administrativa, educativa o social, cuando no aportaban elementos para comprender los paradigmas computacionales o la eficiencia arquitectónica. Esta depuración permitió mantener coherencia temática y evitar una dispersión excesiva del análisis.

Proceso de selección documental bajo el modelo PRISMA

El proceso de selección documental se organizó siguiendo la lógica del modelo PRISMA, con fases de identificación, cribado, elegibilidad e inclusión. En la fase de identificación se reunió el conjunto inicial de documentos científicos y técnicos proporcionados para el análisis. Posteriormente, se revisaron títulos, resúmenes, palabras clave, objetivos y pertinencia temática, con el propósito de determinar su relación con los ejes del artículo.

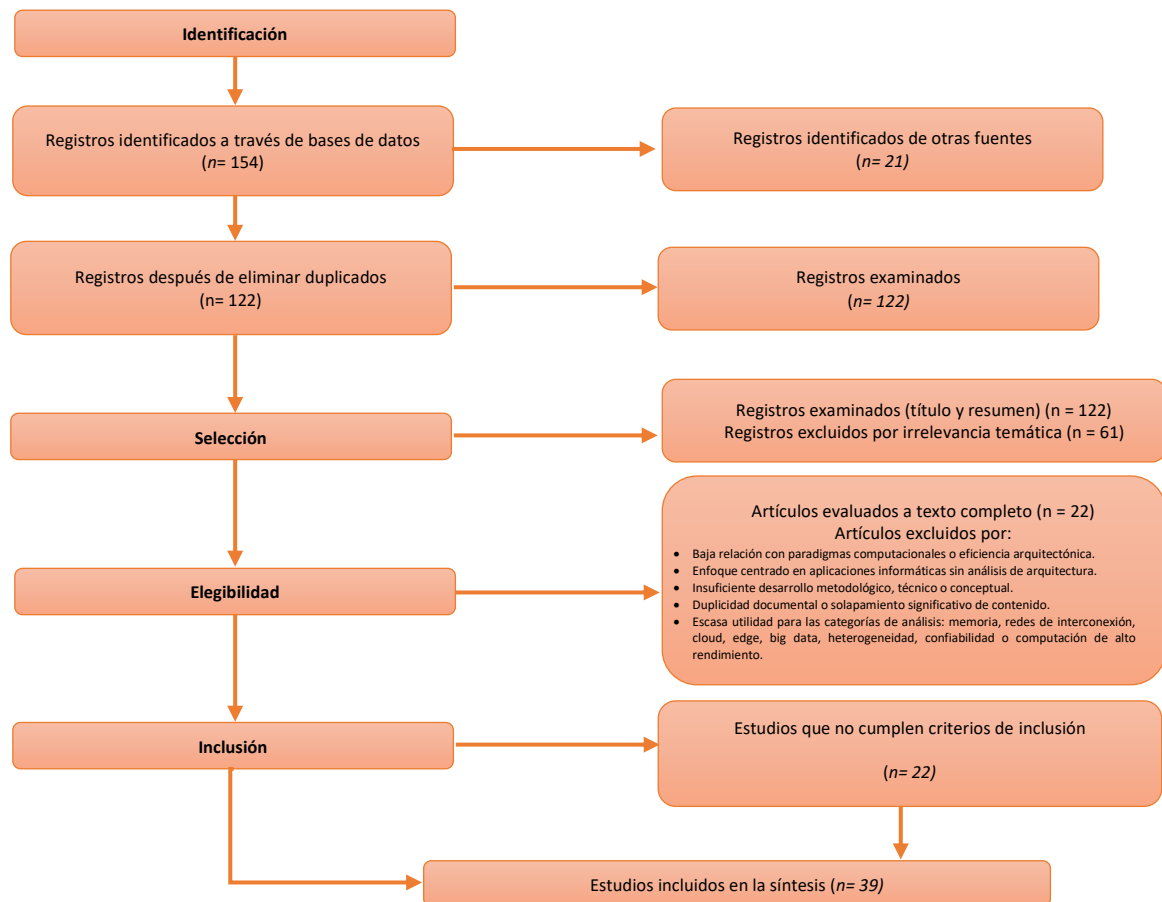
Durante la fase de cribado se descartaron documentos duplicados, textos con baja pertinencia temática o materiales cuyo aporte no se vinculaba directamente con los paradigmas computacionales y la eficiencia arquitectónica. En la fase de elegibilidad se revisaron los documentos completos o sus secciones principales para determinar su contribución a las categorías analíticas. Se valoró especialmente la presencia de conceptos, modelos, resultados, clasificaciones, arquitecturas, métricas de rendimiento, análisis comparativos o marcos teóricos aplicables al objeto de estudio.

