

# **Uma Revisão Sistemática Simplificada e Análise de Estudos sobre Evolução de Esquemas em Aplicações de Persistência Poliglota**

**Lilian N. Araujo<sup>1,2</sup>, Hudson Afonso Batista da Silva<sup>1,3</sup>, Ronaldo S. Mello<sup>1</sup>**

<sup>1</sup>Programa de Pós Graduação em Ciência da Computação (PPGCC)

Departamento de Informática e Estatística (INE)

Universidade Federal de Santa Catarina (UFSC)

Florianópolis – SC – Brasil

<sup>2</sup>Instituto Federal do Paraná (IFPR)

Palmas – PR – Brasil

<sup>3</sup>Instituto Federal do Pará (IFPA)

Marabá – PA – Brasil

[lilian.araujo@ifpr.edu.br](mailto:lilian.araujo@ifpr.edu.br), [hudson.silva@ifpa.edu.br](mailto:hudson.silva@ifpa.edu.br), [r.mello@ufsc.br](mailto:r.mello@ufsc.br)

**Abstract.** *The evolution of schemas in polyglot persistence projects is described in the literature as a research gap due to the challenges posed by the different data models used and the concern of maintaining data integrity and consistency after changes. This study presents a simplified systematic literature review, highlighting technological advancements aimed at simplifying schema evolution in modern and complex systems while also emphasizing the scientific relevance of the issue, confirming the scarcity of specific studies on schema evolution in polyglot persistence applications. Works addressing different database architectures (hybrid, multimodel, polystore) and data models (relational, document, and graph) are analyzed. A comparative analysis of the studies reveals that strategies for schema evolution in polyglot architectures make use of unified data model abstractions, with operations embedded in proprietary languages or systems.*

**Resumo.** *A evolução de esquemas em projetos de persistência poliglota é descrita na literatura como uma lacuna de pesquisa, devido aos desafios frente aos diferentes modelos de dados que utiliza e a preocupação em manter a integridade e consistência dos dados após mudanças. Esse estudo apresenta uma revisão sistemática simplificada da literatura, destacando avanços tecnológicos para simplificar a evolução de esquemas em sistemas modernos e complexos, além de evidenciar a relevância científica da problemática abordada, confirmando a escassez de estudos específicos sobre evolução de esquemas em aplicações de persistência poliglota. Trabalhos que abordam diferentes arquiteturas de bancos de dados (híbrida, multimodelo, polystore) e modelos de dados (relacional, documento e grafo) são analisados. A análise comparativa sobre os estudos encontrados revela que as estratégias para a evolução de esquemas em arquiteturas poliglotas utilizam abstrações unificadas de modelos de dados, com operações embutidas em linguagens ou sistemas proprietários.*

## 1. Introdução

A era do *Big Data* contribuiu positivamente em diversas áreas da Computação, mas com as oportunidades surgem também desafios que são temas de inúmeras pesquisas. Na área de Banco de Dados (BD), em particular, o surgimento dos sistemas de gerência de BD (SGBDs) NoSQL foi um grande avanço nesse contexto, permitindo gerenciar um volume de dados heterogêneos que cresce de maneira exponencial. A tecnologia NoSQL vem para complementar os tradicionais BDs relacionais, pois permite armazenar dados semiestruturados e não estruturados, resolvendo questões de flexibilidade e escalabilidade que são fatores limitantes nos BDs relacionais [Lajam and Mohammed 2022]. Entretanto, o formato de armazenamento estruturado não deixou de ser usado e mais recentemente evoluiu para arquiteturas de BDs denominados NewSQL, que mantêm as garantias ACID dos BDs relacionais e garantem o desempenho escalável dos BDs NoSQL utilizando SQL como interface de acesso [Murazzo et al. 2019].

