A Methodology for Guiding Polyglot Persistence Design

Hudson Afonso Batista da Silva[1,2], **Ronaldo dos Santos Mello**[2]

[1] Instituto Federal de Educação, Ciência e Tecnologia do Pará (IFPA)
Marabá, PA – Brazil

[2]Depto. de Informática e Estatística (INE) – Univ. Federal de Santa Catarina (UFSC)
Florianópolis, SC – Brazil

hudson.silva@ifpa.edu.br, r.mello@ufsc.br

Abstract. *Data-intensive applications often combine different database models, such as relational, document and graph stores. This strategy, known as polyglot persistence, must decide which model is more suitable for each domain data. This is a complex decision in early design stages as it usually depends on factors such as the conceptual data structure and the expected workload. This ongoing PhD research proposes a novel methodology to support polyglot database design from conceptual schemas, workload estimates, and non-functional requirements. The proposal extracts structural and access-related features, applies preliminary decision rules, and records the decision rationale in a polyglot metadata catalog. Preliminary results with relational and document-oriented data models indicate that data model suitability depends on the interaction between schema structure, query pattern and nesting strategy.*

Resumo. *Aplicações intensivas em dados frequentemente combinam diferentes modelos de bancos de dados, como relacional, documento e grafo. Essa estratégia, conhecida como persistência poliglota, deve decidir qual modelo é mais adequado para cada dado do domínio do problema. Esta é uma decisão complexa nas fases iniciais de projeto, pois depende geralmente de fatores como a estrutura conceitual dos dados e a carga de trabalho estimada. Esta pesquisa em andamento em nível de doutorado propõe uma metodologia inédita para apoiar um projeto de persistência poliglota a partir de um esquema conceitual, estimativas de workload e requisitos não funcionais. A proposta extrai características estruturais e de acesso, aplica regras preliminares de decisão e registra as justificativas em um catálogo de metadados poliglota. Resultados preliminares com modelos de dados relacional e orientado a documentos indicam que a adequação do modelo depende da interação entre estrutura do esquema, padrão de consulta e estratégia de aninhamento.*

General Information. Level: Doutorado. **Student:** Hudson Afonso Batista da Silva. **Advisor:** Ronaldo dos Santos Mello. **Program:** Graduate Program in Computer Science - Federal University of Santa Catarina (UFSC). **Admission:** August 2023. **Qualifying exam:** February 2026. **Expected defense:** August 2027. **Completed stages:** mandatory credits, bibliographic review, problem statement and qualifying exam. **Related publications:** [Silva et al. 2024].

1. Introduction

Modern data-intensive applications often manage heterogeneous data with different structures, access patterns and consistency requirements. Although relational databases remain

widely used, NoSQL data models such as document, graph, key-value and wide-column stores have become important alternatives for scenarios involving flexible schemas, large-scale data and specialized access patterns [Sadalage and Fowler 2013]. In this context, *polyglot persistence* advocates the use of more than one database model or technology within the same application, so that each part of the domain data can be stored according to its structural and workload characteristics.

However, deciding which data model should be used for each part of the data domain is still a complex design problem. Existing database design processes provide mature guidelines for relational modeling, and several works discuss NoSQL or multi-model database design. Nevertheless, there is still a lack of consolidated methods that support multiple data model selection from early design artifacts, especially when the decision must jointly consider the conceptual schema, the expected workload, as well as non-functional requirements (NFRs) [Holubová et al. 2021, Roy-Hubara et al. 2022].

This problem is relevant because data model decisions are often made before the physical implementation is available. At this stage, the database designer usually holds a conceptual schema (*e.g.*, an Entity–Relationship (ER) schema), a preliminary understanding of the application workload, and a set of NFRs. These inputs are relevant as they may guide database logical design decisions. For example, highly normalized structures and strong consistency may favor the relational model, natural aggregations and hierarchical access may fit well to document-oriented models, data with dense relationships and frequent traversals may be directed to property graph models, and data with predictable direct-access workloads may be designed for key-value or wide-column models. The challenge is to transform this evidence into explicit and reusable decision rules.