El corpus final quedó conformado por los documentos que cumplían los criterios de inclusión y que aportaban evidencia relevante para explicar la evolución de los paradigmas computacionales y las dimensiones arquitectónicas de la eficiencia en sistemas de alto rendimiento. Esta selección permitió organizar el análisis en

categorías temáticas vinculadas con evolución epistemológica, computación paralela, redes de interconexión, memoria, cloud y edge computing, big data, sistemas heterogéneos, confiabilidad y optimización arquitectónica.

Figura 2

Diagrama PRISMA del proceso de selección de estudios



Nota. El diagrama representa el proceso de identificación, cribado, elegibilidad e inclusión de documentos que conformaron el corpus final de análisis.

Consideraciones de rigor metodológico

Para fortalecer la rigurosidad de la revisión se aplicaron criterios de pertinencia, coherencia, trazabilidad y equilibrio documental. La pertinencia se aseguró mediante la selección de fuentes directamente vinculadas con paradigmas computacionales, arquitectura, rendimiento y eficiencia. La coherencia se garantizó mediante la organización del corpus en categorías analíticas alineadas con el objetivo del estudio.

La trazabilidad se sostuvo mediante el uso de citas autor/año y referencias específicas a los documentos revisados.

El equilibrio documental se procuró integrando fuentes clásicas, estudios técnicos, revisiones recientes y documentos de fundamentación conceptual. Esta estrategia permitió evitar una lectura exclusivamente histórica o exclusivamente técnica, construyendo una interpretación que conecta evolución del conocimiento computacional con desafíos arquitectónicos contemporáneos. Asimismo, se mantuvo una distinción entre aportes conceptuales, estudios de arquitectura, análisis de memoria, investigaciones sobre redes, propuestas de sistemas distribuidos y estudios sobre hardware heterogéneo, con el propósito de no generalizar hallazgos fuera de su campo de aplicación.

La metodología adoptada permitió construir una base analítica suficiente para explicar cómo los paradigmas computacionales han evolucionado en respuesta a restricciones técnicas y científicas cada vez más complejas. En este sentido, el artículo organiza la evidencia disponible para mostrar que la eficiencia arquitectónica es el resultado de una interacción dinámica entre diseño computacional, procesamiento paralelo, memoria, comunicación, almacenamiento, confiabilidad, software y modelos de aplicación.

Resultados

El análisis del corpus permitió identificar seis tendencias principales. La primera corresponde a la evolución epistemológica de la computación, entendida como el tránsito desde la automatización de operaciones formales hacia infraestructuras de producción científica, simulación, inteligencia algorítmica y gestión del conocimiento. Durán (2013) destaca el valor epistémico de las simulaciones computacionales, al considerarlas herramientas capaces de explicar fenómenos científicos mediante modelos, algoritmos, verificación y validación.

La segunda tendencia se relaciona con el paso de arquitecturas secuenciales hacia modelos paralelos y distribuidos. Stefan (2006) plantea la computación paralela integral como una estructura orientada a optimizar recursos mediante paralelismo de instrucciones, datos y especulación. Ngoko y Trystram (2011) advierten que el fin del crecimiento exponencial asociado con la Ley de Moore exige nuevas formas de clasificar y aprovechar el paralelismo. Por tanto, la eficiencia ya no depende solo de

umentar la capacidad del procesador, sino de coordinar múltiples formas de procesamiento.

La tercera tendencia corresponde a las redes de interconexión como dimensión decisiva del rendimiento. Alam y Varshney (2015) señalan que la topología incide en diámetro, latencia, throughput y diversidad de rutas, mientras que Nakao et al. (2021) muestran que la optimización de grafos permite diseñar redes con menor diámetro y menor longitud promedio de caminos. Esto evidencia que la eficiencia arquitectónica requiere redes escalables y de baja latencia.

La cuarta tendencia se vincula con memoria, caché y confiabilidad. Aguilar y Leiss (2004) proponen un protocolo adaptativo de coherencia y reemplazo para cachés proxy web, mientras que Alkhamisi (2022) resalta la importancia de la coherencia de caché en multiprocesadores. Patel et al. (2019) y Criss et al. (2020) incorporan la confiabilidad en DRAM y DDR5, mostrando que la corrección de errores y el control de fallos son esenciales en sistemas escalados.