Atualmente as aplicações estão em constante atualização principalmente quando é necessário o armazenamento e manipulação de dados em diferentes modelos de representação. Com o desenvolvimento de SGBDs chamados *multimodelo* é possível armazenar e gerenciar vários tipos de dados dentro de um único BD [Lu and Holubová 2019]. Esses sistemas combinam modelos relacionais e não relacionais, facilitando a persistência poliglota em contextos de Big Data. O objetivo dos SGBDs multimodelo é atender necessidades específicas e complexas de gestão de dados por aplicações [Kosmerl et al. 2020].

A *persistência poliglota*, por sua vez, é um conceito que vem se consolidando com o rápido avanço das tecnologias da informação, permitindo que diferentes SGBDs com diferentes modelos de dados sejam utilizados por uma mesma aplicação para atender a necessidades específicas de desempenho, escalabilidade ou modelagem de dados [Lajam and Mohammed 2022]. Uma das dificuldades dessa abordagem é a forma como os esquemas de dados podem se desenvolver ao longo do tempo de forma a impactar minimamente na aplicação. Na literatura é possível encontrar diversas ferramentas e métodos para evolução de esquemas em BD de maneira isolada. Ter múltiplos modelos de dados que necessitam de alterações na estrutura de dados pode comprometer a integridade, consistência e manutenção dos dados de uma aplicação, sendo um desafio em aberto na literatura no contexto da persistência poliglota [Hausler 2024].

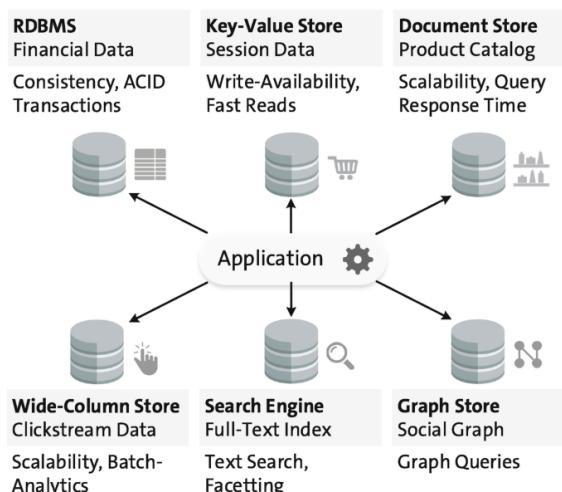
Assim sendo, este artigo apresenta uma revisão sistemática simplificada da literatura sobre a evolução de esquemas em aplicações de persistência poliglota. Não foi encontrada nenhuma revisão similar na literatura. A Seção 2 deste artigo contextualiza evolução de esquemas e persistência poliglota. A Seção 3 descreve as etapas da revisão sistemática simplificada e a Seção 4 uma visão geral dos estudos selecionados. A Seção 5 apresenta uma comparação destes estudos e a Seção 6 é dedicada à conclusão.

## 2. Evolução de Esquemas e Persistência Poliglota

As motivações por trás de modificações em esquemas de dados decorrem da necessidade de reconsiderar a estrutura do BD, as necessidades a serem atendidas e o ambiente computacional. Um BD pede ajustes sempre que atualizações nos requisitos de dados da aplicação acontecem, sendo um processo normal sempre que tecnologias mudam, novos processos surgem, dentre outros [Andany et al. 1991].

A evolução de esquemas de BDs se refere a mudanças em formatos, estruturas e semânticas dos dados ao longo do tempo, exigindo muitas vezes o gerenciamento de versões do esquema. Esse processo pode levar a desafios na reconstrução dos estados anteriores do BD, impactando na confiabilidade de resultados de buscas [Auge and Heuer 2022]. É um desafio grande em particular para os BDs NoSQL, que nem sempre possuem um esquema definido *a priori* ou um esquema fixo [Chillón et al. 2022].