This ongoing PhD research addresses this challenge by proposing a novel methodology for guiding polyglot persistence design. It considers the extraction of structural features from an ER schema, as well as a forecast of frequent queries and NFRs, the application of preliminary decision rules for polyglot logical design, and the persistence of the decision rationale in a polyglot metadata catalog. The goal is not to replace benchmark-based validation, but to reduce the number of candidate data models and schema alternatives to be further evaluated.

The current stage of the research includes the definition of initial decision rules and preliminary experiments comparing relational and document-oriented alternatives. These experiments indicate that model suitability depends not only on the selected database paradigm, but also on the interaction between schema structure, query pattern, indexing and nesting strategy. Therefore, the preliminary results are being used to refine the decision rules and to guide the next experimental steps.

The main contributions of this work-in-progress are: (i) a research problem formulation for ER-based polyglot data model selection; (ii) a preliminary methodology that combines conceptual schema features, workload information and NFRs; and (iii) an initial experimental evaluation that intends to refine the proposed decision rules.

2. Related Work

Some studies in the literature regard polyglot persistence modeling. However, they emphasize only some variables related to the decision problem, such as NFRs, estimated workload, or data placement. Table 1 summarizes representative works and highlights

the originality of this research. The analyzed approaches do not consider all the expected inputs we account as relevant for an explainable methodology for early polyglot persistence design, in particular the consideration of a conceptual schema.

Tabela 1. Comparison of representative related work

Work	ER / conceptual schema	Workload	NFRs	Explicit rules	Main difference
Roy-Hubara et al. [Roy-Hubara et al. 2022]	✗	✓	✓	✓	Does not derive rules from conceptual schema structures.
Holubova et al. [Holubová et al. 2021]	✗	✗	✗	✗	Focuses on multi-model representation and research challenges.
Eisenhuth and Jablonski [Eisenhuth and Jablonski 2022]	✗	✓	✓	✓	Does not consider conceptual modeling.
Schaarschmidt et al. [Schaarschmidt et al. 2015]	✗	✗	✓	✓	Considers SLA/schema annotations, not ER-based rule derivation.
This work	✓	✓	✓	✓	**Combines ER schema, workload and NFRs into explicit decision rules.**

Legend: ✓ addressed; ✗ not addressed.

3. Proposed Methodology

On following Wazlawick's methodological guidance for Computer Science research [Wazlawick 2009], this work is characterized as an applied and constructive investigation. It proposes a data design methodology and refines it through literature-based rule synthesis and preliminary benchmark-based experiments. The proposed methodology aims to support early-stage polyglot persistence design by analyzing three complementary inputs: a conceptual (ER-based) schema, the expected workload in terms of expected frequent queries, and NFRs. It considers design-time information to identify candidate data models before benchmark-based validation, making the initial decision process more explicit and reducing the space of alternatives to be benchmarked.

Figure 1 shows the main stages of the proposal. In Stage 1, the designer provides the aforementioned inputs. In Stage 2, the methodology analyzes these inputs as decision criteria. From the ER schema, it extracts structural properties such as entity types, relationship cardinalities, relationship density, normalization level, containment structures and highly connected regions. From the workload, it considers access patterns such as point lookups, joins, aggregations, traversals, scans and query frequency. From the NFRs, it considers requirements such as consistency, integrity, scalability, availability, flexibility, and performance. In Stage 3, the decision rules indicate suitable candidate logical data models. Our focus is on relational and NoSQL data models.

It is expected that Stage 2 works as follows. From the ER schema, the methodology identifies candidate root entities, cardinalities, relationship paths, containment-like structures and possible nesting boundaries. From the workload, it classifies the dominant access pattern of each query, such as scan, lookup, filtering over related entities, grouping or aggregation. NFRs are then used as decision criteria. For example, a document data model can be considered a candidate when the ER structure allows meaningful nesting and the workload can benefit from local access to related data.

