Accelerating Fully Homomorphic Encryption at Scale: Lessons from Storage-Centric System Design

Xuan Wang* and Tajana Rosing*
*University of California San Diego
Email: {xuw009, tajana}@ucsd.edu

***Abstract*—Fully Homomorphic Encryption (FHE) is widely regarded as a promising solution for privacy-preserving computation, enabling data to be processed without ever being decrypted. While advances in cryptography and architectural acceleration have made FHE increasingly practical, the ethical and societal impact of FHE systems is shaped not only by cryptographic guarantees but also by system and architecture design choices. In practice, ciphertext expansion, long-latency primitives, and massive data movement place FHE workloads under severe memory, storage, and energy pressure, often restricting deployability to large, resource-rich infrastructures.**

In this paper, we reflect on lessons learned from recent large-scale FHE acceleration efforts that adopt storage-centric system designs. Drawing from prior experience with memory-aggregation and in-storage FHE systems, we show that architectural decisions, such as where computation is placed, how data movement is managed, and which system assumptions are made, directly influence the accessibility, sustainability, and real-world viability of privacy-preserving computation. We argue that designs that ignore storage and I/O realities risk turning privacy into a luxury available only to a small subset of users, while also incurring substantial environmental costs. Rather than proposing new cryptographic mechanisms, this work highlights ethical trade-offs that emerge at the system level and discusses how storage-aware and locality-driven architectures can mitigate them. We conclude by outlining design principles and open challenges for building FHE systems that better align strong privacy guarantees with broader societal goals, including energy efficiency, scalability, and equitable access.

I. Introduction

Modern data-driven applications increasingly rely on large-scale analysis of sensitive information to support decision-making across domains such as healthcare, finance, scientific discovery, and machine learning. These workloads often involve comparing user-generated queries or measurements against large reference datasets to extract meaningful insights, detect patterns, or enable predictive modeling. As data volumes grow rapidly and analytical pipelines become more complex, cloud computing has emerged as the dominant platform for hosting and processing such large-scale workloads, offering elasticity, performance, and ease of collaboration.

Despite these advantages, the use of shared and remote computing infrastructures raises significant data sovereignty and information security concerns. Sensitive data, including personal health records, proprietary enterprise information, and regulated datasets, is often subject to strict legal and ethical constraints on how it can be stored, shared, and processed. Regulations such as the General Data Protection Regulation (GDPR), HIPAA [2], and similar frameworks worldwide restrict direct data exposure [12][19], while organizations are incentivized to protect their data to preserve privacy, trust, and competitive advantage [10]. These constraints create a fundamental tension between the desire to leverage large-scale computation and the need to maintain strong guarantees of confidentiality and control over sensitive information.

Fully Homomorphic Encryption (FHE) [7], [9], [4], [5] enables computation directly on encrypted data, offering strong privacy guarantees for sensitive workloads ranging from database search and analytics to machine learning inference [18], [3], [24], [22], [8]. Recent advances in cryptographic schemes and hardware acceleration have significantly reduced the computational overhead of FHE, bringing previously impractical workloads closer to deployment [14], [23], [20], [1], [11]. FHE is increasingly viewed as a promising foundation for privacy-preserving computing infrastructures.

Despite this progress, *ciphertext size explosion* still remains as a fundamental challenge. Compared to plaintext data, FHE ciphertexts can be hundreds to thousands of times larger [16], [23], and auxiliary cryptographic material such as evaluation and key-switching keys further amplifies memory and storage requirements. As a result, realistic FHE workloads often expand from gigabytes of plaintext data to hundreds of gigabytes or even hundreds of terabytes of encrypted state. This expansion places FHE systems under extreme pressure across the memory hierarchy, turning data movement, storage capacity, and I/O bandwidth into first-order bottlenecks rather than secondary concerns. For instance, previous works INSPIRE [17] and SmartPIR [6] have identified the storage I/O bottleneck as the dominant delay resource for the Private Information Retrieval (PIR) workloads (i.e, occupies over 90% overall execution time), which is an important methodology for private indexed data access. Additionally, even for workloads that are not conventionally I/O bound in cleartext domain, *ciphertext size explosion* could also become an issue in the FHE domain. For instance, BERT-Base requires 1.5TB of memory, and Llama3-8B requires 112TB to store the weight [13].