La quinta tendencia se relaciona con la gestión distribuida de datos y el ecosistema Cloud-Big Data. Zhang et al. (2015) muestran que la gestión in-memory responde a la necesidad de analítica de baja latencia, mientras que Mazumdar et al. (2019) sostienen que el almacenamiento y la colocación de datos deben considerar costo, rendimiento y latencia. Qin (2025) propone una arquitectura distribuida con capas de almacenamiento, procesamiento e interfaz, además de tolerancia a fallos mediante múltiples réplicas.

La sexta tendencia corresponde a la heterogeneidad y aceleración. Los estudios sobre GPU, APU, coprocesadores y procesamiento heterogéneo coinciden en que la aceleración mejora el rendimiento solo cuando se gestionan adecuadamente la transferencia de datos, la distribución de tareas y la adaptación del algoritmo al hardware. Cao, Wu y Wang (2011) muestran que la compresión inteligente puede optimizar la visualización de datos volumétricos, mientras que Kozar et al. (2025) destacan la necesidad de mecanismos adaptativos para ejecutar consultas en el procesador más adecuado.

Resultados por categorías de análisis

En la categoría de evolución epistemológica y conceptual, los documentos evidencian que los paradigmas computacionales han transformado la producción y validación del conocimiento. Think y Tung (2026) identifican etapas desde la computación mecánica hasta la inteligencia algorítmica; Vallverdú i Segura (2009) plantea la necesidad de una epistemología computacional para la e-Science; y Durán (2013) reconoce la capacidad explicativa de las simulaciones computacionales. Así, la computación contemporánea no solo ejecuta cálculos, sino que organiza nuevas formas de observación, modelación y explicación científica.

En arquitecturas paralelas y de alto rendimiento, los hallazgos muestran que la eficiencia depende de coordinar procesamiento concurrente, redes, memoria y modelos de ejecución. Van der Steen (2008) clasifica las arquitecturas HPC en SIMD, MIMD, memoria compartida, memoria distribuida y clústeres. Stefan (2006) propone integrar paralelismo temporal, espacial y especulativo, mientras que Dally (1990) y Agarwal (1991) advierten que la comunicación entre nodos puede limitar el rendimiento. Esto confirma que los sistemas de alto rendimiento deben diseñarse como arquitecturas balanceadas.

En redes de interconexión, los resultados evidencian que topología, latencia, bisección y localidad de comunicación son factores decisivos. Proietti et al. (2015) proponen una arquitectura óptica escalable y de baja latencia, mientras que Nakao et al. (2021) muestran que la optimización de grafos puede superar topologías convencionales. Las redes de interconexión, por tanto, constituyen una frontera clave de innovación para sistemas masivos.

En memoria, caché y confiabilidad, los resultados muestran que el rendimiento está condicionado por el acceso a datos. Vaithianathan (2025) identifica el memory wall como desafío central; Zhang et al. (2015) evidencian que los sistemas in-memory reducen el impacto del disco, pero aumentan exigencias de caché, concurrencia y tolerancia a fallos; y Patel et al. (2019) muestran que el ECC en DRAM mejora confiabilidad, aunque complejiza la caracterización de errores.

En cloud, edge y big data, los hallazgos muestran una transición desde infraestructuras centralizadas hacia modelos distribuidos y sensibles a la ubicación. Shawish y Salama (2014) explican la nube como paradigma de servicios virtualizados; Zhou et al. (2017)

muestran que los modelos post-nube responden a limitaciones de latencia; y Li et al. (2018) describen el edge computing como extensión hacia la periferia de la red. A la vez, Mazumdar et al. (2019) y Qin (2025) evidencian que el big data exige almacenamiento distribuido, colocación óptima y tolerancia a fallos.

En heterogeneidad y aceleración, los resultados indican que GPU, APU, FPGA y coprocesadores mejoran el desempeño cuando el modelo de ejecución se ajusta al hardware. Wilkinson y Skylaris (2013) advierten que las transferencias CPU-GPU pueden limitar el beneficio de la aceleración; Kalidas et al. (2015) muestran que los métodos de acceso a datos inciden en rendimiento y energía; y Kozar et al. (2025) sostienen que el procesamiento heterogéneo exige compilación adaptativa y optimización dinámica.

Vacíos identificados en la literatura revisada

La revisión permitió identificar cinco vacíos. El primero corresponde a la limitada integración entre perspectivas epistemológicas y técnicas, ya que los estudios sobre evolución computacional, simulación y epistemología no siempre se conectan con análisis arquitectónicos sobre memoria, redes, nube, edge o hardware heterogéneo.