A utilização de diferentes modelos de dados por um SGBD, ou vários SGBDs interconectados, para atender a diversas exigências de armazenamento e acesso é denominada *Persistência Poliglota*. O termo "poliglota" sugere a capacidade de vários SGBDs heterogêneos colaborarem entre si [Lajam and Mohammed 2022]. Além disso, essa abordagem é descrita como uma estratégia heterogênea de persistência [Sadlage and Fowler 2013], uma vez que depender de uma tecnologia de BD para todas as demandas pode levar a soluções com baixo desempenho. A Figura 1 mostra um exemplo de um ambiente de persistência poliglota em que diferentes modelos de dados são utilizados em módulos específicos da aplicação.



**Figure 1. Exemplo de aplicação com persistência poliglota [Gessert et al. 2020].**

Um conceito fortemente associado à persistência poliglota são os *SGBDs poly-store*. Um SGBD polystore é um sistema que fornece acesso integrado e transparente a várias fontes de dados heterogêneas, ou seja, em um único ambiente utiliza-se diferentes tipos e formatos de dados. Esses sistemas usam uma ou várias interfaces para permitir consultas [Glake et al. 2022]. Atuam como uma camada lógica unificada sobre diferentes modelos de dados, permitindo a integração de consultas sem a necessidade de mover dados. Trata-se de interoperabilidade e não da integração nativa que caracteriza os SGBDs multimodelo.

Outro conceito associado são os *SGBDs híbridos*, que em alguns casos são *sistemas fortemente acoplados*, ou seja, um sistema único capaz de aplicar leituras e escritas em múltiplos BDs heterogêneos normalmente não autônomos, e outros são *sistemas fracamente acoplados*, correspondendo a uma rede de armazenamento de SGBDs autônomos, onde um componente mediador tem direitos de leitura, mas não direitos para escrever dados ou reconfigurar os SGBDs individuais [Glake et al. 2022]. Em resumo, um SGBD híbrido possui uma camada de armazenamento que suporta múltiplos mode-

los de dados, uma camada de processamento que provê acesso integrado a dados representados nesses modelos, e uma interface unificada para a gestão de todos esses dados heterogêneos.

Como mencionado anteriormente, temos ainda o conceito de *SGBD multimodelo*, que é um único SGBD com suporte de acesso e armazenamento a diversos modelos de dados. Esta arquitetura de SGBD geralmente adota uma interface de acesso única para a gestão dos dados, muitas vezes denominada de linguagem multimodelo [Lu and Holubová 2019].

Apesar desses conceitos não serem consenso na literatura, nesse novo paradigma da persistência poliglota os administradores de BDs devem ter múltiplas habilidades para aprender como tecnologias recentes de gerência de dados, como os SGBDs NoSQL, funcionam, para manipular os dados dentro e para fora desses SGBDs.

### 3. Revisão Sistemática Simplificada da Literatura

Esse trabalho apresenta os avanços tecnológicos encontrados na literatura referente à evolução de esquemas no contexto de persistência poliglota. Para alcançar esse objetivo foi adotado grande parte das etapas de um guia de revisão sistemática popular na literatura [Kitchenham 2004]. O planejamento da revisão incluiu a identificação da necessidade da pesquisa, a criação de um protocolo de revisão e a definição das etapas associadas à identificação, seleção de trabalhos relacionados e extração dos dados.

A evolução de esquemas é um tópico significativo, especialmente ao levar em conta as diversas estruturas de BDs presentes em diversos trabalhos da área. Essa importância contribuiu diretamente para o desenvolvimento do protocolo de pesquisa relacionado à evolução de esquemas em aplicações de persistência poliglota, e algumas questões de pesquisa foram formuladas:

- **Q1:** Quais são os fatores que influenciam a evolução de esquemas em um ambiente poliglota?
- **Q2:** Como são projetadas as arquiteturas que suportam evolução de esquemas de forma flexível e escalável?
- **Q3:** Quais modelos de dados foram avaliados e/ou considerados nos estudos relacionados?