Importantly, *ciphertext size explosion* is not merely a performance issue. It directly shapes who can deploy FHE in practice and under what conditions. Prior FHE accelerators [20], [14], [16], [23], [15] that implicitly assume encrypted datasets fit within on-chip buffers or high-bandwidth memory risk limiting FHE deployment to a narrow class of resource-rich environments, such as hyperscale data centers equipped with

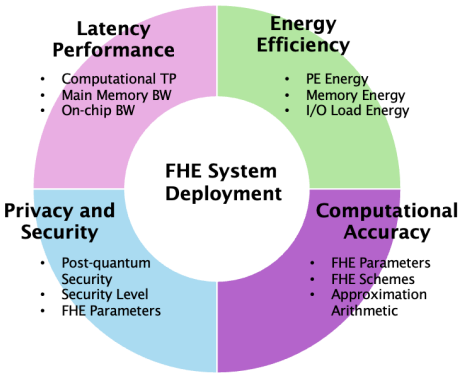

Fig. 1: System-Level Objectives for FHE Deployment.

specialized accelerators. In contrast, organizations operating under tighter cost, energy, or infrastructure constraints may find that the storage and I/O overhead of FHE renders strong privacy guarantees unattainable. In this sense, architectural assumptions about ciphertext size and data locality become ethical design choices that influence accessibility and equity.

The storage implications of ciphertext expansion also raise sustainability concerns. Moving and repeatedly materializing massive encrypted datasets across system boundaries incurs substantial energy cost, often dominating the computation itself. Designs that optimize homomorphic arithmetic while overlooking storage and I/O realities can inadvertently trade privacy for excessive energy consumption, undermining broader societal goals around environmental responsibility. As FHE workloads scale, the question is no longer only whether encrypted computation is correct or fast, but whether it is sustainable and deployable at all.

In this paper, we reflect on lessons learned from our recent large-scale FHE acceleration efforts that adopt storage-centric system designs to confront ciphertext expansion head-on. Drawing from experience with memory-disaggregated and in-storage FHE systems, we examine how placing computation closer to data, reducing unnecessary data movement, and explicitly accounting for storage constraints can reshape the ethical trade-offs of privacy-preserving computation. Rather than proposing new cryptographic mechanisms, we argue that system-and architecture-level decisions play a decisive role in determining whether FHE remains a theoretical privacy ideal or becomes a practical, sustainable, and broadly accessible technology. We conclude by distilling design principles and open challenges for future FHE systems, emphasizing the need to treat ciphertext size, storage capacity, and data movement as first-class concerns—not only for performance, but for aligning privacy-preserving computation with long-term societal and environmental goals.

II. Ciphertext Expansion as a System and Ethical Challenge

As introduced in Section I, this ciphertext expansion fundamentally reshapes the system-level trade-offs governing privacy-preserving computation. Figure 1 shows a summarized four-pillar view of system-level objectives for developing deployable FHE systems. Such must simultaneously balance privacy and security, latency performance, energy efficiency, and computational accuracy. Ciphertext expansion places immediate pressure on all four pillars by stressing the memory hierarchy and amplifying data movement as the FHE parameters increase. Designs that are effective for plaintext workloads—where data fits comfortably within caches, main memory, or accelerator-attached high-bandwidth memory—quickly break down once encrypted data is introduced. Even aggressively provisioned accelerators struggle to retain complete encrypted working sets locally, causing frequent spills across the memory hierarchy and repeated transfers over system interconnects.

From a performance perspective, ciphertext expansion shifts bottlenecks away from raw compute throughput toward memory and bandwidth constraints, as reflected in the latency-performance pillar of Figure 1. Encrypted workloads increasingly become limited by main-memory bandwidth, on-chip buffer capacity, and I/O throughput rather than homomorphic arithmetic itself. Optimizing computational throughput in isolation is therefore insufficient once ciphertext sizes exceed local memory capacity.