El segundo vacío se relaciona con la fragmentación del análisis de eficiencia. Muchos estudios se enfocan en componentes específicos, como DRAM, caché, GPU, redes o almacenamiento distribuido, pero pocos explican cómo interactúan en sistemas de alto rendimiento.

El tercer vacío corresponde a la escasez de investigaciones comparativas entre paradigmas. Aunque la literatura describe cloud, edge, fog, in-memory, GPU y redes ópticas, son menos frecuentes los estudios que los comparan bajo criterios comunes como latencia, escalabilidad, costo, energía, confiabilidad, seguridad y adaptabilidad.

El cuarto vacío se vincula con la sostenibilidad energética y ambiental. Aunque algunos documentos mencionan rendimiento, energía y potencia, el análisis energético aún aparece subordinado al desempeño, pese al crecimiento de centros de datos, inteligencia artificial, big data y supercomputación.

El quinto vacío corresponde a la débil articulación entre sistemas emergentes y formación académica. La evolución hacia arquitecturas distribuidas, heterogéneas,

post-nube y orientadas a datos plantea desafíos para la actualización curricular y la formación de profesionales capaces de comprender sistemas integrados.

En términos generales, los resultados muestran que los paradigmas computacionales han evolucionado como respuesta a restricciones de procesamiento, memoria, comunicación, almacenamiento, latencia, confiabilidad y datos. La eficiencia arquitectónica emerge como una propiedad multidimensional, condicionada por la interacción entre hardware, software, red, memoria, datos, energía y aplicación. Por ello, el desafío principal consiste en superar análisis fragmentados y avanzar hacia modelos integradores que permitan diseñar sistemas de alto rendimiento más eficientes, confiables y adaptativos.

Discusión

La revisión sistemática permite sostener que los paradigmas computacionales han evolucionado como respuesta a restricciones sucesivas de procesamiento, memoria, comunicación, almacenamiento, confiabilidad y volumen de datos. Esta evolución no debe entenderse solo como una transición tecnológica, sino como una reorganización de las formas en que la computación representa, procesa, distribuye y valida información. Thinh y Tung (2026) explican que la ciencia de la computación ha transitado desde la computación mecánica hasta la inteligencia algorítmica, mientras que Vallverdú i Segura (2009) plantea que la e-Science exige una epistemología computacional capaz de explicar el papel de supercomputadores, inteligencia artificial, bases de datos y computación distribuida en la producción científica. Por tanto, los paradigmas computacionales no solo resuelven problemas técnicos, sino que transforman las prácticas de conocimiento y los criterios de eficiencia.

Uno de los principales puntos de discusión se relaciona con el hecho de que la eficiencia arquitectónica ya no depende exclusivamente del aumento de velocidad del procesador. La literatura revisada muestra que el rendimiento de los sistemas de alto desempeño está condicionado por la interacción entre cómputo, memoria, redes, transferencia de datos, paralelismo, almacenamiento y confiabilidad. Van der Steen (2008) organiza las arquitecturas de alto rendimiento en clases como SIMD, MIMD, memoria compartida, memoria distribuida, ccNUMA y clústeres, lo que evidencia que no existe una única arquitectura óptima, sino configuraciones ajustadas a problemas y cargas de trabajo específicas.

La computación paralela confirma esta transformación. Stefan (2006) propone una arquitectura paralela integral orientada a optimizar recursos mediante paralelismo de instrucciones, datos y especulación. De forma complementaria, Ngoko y Trystram (2011) advierten que el fin del crecimiento exponencial asociado con la Ley de Moore obliga a reconsiderar las formas tradicionales de clasificar y aprovechar el paralelismo. Así, la eficiencia contemporánea no se logra únicamente agregando más núcleos, sino coordinando adecuadamente paralelismo, comunicación y asignación de trabajo.

Las redes de interconexión aparecen como una dimensión crítica de la eficiencia sistémica. Dally (1990) demuestra que los sistemas VLSI están limitados por el cableado y que el desempeño depende de la densidad de interconexión, la latencia y el ancho de bisección. Agarwal (1991) amplía este análisis al mostrar que la latencia de red depende de retardos de conmutación, retardos de cableado, tamaño de paquete, localidad de comunicación y restricciones físicas de diseño. En esta línea, Alam y Varshney (2015) sostienen que la topología de interconexión influye en el diámetro, la distancia promedio entre nodos, el tráfico y el rendimiento, mientras que Nakao et al. (2021) evidencian que la optimización de grafos puede generar topologías con menor diámetro y menor longitud promedio de caminos. Por ello, el diseño de redes debe incorporar optimización, teoría de grafos y criterios de baja latencia.