By now, we are defining decision rules as preliminary hypotheses derived from two sources: (i) a literature-based analysis of studies comparing relational and NoSQL

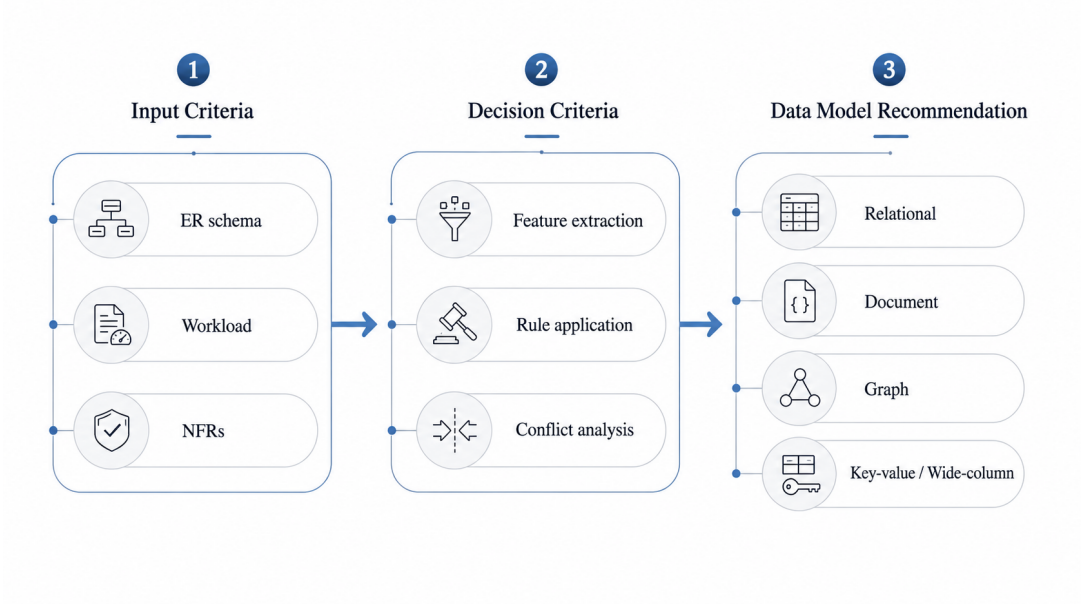

Figura 1. Three-stage overview of the proposed methodology

models; and (ii) the structural and workload features extracted from the application design. Therefore, the rules are not created arbitrarily. They are first synthesized from qualitative evidence reported in previous SQL vs. NoSQL performance evaluations, and are now being refined through our own experiments.

For instance, studies on document-oriented database design suggest advantages in scenarios where schemas are transformed into denormalized, document-centric structures and queries avoid explicit relational joins [Maia et al. 2017, Antas et al. 2022, Ferreira et al. 2025, Samanta and Chaki 2023]. Studies on key-value databases report benefits under strict structural and query constraints, mainly when access is limited to direct key-based operations [Gembalczyk et al. 2017, Leung and Zhou 2016, Klein et al. 2015]. Similarly, studies comparing graph and relational models provide evidence that graph databases can be suitable when the workload is relationship-centric and requires traversals over highly connected data [Baker Effendi et al. 2020, Hernández et al. 2016, Monteiro et al. 2023, Kotiranta et al. 2022]. Table 2 summarizes the current decision rule directions in the literature.

These rules do not produce a final physical design. Instead, they reduce the design space by indicating which data models should be considered and experimentally evaluated. When multiple rules indicate different candidate models for the same schema region, the methodology treats this situation as a design conflict and keeps the competing candidates for further evaluation.