Ciphertext expansion also has direct implications for energy efficiency. As encrypted datasets spill across memory and storage boundaries, energy consumption associated with memory accesses and I/O transfers often dominates the overall energy budget, outweighing the energy cost of cryptographic computation. This places energy efficiency in direct tension with latency and security objectives, as depicted in the energy-efficiency pillar of Figure 1. Without careful attention to data placement and movement, scaling privacy-preserving computation risks incurring disproportionate environmental and operational costs.

The privacy-and-security pillar is likewise affected. Achieving higher post-quantum security levels or stronger cryptographic guarantees typically requires larger parameters, which further inflate ciphertext size and key material. Architectural assumptions that overlook these effects may achieve strong security in principle while rendering systems impractical or inaccessible in real deployments. In this sense, ciphertext expansion transforms system design decisions into ethical ones by shaping who can feasibly deploy privacy-preserving computation and under what constraints.

Finally, ciphertext expansion interacts closely with computational accuracy. Maintaining acceptable numerical precision in approximate FHE schemes often necessitates larger ciphertext parameters and deeper arithmetic circuits, reinforcing the coupling between accuracy and system cost.

As illustrated by the accuracy pillar in Figure 1, taken together, ciphertext expansion exposes a fundamental mismatch between traditional accelerator-centric design assumptions and the realities of large-scale encrypted computation. Addressing this mismatch requires treating storage capacity and data movement as first-class system concerns, not only for perfor-

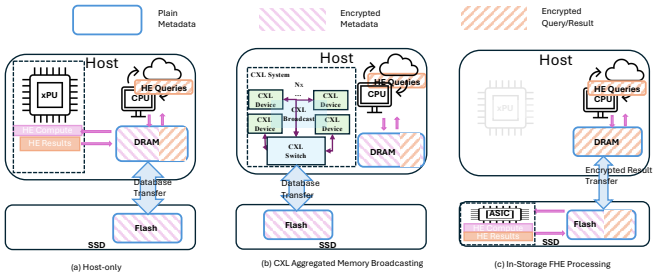

Fig. 2: Evolution of FHE System design. (a) Conventional FHE Accelerator Designs. (b) Memory-Agregated Designs Leverage CXL. (c) FHE In-Storage Processing Solutions.

mance, but for aligning privacy-preserving computation with broader ethical goals. In the following section, we examine how storage-centric system designs respond to these pressures by rethinking where computation is placed and how encrypted data is managed across the system stack.

III. Lessons from Storage-Centric FHE System Design

Confronted with the system-level challenges introduced by ciphertext expansion, our work progressed through a sequence of architectural explorations that increasingly elevated the role of storage in FHE execution. Rather than converging on a single solution, these explorations reflect an evolving understanding of how memory capacity, data movement, and computation placement jointly determine the practicality and ethical implications of privacy-preserving computation at scale. Figure 2 summarizes this progression.

A. First Exploration: Scaling Capacity via Memory Disaggregation

Figure 2 (a) reflects a common assumption in prior FHE accelerators [20], [14], [16], [23], [15], where data are assumed to fit into DRAM/HBM. As a result, if the encrypted database size goes beyond the main memory capacity, the encrypted data has to be fetched from storage into system memory, processed by compute units, and written back to storage after execution. While effective for small encrypted workloads, this model quickly breaks down under ciphertext expansion, as large ciphertexts and auxiliary keys must be repeatedly transferred across the memory hierarchy. In this regime, storage and I/O become dominant contributors to both latency and energy consumption.

A natural first solution is shown in Figure 2(b), where we proposed to scale effective memory capacity through memory aggregation in a scalable CXL-based multi-ASIC FHE system [22]. By leveraging a composable CXL switch as interconnects and shared memory pools, the HE dataset, ciphertext, and cryptographic keys can be distributed across multiple devices and reused by multiple accelerators. This reduces redundant data replication and alleviates the immediate capacity constraints imposed by accelerator-local memory.

From a system perspective, memory aggregation improves utilization and flexibility, allowing capacity to scale independently of compute throughput. From an ethical standpoint, it lowers the barrier to deploying FHE workloads by reducing reliance on monolithic accelerators with oversized local memory. However, this approach retains a critical assumption: that moving ciphertexts across the interconnect remains affordable.

