**How do Query Languages work with Storage and Retrieval?**

Chase Rawson

College of Professional Studies, University of New Hampshire

DAT 670: Advanced Data Analytics

Dr. Tony Sulpizio

November 13, 2023

Chapter 3 of the textbook Designing Data-Intensive Applications, titled Storage and Retrieval, centers on unraveling the complexities of how databases manage the intricate processes of data storage and retrieval. The chapter also dives into fundamental questions surrounding what transpires during data storage and how databases respond to subsequent queries.

Databases, or storage engines, are categorized either for transaction processing (OLTP) or optimized for analytics (OLAP). In an OLTP system, requests are handled by selectively engaging with limited records per query, utilizing keys and indexes. Less familiar to end users, data warehouses and analytic systems handle fewer queries but with intensive processing demands, relying on disk bandwidth instead of seek time. Within an OLTP system are two design ideas: the log-structured school, allowing file appending and deletion but not updating written files, and the update-in-place school, treating the disk as pages to be overwritten. (Kleppmann, 2017)

The information in Chapter 3, which dives into how databases handle storage and retrieval, relates to the content of Chapter 2, where a diverse array of data models are introduced. Chapter 2, titled Data Models and Query Languages, provides a broad overview of various data models, emphasizing that no single model fits all applications. It traces the historical evolution from the hierarchical to the relational model, which addressed the limitations of representing many-to-many relationships. (Kleppmann, 2017)

Chapter 3 concludes with a brief exploration of intricate indexing structures and databases optimized for in-memory data storage. A shift occurs from the internal workings of storage engines to the high-level architecture of data warehouses. This shift sets the contrasting nature of analytic workloads and OLTP, emphasizing the importance of compact data encoding to minimize disk reads during sequential scans.

**References**

Kleppmann, M. (2017). *Designing Data-Intensive Applications*. O’Reilly.