

Letter from the Editor-in-Chief

Traditional data management systems built on relational models have, for decades, provided the bedrock of reliable data storage and efficient queries. Yet their inherently rigid structure can constrain advanced analytics and sophisticated automation—limitations that have grown more pronounced as applications demand ever richer insight from increasingly varied data. Large Language Models (LLMs) promise a disruptive new chapter, offering a deep, context-aware grasp of both data and user intent. By moving beyond the static assumptions of relational schemas, LLMs can potentially transform how we query, integrate, and analyze data, easing the manual labor of transformations and freeing us to explore more intuitive, semantic ways of managing information.

We are, however, still in the early days of tapping these capabilities for real-world data systems. It remains unclear which designs or methodologies will prove most effective, how well these models will scale, or the degree to which they can seamlessly merge with long-standing DBMS technologies. Some paths will lead to breakthroughs in usability and analytics power; others may reveal unforeseen complexities that challenge our notions of correctness, performance, and trust. It is precisely this uncertainty—coupled with the tremendous opportunities—that makes LLMs such an exciting frontier for data management research and practice.

This issue explores how LLMs open transformative possibilities in data management. For example, instead of relying on tightly defined schemas or specialized data formats, LLMs enable a more flexible approach that blends both structured and unstructured information. Their capacity for richer semantic understanding can alleviate long-standing burdens in tasks like data integration and data quality, where mismatched schemas and sparse documentation often lead to labor-intensive fixes.

Equally important, LLMs encourage us to rethink analytics from a more holistic perspective. They provide a single interpretive layer for text, time-series, images, relational tables, and other data types, allowing diverse inputs to be processed with greater coherence and context-awareness. This unifying capability can make data exploration more intuitive, especially in large-scale environments where data is varied and often lacks a strict organizational scheme. In doing so, LLMs help unlock insights that might otherwise remain obscured, fostering a new level of accessibility and analytic power for modern data systems.

The contributors to this issue offer important glimpses into what the age of LLMs and generative AI might look like for data management. Trummer’s work on operator-level customizations reveals how systems can adapt to shifting query workloads in near-real time, while Freire et al. demonstrate how complex data integration tasks can be simplified by deeper semantic modeling. Lin et al. push further by showing how LLMs essentially behave like experienced data engineers, orchestrating cleaning and reconciliation efforts that were once painfully manual. Wang et al. illustrate how large and unstructured data repositories can be tamed through language-driven interfaces, making analytics more flexible and intuitive. Throughout these pages, the central lesson is that we are beginning to glimpse a future where LLMs help us rethink the very nature of data systems: how queries are processed, how data is unified, and how organizations can derive insights from an ever-expanding universe of information.

It is my pleasure to extend special thanks to Dr. Steven Euijong Whang, the Associate Editor who brought together these forward-thinking works. As you delve into the research presented here, I invite you to envision how data management might evolve in an era where generative AI not only understands our questions and the data itself, but also orchestrates every necessary process through a flexible suite of tools. The opportunities are immense, and I look forward to the breakthroughs that our field is poised to achieve.

Haixun Wang
EvenUp