Otro eje central corresponde a la memoria. Vaithianathan (2025) identifica el memory wall como un problema estructural de las arquitecturas HPC, debido a que la velocidad de procesamiento ha crecido más rápido que la capacidad de acceso eficiente a memoria. Zhang et al. (2015) muestran que los sistemas in-memory reducen el cuello de botella del disco y habilitan analítica interactiva, aunque también hacen más visibles desafíos como uso de caché, concurrencia, tolerancia a fallos y consistencia. En consecuencia, la memoria no puede entenderse como un recurso pasivo, sino como un componente activo que condiciona el rendimiento global.

La coherencia de caché y la confiabilidad refuerzan esta idea. Aguilar y Leiss (2004) proponen un protocolo adaptativo de coherencia y reemplazo para cachés proxy web, orientado a reducir latencia, tráfico y carga de servidores. Alkhamisi (2022) señala que, en sistemas multiprocesador, la coherencia de caché es indispensable para mantener consistencia entre copias de datos compartidos. A su vez, Patel et al. (2019) muestran que la corrección de errores en DRAM mejora la confiabilidad, aunque puede ocultar

patrones reales de fallos, mientras que Criss et al. (2020) plantean que DDR5 busca acotar fallos para reducir la sobrecarga de ECC. Estos aportes evidencian que confiabilidad, consistencia e integridad son dimensiones inseparables del rendimiento.

La nube representó un cambio paradigmático, pero no resolvió todos los problemas de eficiencia. Shawish y Salama (2014) describen cloud computing como un modelo de servicios basado en virtualización, elasticidad y pago por uso. Sin embargo, Zhou et al. (2017) advierten que la nube centralizada enfrenta limitaciones ante aplicaciones ubicuas y sensibles a latencia. Li et al. (2018) profundizan esta perspectiva al señalar que edge y fog computing extienden servicios desde centros de datos hacia la periferia de la red. Por tanto, el debate no debe plantearse como sustitución de la nube, sino como redistribución del procesamiento en arquitecturas híbridas.

El crecimiento de big data introduce otra dimensión de complejidad. Mazumdar et al. (2019) muestran que la ubicación, almacenamiento y migración de datos en ecosistemas Cloud-Big Data deben considerar costo, rendimiento, latencia, privacidad y disponibilidad. Qin (2025) propone una arquitectura distribuida de gestión de big data con capas de almacenamiento, procesamiento e interfaz, además de mecanismos de tolerancia a fallos mediante múltiples réplicas. Así, la eficiencia arquitectónica no depende solo de la capacidad de cómputo, sino también de dónde se almacenan los datos, cómo se distribuyen y qué garantías de confiabilidad ofrece el sistema.

La heterogeneidad arquitectónica constituye otro punto crítico. Wilkinson y Skylaris (2013) evidencian que las GPU pueden acelerar operaciones intensivas como las FFT, aunque la transferencia de datos entre CPU y GPU puede limitar las ganancias. Kalidas et al. (2015) muestran que los métodos de acceso a datos en GPU y APU influyen en desempeño, energía y potencia. Kozar et al. (2025) amplían esta discusión al señalar que el hardware heterogéneo exige compilación adaptativa, distribución de cargas y optimización en tiempo de ejecución. En consecuencia, la aceleración no debe entenderse como simple incorporación de coprocesadores, sino como rediseño de la relación entre algoritmo, datos, memoria y arquitectura.

Los estudios revisados también permiten debatir la relación entre especialización y flexibilidad. Las arquitecturas heterogéneas, las GPU, las redes ópticas y los sistemas in-memory pueden mejorar el rendimiento en tareas específicas, pero introducen

problemas de portabilidad, transferencia de datos, complejidad de programación y dependencia de la carga de trabajo. Proietti et al. (2015) muestran que las interconexiones ópticas pueden ofrecer baja latencia y alta escalabilidad, aunque su adopción exige compatibilidad con infraestructuras, costos y modelos de comunicación. Esta tensión constituye uno de los dilemas centrales de la eficiencia arquitectónica contemporánea.

Desde una perspectiva integradora, los paradigmas computacionales no avanzan por reemplazo lineal, sino por acumulación y redistribución de funciones. La nube no elimina la computación local; edge computing no elimina la nube; GPU y APU no eliminan CPU; los sistemas in-memory no eliminan los problemas de consistencia; y las redes optimizadas no eliminan las restricciones físicas. Cada paradigma responde a un conjunto de restricciones y, al hacerlo, genera nuevos desafíos técnicos.