Tabela 2. Preliminary decision rules for data model selection

Candidate model	ER indicators	Workload indicators	NFR indicators
Relational	Structured entities; integrity constraints; N:N relationships; normalized regions	Joins; selective lookups; transactional updates; precise constraints	Consistency; integrity; transactions
Document	Hierarchical or aggregate-like structures; containment; related entities often accessed together	Root-based reads; localized queries; filters over denormalized documents; reduced joins	Flexibility; read performance; schema evolution
Graph	Highly connected entities; dense relationships; relationship-centric regions	Path queries; traversals; neighborhood exploration; recursive or multi-hop access	Relationship exploration; expressiveness; flexibility
Key-value	Identifiable records; limited relationship dependency; flattened structures	Direct key lookup; session/cache-like access; simple read/write operations	Low latency; scalability; availability
Wide-column	Sparse or high-volume records; attribute families; partitionable data	High-volume writes; range scans; partition-based analytical access; append-oriented workloads	Scalability; write throughput; availability

4. Preliminary Evaluation

A preliminary evaluation focused on decision rules for the document data model. It followed an incremental refinement process and considered a polystore benchmark – *M2Bench* [Kim et al. 2022]. First, we evaluated the original M2Bench logical design, which includes a document modeling for part of an *e-commerce* schema. We then compared this design with a relational representation for queries that, according to previous SQL vs. document studies, could favor document-oriented execution. The initial tests did not show clear advantages for the considered document database (MongoDB), suggesting that query pattern alone was not sufficient to explain document model performance.

Next, we analyzed the workload and schema design adopted by the related work of Corovcak and Koupil [Corovcák and Koupil 2025], where *customer, product* and *brand* information are embedded into an *order* document, and MongoDB achieved better results for some queries. It suggested that the relevant factor was not only the query type, but its interaction with the document schema. We then compared the two MongoDB schemas: the M2Bench document schema and the nested schema of Corovcak and Koupil. The workload contained six queries: a full scan over *orders* (Q1); *orders* counting (Q2); *orders* containing a specific *product* (Q3); *order* details (Q4); *orders* without expensive *items* (Q5); and *orders* counting per *customer* (Q6). Table 3 reports the execution time for scale factor 1 (SF = 1), with the best time for each query shown in bold.

Tabela 3. Preliminary comparison between two MongoDB document schemas

Query	Workload pattern	M2Bench (ms)	Nested (ms)
Q1	Order scan	**170**	307
Q2	Count orders	**41**	49
Q3	Orders by product	193	**84**
Q4	Order details	**36**	38
Q5	Orders without expensive items	216	**103**
Q6	Orders per customer	**96**	132

The results reinforce the need to consider the logical document schema together with the workload, since document nesting is not universally beneficial. The original M2Bench schema performs better for scans, counts, direct order retrieval, and grouping

queries (Q1, Q2, Q4 and Q6). A likely explanation is the smaller document size, which may reduce the amount of data read during scans and aggregations. In contrast, the nested schema performs better for queries that filter data from related entities (Q3 and Q5). In these cases, embedding related information can improve data locality and reduce the need for expensive aggregation stages.

These findings help refine the rules for the document data model. This model should not be recommended solely because entities are frequently accessed together. The degree of nesting, expected query pattern and document size must also be considered. It then suggested a more accurated decision rule: document nesting is useful when queries benefit from data locality over related entities, but it may hurt workloads dominated by large scans, broad aggregations, or access to only a small subset of attributes.

5. Discussion

The preliminary results suggest that data model selection should not be based only on the general characteristics of a database paradigm. Even within the same paradigm, such as document databases, different schema designs may lead to different performance behaviors. In our experiments, document nesting improved queries that benefit from data locality over related entities, but introduced overhead for scans and broad aggregations.

This reinforces that the decision rules must jointly consider schema structure, query pattern, data volume and NFRs. Therefore, the methodology should not produce an absolute decision without context. Instead, it should reduce the design space, indicate competing candidate logical data models when necessary, and guide the designer toward the alternatives that should be experimentally evaluated.

6. Conclusion and Future Work

This paper presents an ongoing PhD research on polyglot persistence modeling from early design artifacts. The proposed methodology combines ER schema structures, workload information and NFRs to derive preliminary decision rules for the selection of relational, document, graph, key-value, and wide-column data models for further logical data design. Preliminary experiments with relational and document alternatives suggest that model suitability depends on the interaction between schema structure, query pattern, indexing and nesting strategy.