However, as encrypted datasets grow into hundreds of terabytes, this assumption no longer holds. Even with shared memory, large ciphertexts must traverse the interconnect repeatedly, making bandwidth and energy consumption first-order bottlenecks. Memory aggregation mitigates capacity pressure but does not fundamentally address the inefficiency of transporting massive encrypted datasets across system boundaries. Additionally, although the CXL multi-ASIC system itself still scales, the overall performance benefit will be constrained due to the SSD congestion [22].

B. Second Exploration: Moving FHE Computation into Storage

Figure 2(c) demonstrates our exploration for the in-storage processing solutions regarding ciphertext expansion [21]. As ciphertext expansion pushes storage onto the critical execution path, storage can no longer be treated as a passive data source. Repeated transfers between flash and compute nodes incur prohibitive latency and energy overhead, motivating the placement of selected FHE computation closer to where encrypted data resides. By executing portions of FHE workloads near or within the storage device, storage-centric designs reduce unnecessary data movement and exploit the internal bandwidth and parallelism of modern storage systems. This fundamentally alters the cost structure of large-scale encrypted computation, where the performance becomes less dependent on external interconnects, and energy consumption is reduced by avoiding repeated data transfers.

This shift also reframes the ethical implications of FHE deployment. Reducing data movement improves sustainability and lowers operational cost, addressing concerns that privacy-preserving computation may impose excessive environmental overhead. Moreover, aligning FHE execution with commodity storage infrastructure improves deployability beyond highly provisioned environments, making strong privacy guarantees more broadly accessible.

At the same time, embedding computation within storage introduces new trade-offs. Storage devices operate under strict power, area, and reliability constraints, and integrating FHE computation requires careful co-design with storage controllers and firmware. Questions of isolation, multi-tenancy, and governance become more prominent when computation is embedded within shared storage infrastructure. These considerations highlight that no single architectural response resolves all ethical tensions; instead, each design choice redistributes costs and responsibilities across the system stack.

Taken together, the progression illustrated in Figure 2 demonstrates that addressing ciphertext expansion is not a

one-time optimization problem, but an evolving design challenge. Memory disaggregation and storage-centric execution represent successive steps toward reconciling strong privacy guarantees with the realities of large-scale systems, underscoring that ethical outcomes in privacy-preserving computation are inseparable from architectural decisions about where data resides and how it moves.

IV. Design Implications and Open Ethical Questions

The progression from accelerator-centric designs to memory aggregation and, ultimately, storage-centric execution highlights that the ethical impact of privacy-preserving computation is inseparable from system architecture. As ciphertext expansion pushes FHE workloads beyond conventional memory hierarchies, architects are forced to confront questions that extend well beyond performance optimization. Drawing on the lessons illustrated in Figure 2, we outline several design implications and open ethical questions that merit broader discussion within the systems community.

Treat storage and data movement as first-class ethical concerns. Early FHE accelerators implicitly treated storage as a passive data source, while subsequent memory-aggregated designs assumed that increased interconnect bandwidth could absorb growing ciphertext movement. As shown by the transition to storage-centric execution, ciphertext expansion makes data movement a dominant contributor to both system cost and energy consumption. Ignoring storage placement and I/O behavior risks promoting designs that are technically functional yet environmentally and economically unsustainable. Ethical system design for FHE therefore requires explicit consideration of storage capacity, bandwidth, and data movement alongside compute efficiency.

Rethink deployment accessibility and centralization. Architectures that assume abundant accelerator-local memory or high-bandwidth interconnects implicitly favor a small set of well-provisioned environments. While memory aggregation and in-storage execution reduce some of these barriers, they also shift complexity and control across the system stack. An open question is how to balance scalability with accessibility, ensuring that strong privacy guarantees do not become confined to centralized infrastructures operated by a few large providers.

Expand evaluation norms beyond performance metrics. The evolution from compute-centric to storage-centric FHE systems demonstrates that latency and throughput alone provide an incomplete view of system impact. Energy consumption, storage amplification, and infrastructure requirements increasingly determine whether FHE can be deployed in practice and at what societal cost. Incorporating such metrics into evaluation practices is essential for aligning research incentives with ethical goals related to sustainability and equitable access.