En este sentido, la eficiencia arquitectónica en sistemas de alto rendimiento requiere integración entre procesamiento, memoria y comunicación; adaptación del paradigma a la carga de trabajo; confiabilidad para preservar la integridad de los datos; y flexibilidad para combinar nube, borde, aceleradores, redes y almacenamiento según necesidades específicas. La evidencia no permite afirmar que exista un paradigma único capaz de resolver todos los problemas de eficiencia. Por el contrario, muestra que la arquitectura computacional contemporánea se orienta hacia modelos híbridos, distribuidos, heterogéneos y adaptativos.

En suma, los paradigmas computacionales y la eficiencia arquitectónica deben analizarse como dimensiones interdependientes. La evolución de la computación ha ampliado las capacidades humanas de cálculo, simulación, análisis de datos y automatización cognitiva, pero también ha generado nuevas restricciones vinculadas con memoria, red, energía, confiabilidad y complejidad. Por ello, los sistemas de alto rendimiento no pueden comprenderse solo por sus componentes técnicos, sino por la forma en que articulan paradigmas, arquitecturas, datos y conocimiento para responder a problemas cada vez más complejos.

Conclusiones

En conclusión, la revisión sistemática desarrollada permite sostener que los paradigmas computacionales han evolucionado como respuesta a restricciones

técnicas, científicas y organizacionales relacionadas con procesamiento, memoria, comunicación, almacenamiento, confiabilidad, escalabilidad y volumen de datos. Esta evolución no se limita a una secuencia de innovaciones tecnológicas, sino que expresa una transformación profunda en la manera en que la computación organiza el conocimiento, automatiza procesos, modela fenómenos, ejecuta tareas complejas y sostiene infraestructuras científicas y productivas. Desde la computación mecánica y formal hasta la inteligencia algorítmica, la e-Science, la computación en la nube, el edge computing y las arquitecturas heterogéneas, cada paradigma ha surgido para responder a problemas concretos de eficiencia, capacidad, acceso, distribución o especialización.

En síntesis, uno de los hallazgos centrales de la revisión es que la memoria constituye una frontera crítica para la eficiencia computacional contemporánea. La literatura revisada evidencia que el fenómeno del memory wall, la coherencia de caché, la confiabilidad de DRAM, la corrección de errores y el procesamiento in-memory condicionan de manera directa el rendimiento de los sistemas de alto desempeño. Vaithianathan (2025), Zhang et al. (2015), Aguilar y Leiss (2004), Alkhamisi (2022), Patel et al. (2019) y Criss et al. (2020) permiten concluir que no basta con incrementar capacidad de cómputo si los sistemas no resuelven adecuadamente el acceso, consistencia, ubicación, integridad y disponibilidad de los datos. La memoria deja de ser un soporte pasivo para convertirse en un componente estratégico de la arquitectura.

Asimismo, el análisis permitió concluir que la computación en la nube representó un avance fundamental hacia la virtualización, elasticidad y provisión de recursos bajo demanda; sin embargo, sus limitaciones de latencia, ancho de banda, ubicación y sensibilidad contextual impulsaron el surgimiento de paradigmas post-nube. Shawish y Salama (2014), Zhou et al. (2017), y Li et al. (2018) muestran que cloud computing, fog computing, edge computing, mobile edge computing y dew computing deben comprenderse como paradigmas complementarios, no necesariamente excluyentes. La eficiencia arquitectónica actual exige distribuir el procesamiento entre centros de datos, redes intermedias, dispositivos cercanos al usuario y fuentes de datos, de acuerdo con los requerimientos de cada aplicación.

Otro hallazgo relevante se relaciona con la heterogeneidad arquitectónica. La revisión evidencia que GPU, APU, FPGA, coprocesadores y hardware especializado permiten mejorar el rendimiento de ciertas cargas de trabajo, pero su eficacia depende de la correspondencia entre algoritmo, arquitectura y patrón de datos. Wilkinson y Skylaris (2013), Kalidas et al. (2015), Cao et al. (2011) y Kozar et al. (2025) muestran que la aceleración puede verse limitada por transferencias CPU-GPU, caminos de acceso a memoria, distribución de tareas o falta de adaptación del software. Por tanto, la heterogeneidad no debe concebirse como simple incorporación de aceleradores, sino como un rediseño del ecosistema computacional para asignar cada operación al recurso más adecuado.