As future work, we intend to refine the decision rules for all considered data models, extend the experimental evaluation to more workloads and scale factors, define conflict-resolution strategies when different rules indicate competing data models, and develop the metadata catalog.

Referências

Antas, J., Rocha Silva, R., and Bernardino, J. (2022). Assessment of sql and nosql systems to store and mine covid-19 data. *Computers*, 11(2):29.

Baker Effendi, S., van der Merwe, B., and Balke, W.-T. (2020). Suitability of graph database technology for the analysis of spatio-temporal data. *Future Internet*, 12(5):78.

Corovcák, M. and Koupil, P. (2025). SQL vs nosql: Six systems compared. In Mannion, M., Männistö, T., and Maciaszek, L. A., editors, *Proceedings of the 20th International*

Conference on Evaluation of Novel Approaches to Software Engineering, ENASE 2025, Porto, Portugal, April 4-6, 2025, pages 41–53. SCITEPRESS.

Eisenhuth, P. and Jablonski, S. (2022). Knowledge-based recommendation for polyglot persistence. In *CDMS@VLDB*.

Ferreira, S., Mendonça, J., Nogueira, B., Tiengo, W., and Andrade, E. (2025). Benchmarking consistency levels of cloud-distributed nosql databases using ycsb. *IEEE Access*.

Gembalczyk, D., Schuhknecht, F. M., and Dittrich, J. (2017). An experimental analysis of different key-value stores and relational databases. In *Datenbanksysteme für Business, Technologie und Web*, pages 351–360. Gesellschaft für Informatik.

Hernández, D., Hogan, A., Riveros, C., Rojas, C., and Zerega, E. (2016). Querying wikidata: Comparing sparql, relational and graph databases. In *International Semantic Web Conference*, pages 88–103. Springer.

Holubová, I., Contos, P., and Svoboda, M. (2021). Multi-model data modeling and representation: State of the art and research challenges. In *Proceedings of the 25th International Database Engineering & Applications Symposium*, pages 242–251.

Kim, B. et al. (2022). M2bench: A database benchmark for multi-model analytic workloads. *Proceedings of the VLDB Endowment*, 16(4):747–759.

Klein, J. et al. (2015). Application-specific evaluation of nosql databases. In *2015 IEEE International Congress on Big Data*, pages 526–534. IEEE.

Kotiranta, P., Junkkari, M., and Nummenmaa, J. (2022). Performance of graph and relational databases in complex queries. *Applied Sciences*, 12(13):6490.

Leung, F. and Zhou, B. (2016). Performance evaluation of twitter datasets on sql and nosql dbms. In *Web Intelligence*, volume 14, pages 275–286. SAGE Publications.

Maia, D. C. M. et al. (2017). Performance analysis on voluntary geographic information systems with document-based nosql database. In *Developments and Advances in Intelligent Systems and Apps.*, pages 181–197. Springer.

Monteiro, J., Sá, F., and Bernardino, J. (2023). Experimental evaluation of graph databases: Janusgraph, nebula graph, neo4j, and tigergraph. *Applied Sciences*, 13(9):5770.

Roy-Hubara, N., Shoval, P., and Sturm, A. (2022). Selecting databases for polyglot persistence applications. *Data & Knowledge Engineering*, 137:102022.

Sadalage, P. J. and Fowler, M. (2013). *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*. Pearson Education.

Samanta, A. K. and Chaki, N. (2023). An enumerated analysis of nosql data models using statistical tools. *Innovations in Systems and Software Engineering*, 19(1):5–14.

Schaarschmidt, M., Gessert, F., and Ritter, N. (2015). Towards automated polyglot persistence. In *Datenbanksysteme für Business, Technologie und Web (BTW 2015)*, pages 73–82. Gesellschaft für Informatik eV.

Silva, H., Bornia, L., and Mello, R. (2024). Um Estudo sobre Modelagem Poliglota de Dados. In *Anais da XIX Escola Regional de Banco de Dados*, pages 11–20. SBC.

Wazlawick, R. S. (2009). *Metodologia de Pesquisa para Ciência da Computação*, volume 2. Elsevier Rio de Janeiro.