Acknowledge new governance and trust boundaries. Moving FHE computation closer to or into storage introduces new trust and governance considerations. When computation is embedded within shared storage infrastructure, questions arise regarding isolation, multi-tenancy, accountability, and control over execution environments. While FHE cryptographically protects data values, architectural placement decisions still shape who operates, audits, and ultimately benefits from privacy-preserving computation.

In summary, ciphertext expansion reveals that privacy is not delivered by cryptography alone. Architectural responses to storage and data movement constraints play a decisive role in determining whether FHE can be deployed sustainably, equitably, and at scale. By foregrounding these system-level tradeoffs, we hope to encourage a broader conversation about how privacy-preserving technologies should be designed, evaluated, and governed as they transition from research prototypes to societal infrastructure.

V. Conclusion

Fully Homomorphic Encryption offers a compelling foundation for privacy-preserving computation, yet its real-world impact is determined as much by system and architecture design as by cryptographic guarantees. Through the lens of ciphertext expansion, this paper illustrates how storage capacity, data movement, and computation placement fundamentally shape the performance, sustainability, and accessibility of FHE at scale. The progression from accelerator-centric designs to memory aggregation and, ultimately, storage-centric execution demonstrates that enabling privacy in large systems requires confronting architectural realities rather than relying on abstract assumptions.

More broadly, these experiences highlight that ethical outcomes in privacy-preserving computation are not an afterthought, but a direct consequence of system-level design choices. Treating storage and data movement as first-class considerations is essential for ensuring that strong privacy guarantees remain deployable, energy-efficient, and broadly accessible. We hope this work encourages the systems community to evaluate privacy-preserving technologies not only by their correctness or speed, but by how architectural decisions shape their societal and environmental impact as they move toward widespread deployment.

Acknowledgment

This work was supported in part by PRISM and CoCoSys—centers in JUMP 2.0, an SRC program sponsored by DARPA, and by the NSF under Grants No. 2112665, 2112167, 2052809, and 2211386.

REFERENCES

[1] A. Acar, H. Aksu, A. S. Uluagac, and M. Conti, "A survey on homomorphic encryption schemes: Theory and implementation," *ACM Computing Surveys (Csur)*, vol. 51, no. 4, pp. 1–35, 2018.

[2] A. Act *et al.*, "Health insurance portability and accountability act of 1996," *Public law*, vol. 104, p. 191, 1996.

[3] P. Antonopoulos *et al.*, "Azure sql database always encrypted," in *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 2020, pp. 1511–1525.

[4] Z. Brakerski, "Fully homomorphic encryption without modulus switching from classical gapsvp," in *Annual cryptology conference*. Springer, 2012, pp. 868–886.

[5] Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "(leveled) fully homomorphic encryption without bootstrapping," *ACM Transactions on Computation Theory (TOCT)*, vol. 6, no. 3, pp. 1–36, 2014.

[6] Z. Chen, H. You, Q. Wei, H. Lu, L. Ju, and Z. Shen, "Smartpir: A private information retrieval system using computational storage devices," in *Proceedings of the 58th IEEE/ACM International Symposium on Microarchitecture®*, 2025, pp. 1749–1762.

[7] J. H. Cheon *et al.*, "Homomorphic encryption for arithmetic of approximate numbers," in *International Conference on the Theory and Application of Cryptology and Information Security*, 2017, pp. 409–437.

[8] J. H. Cheon, M. Kang, T. Kim, J. Jung, and Y. Yeo, "Batch inference on deep convolutional neural networks with fully homomorphic encryption using channel-by-channel convolutions," *IEEE Transactions on Dependable and Secure Computing*, 2024.

[9] J. H. a. Cheon, "Bootstrapping for approximate homomorphic encryption," in *Advances in Cryptology–EUROCRYPT 2018*. Springer, 2018, pp. 360–384.

[10] A. Gangwal, A. Ansari, I. Ahmad, A. K. Azad, and W. M. A. W. Sulaiman, "Current strategies to address data scarcity in artificial intelligence-based drug discovery: A comprehensive review," *Computers in Biology and Medicine*, vol. 179, p. 108734, 2024.

[11] Y. Gong, X. Chang, J. Mišić, V. B. Mišić, J. Wang, and H. Zhu, "Practical solutions in fully homomorphic encryption: a survey analyzing existing acceleration methods," *Cybersecurity*, vol. 7, no. 1, p. 5, 2024.