En conclusión, la principal contribución de esta revisión sistemática consiste en demostrar que los paradigmas computacionales y la eficiencia arquitectónica mantienen una relación estrecha, dinámica y multidimensional. Cada paradigma surge para superar limitaciones de los anteriores, pero al mismo tiempo introduce nuevos desafíos. La nube mejora elasticidad, pero genera problemas de latencia; el borde reduce proximidad y retardo, pero complejiza la gestión distribuida; las GPU aceleran cálculos paralelos, pero pueden generar cuellos de botella de transferencia; los sistemas in-memory reducen dependencia del disco, pero aumentan exigencias de consistencia y tolerancia a fallos; las redes optimizadas reducen latencia, pero enfrentan restricciones físicas y de implementación. Esta dinámica evidencia que la evolución computacional no es lineal, sino acumulativa, híbrida y adaptativa.

Referencias

- Agarwal, A. (1991). Limits on interconnection network performance. Massachusetts Institute of Technology, Laboratory for Computer Science.
- Aguilar, J., & Leiss, E. L. (2004). An adaptive coherence-replacement protocol for web proxy cache systems. *Computación y Sistemas*, 8(1), 1–14.
- Aguilar, J., & Leiss, E. L. (2006). A coherence-replacement protocol for web proxy cache systems. *International Journal of Computers and Applications*, 28(1).
- Alam, M., & Varshney, A. K. (2015). A comparative study of interconnection network. *International Journal of Computer Applications*, 127(4), 37–43.
- Alkhamisi, K. H. (2022). Cache coherence issues and solution: A review. *International Journal of Information Systems and Computer Technologies*, 1(2), 1–6.
- Cao, Y., Wu, G., & Wang, H. (2011). A smart compression scheme for GPU-accelerated volume rendering of time-varying data. In 2011 International Conference on Virtual Reality and Visualization.
- Cho, Y. S., Choi, E. J., & Cho, K. R. (2005). Modeling and analysis of the system bus latency on the SoC platform. In *Proceedings of the 2005 Asia and South Pacific Design Automation Conference*, 738–741. <https://doi.org/10.1145/1117278.1117293>
- Criss, K., Bains, K., Agarwal, R., Bennett, T., Grunzke, T., Kim, J. K., Chung, H., & Jang, M. (2020). Improving memory reliability by bounding DRAM faults: DDR5 improved reliability features. In *The International Symposium on Memory Systems*.
- Dally, W. J. (1990). Performance analysis of k-ary n-cube interconnection networks. *IEEE Transactions on Computers*, 39(6), 775–785.
- Durán, J. M. (2013). Explaining simulated phenomena: A defense of the epistemic power of computer simulations [Doctoral dissertation, Universität Stuttgart].
- Gunarathne, T., Wu, T. L., Qiu, J., & Fox, G. (2010). Cloud computing paradigms for pleasingly parallel biomedical applications. In *Proceedings of the 19th ACM*

- International Symposium on High Performance Distributed Computing, 460–469. <https://doi.org/10.1145/1851476.1851544>
- Gupta, A. K., & Dally, W. J. (2005). Topology optimization of interconnection networks. Computer Systems Laboratory, Stanford University.
- Hanf, W. (1975). The Boolean algebra of logic. *Bulletin of the American Mathematical Society*, 81(3), 587–589.
- Kalidas, R., Daga, M., Krommydas, K., & Feng, W. C. (2015). On the performance, energy, and power of data-access methods in heterogeneous computing systems. In *Proceedings of the 2015 IEEE International Workshop on High-Performance, Power-Aware Computing*.
- Kozar, A., von Bleichert, J., Breß, S., Grulich, P. M., Lutz, C., Rabl, T., Rosenfeld, V., Traub, J., Zeuch, S., & Markl, V. (2025). Query processing on heterogeneous hardware. In K.-U. Sattler et al. (Eds.), *Scalable Data Management for Future Hardware* (pp. 39–64). Springer. https://doi.org/10.1007/978-3-031-74097-8_2
- Li, B., & Song, G. (2020). Computational logistics for container terminal logistics hubs based on computational lens and computing principles. *IEEE Access*, 8. <https://doi.org/10.1109/ACCESS.2020.3033849>
- Li, C., Xue, Y., Wang, J., Zhang, W., & Li, T. (2018). Edge-oriented computing paradigms: A survey on architecture design and system management. *ACM Computing Surveys*, 51(2), Article 39. <https://doi.org/10.1145/3154815>
- Mazumdar, S., Seybold, D., Kritikos, K., & Verginadis, Y. (2019). A survey on data storage and placement methodologies for Cloud-Big Data ecosystem. *Journal of Big Data*, 6, Article 15. <https://doi.org/10.1186/s40537-019-0178-3>
- McCune, W., Veroff, R., Fitelson, B., Harris, K., Feist, A., & Wos, L. (2002). Short single axioms for Boolean algebra. *Journal of Automated Reasoning*.
- Nakao, M., Sakai, M., Hanada, Y., Murai, H., & Sato, M. (2021). Graph optimization algorithm for low-latency interconnection networks. *Parallel Computing*, 106, Article 102805. <https://doi.org/10.1016/j.parco.2021.102805>

- Ngoko, Y., & Trystram, D. (2011). Revisiting Flynn's classification: The portfolio approach. In Euro-Par 2011: Parallel Processing Workshops.
- O'Flynn, E., Danial, A., & Gajewski, J. (2022). Global surgery education and training programmes: A scoping review and taxonomy. *Indian Journal of Surgery*, 84(Suppl. 1), S193–S206. <https://doi.org/10.1007/s12262-021-03081-w>
- Patel, M., Kim, J. S., Hassan, H., & Mutlu, O. (2019). Understanding and modeling on-die error correction in modern DRAM: An experimental study using real devices. In 2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks.
- Păun, G., Rozenberg, G., & Salomaa, A. (1998). DNA computing: New computing paradigms. Springer.
- Pham Van Thinh, & Nguyen Thanh Tung. (2026). From mechanical computation to algorithmic intelligence: An evolutionary epistemological analysis of computer science. *European Journal of Applied Science, Engineering and Technology*, 4(2), 312–320.
- Primiero, G. (2025). The philosophy of computer science. In E. N. Zalta & U. Nodelman (Eds.), *The Stanford Encyclopedia of Philosophy*. Stanford University.
- Proietti, R., Cao, Z., Nitta, C. J., Li, Y., & Yoo, S. J. B. (2015). A scalable, low-latency, high-throughput, optical interconnect architecture based on arrayed waveguide grating routers. *Journal of Lightwave Technology*, 33(4), 911–920.
- Qin, J. (2025). Design of big data management system based on distributed architecture. In 2025 5th International Conference on Big Data, Artificial Intelligence and Risk Management. ACM. <https://doi.org/10.1145/3800227.3800318>
- Sharma, N., & Prabha, C. (s. f.). Computing paradigms: An overview.
- Shawish, A., & Salama, M. (2014). Cloud computing: Paradigms and technologies. In F. Xhafa & N. Bessis (Eds.), *Inter-cooperative Collective Intelligence: Techniques and Applications* (pp. 39–67). Springer. https://doi.org/10.1007/978-3-642-35016-0_2

- Song, G., Li, B., & He, Y. (2019). Container terminal logistics generalized computing architecture and green initiative computational pattern performance evaluation. *Information*, 10(12), Article 383. <https://doi.org/10.3390/information10120383>
- Stanford University. (s. f.). Flynn's taxonomy [Lecture slides].
- Ștefan, G. (2006). Integral parallel computation. *Proceedings of the Romanian Academy, Series A*, 7(3), 1–8.
- Vaithianathan, M. (2025). Memory hierarchy optimization strategies for high-performance computing architectures. *International Journal of Emerging Trends in Computer Science and Information Technology*, 6(1), 23–34. <https://doi.org/10.63282/3050-9246/IJETCSIT-V6I1P103>
- Vallverdú i Segura, J. (2009). Computational epistemology and e-Science: A new way of thinking. *Minds and Machines*. <https://doi.org/10.1007/s11023-009-9168-0>
- Van der Steen, A. J. (2008). Overview of recent supercomputers. NCF/HPC Research.
- Wilkinson, K., & Skylaris, C. K. (2013). Porting ONETEP to graphical processing unit-based coprocessors. 1. FFT box operations. *Journal of Computational Chemistry*. <https://doi.org/10.1002/jcc.23410>
- Zhang, H., Chen, G., Ooi, B. C., Tan, K. L., & Zhang, M. (2015). In-memory big data management and processing: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 27(7), 1920–1948. <https://doi.org/10.1109/TKDE.2015.2427795>
- Zhou, Y., Zhang, D., & Xiong, N. (2017). Post-cloud computing paradigms: A survey and comparison. *Tsinghua Science and Technology*, 22(6), 714–732.

Conflicto de intereses

Los autores declaran no tener conflictos de intereses.