[12] L. Hilty, W. Lohmann, S. Behrendt, M. Evers-Wölk, K. Fichter, R. Hintemann, M. Janßen, H. Moser, and M. Köhn, "Grüne software: Ermittlung und erschließung von umweltschutzpotenzialen der informations- und kommunikationstechnik (green it)," *UBA TEXTE*, vol. 22, 2015.

[13] S. Jayashankar, J. Kim, M. B. Sullivan, W. Zheng, and D. Skarlatos, "A scalable multi-gpu framework for encrypted large-model inference," *arXiv preprint arXiv:2512.11269*, 2025.

[14] J. Kim, S. Kim, J. Choi, J. Park, D. Kim, and J. H. Ahn, "Sharp: A short-word hierarchical accelerator for robust and practical fully homomorphic encryption," in *Proceedings of the 50th Annual International Symposium on Computer Architecture*, 2023, pp. 1–15.

[15] J. Kim, G. Lee, S. Kim, G. Sohn, M. Rhu, J. Kim, and J. H. Ahn, "Ark: Fully homomorphic encryption accelerator with runtime data generation and inter-operation key reuse," in *2022 55th IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE, 2022, pp. 1237–1254.

[16] S. Kim, J. Kim, M. J. Kim, W. Jung, J. Kim, M. Rhu, and J. H. Ahn, "Bts: An accelerator for bootstrappable fully homomorphic encryption," in *Proceedings of the 49th annual international symposium on computer architecture*, 2022, pp. 711–725.

[17] J. Lin, L. Liang, Z. Qu, I. Ahmad, L. Liu, F. Tu, T. Gupta, Y. Ding, and Y. Xie, "Inspire: in-s torage p rivate i nformation re trieval via protocol and architecture co-design," in *Proceedings of the 49th Annual International Symposium on Computer Architecture*, 2022, pp. 102–115.

[18] Y. Nam, A. Moitra, Y. Venkatesha, X. Yu, G. De Micheli, X. Wang, M. Zhou, A. Vega, P. Panda, and T. Rosing, "Rhychee-fl: Robust and efficient hyperdimensional federated learning with homomorphic encryption," in *2025 Design, Automation & Test in Europe Conference (DATE)*. IEEE, 2025, pp. 1–7.

[19] T. Prantl, L. Horn, S. Engel, L. Iffländer, L. Beierlieb, C. Krupitzer, A. Bauer, M. Sakarvadia, I. Foster, and S. Kounev, "De bello homomorphico: Investigation of the extensibility of the openfhe library with basic mathematical functions by means of common approaches using the example of the ckks cryptosystem," *International Journal of Information Security*, vol. 23, no. 2, pp. 1149–1169, 2024.

[20] N. Samardzic, A. Feldmann, A. Krastev, N. Manohar, N. Genise, S. Devadas, K. Eldefrawy, C. Peikert, and D. Sanchez, "Craterlake: a hardware accelerator for efficient unbounded computation on encrypted data," in *Proceedings of the 49th Annual International Symposium on Computer Architecture*, 2022, pp. 173–187.

[21] X. Wang, T. Zhang, K. Fan, A. Vega, M. Zhou, and T. Rosing, "Pathe: A privacy-preserving database pattern search platform with homomorphic encryption," in *2026 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2026, pp. 1–6.

[22] X. Wang, M. Zhou, G. De Micheli, Y. Nam, S. Pinge, A. Vega, and T. Rosing, "Pathe: A privacy-preserving database pattern search platform with homomorphic encryption," in *IEEE/ACM International Conference on Computer-Aided Design*, 2025.

[23] M. Zhou, Y. Nam, X. Wang, Y. Lee, C. Wilkerson, R. Kumar, S. Taneja, S. Mathew, R. Cammarota, and T. Rosing, "Ufc: A unified accelerator for fully homomorphic encryption," in *2024 57th IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE, 2024, pp. 352–365.

[24] M. Zuber and R. Sirdey, "Efficient homomorphic evaluation of k-nn classifiers," *Proceedings on Privacy Enhancing Technologies*, 2021.