Na sequência, foi definida uma *string* de busca para a coleta de trabalhos relacionados. O objetivo aqui é definir as palavras-chave a serem utilizadas para busca de estudos em diferentes bases de dados bibliográficos. A string de busca definida é a seguinte:

((“multimodel databases” OR “polystore databases” OR “multi-model databases” OR “polyglot” OR “polyglot persistence” OR “nosql”) AND (“schema evolution” OR “schema modification” OR “schema change” OR “schema propagation”)) AND (E-Publication Date: (01/01/2012 TO 12/31/2024)))

Em algumas bases, os trabalhos também puderam ser filtrados pelo idioma “Inglês” e área da “Ciência da Computação”.

ALL ( “multimodel databases” OR “polystore databases” OR “multi-model databases” OR “polyglot” OR “polyglot persistence” OR “nosql” ) AND ( ALL (

"schema evolution" OR "schema modification" OR "schema change" OR "schema propagation" ) ) AND PUBYEAR > 2011 AND PUBYEAR < 2025 AND ( LIMIT-TO ( SUBJAREA , "COMP" ) ) AND ( LIMIT-TO ( DOCTYPE , "cp" ) OR LIMIT-TO ( DOCTYPE , "ar" ) ) AND ( LIMIT-TO ( LANGUAGE , "English" ) )

A base *ScienceDirect* possui limite para o número de operadores lógicos utilizados na string. Dessa forma, as palavras "*polyglot persistence*" e "*schema modification*" foram removidas como mostrado a seguir:

(( "multimodel databases" OR "polystore databases" OR "multi-model databases" OR "polyglot" OR "nosql" ) AND ( "Schema evolution" OR "schema change" OR "schema propagation" ))

Indexadores como *Google Scholar* e *DBLP* também foram utilizados e foi extraído um (1) trabalho apresentado em conferência. As bases onde as buscas foram realizadas são as seguintes: *ACM*, *IEEE Xplore*, *Springer Link*, *Scopus* e *Science Direct*. O total de artigos extraídos no período de 2012 a 2024 são apresentados na Tabela 1.

**Table 1. Estudos selecionados.**

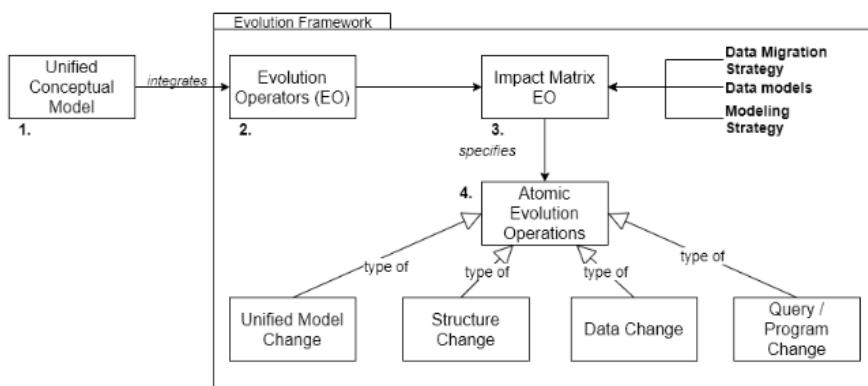
Base	Seleção Inicial	Seleção Final
ACM	89	1
IEEE Xplore	11	1
Science Direct	51	0
Scopus	190	0
Springer Link	92	0
Indexadores		1
<b>Total</b>	<b>433</b>	<b>3</b>

Inicialmente foram 433 artigos extraídos. Após excluir 65 artigos duplicados e usar as palavras *schema evolution* e *polyglot* como critérios de inclusão, e as palavras *review*, *art state*, *systematic review* como critérios de exclusão, restaram 93 artigos. A leitura dos resumos permitiu excluir artigos que não tinham relação com o tema. Evolução de esquemas específico para aplicações de persistência poliglota não foram encontrados. Assim sendo, trabalhos que utilizam mais de um modelo de dados para atender necessidades de uma aplicação foram considerados, sendo eles híbridos, multimodelo e *polystore*. Por fim, foram selecionados 1 artigo da base ACM, 1 artigo da IEEE e um 1 artigo de conferência encontrado em indexador, totalizando 3 estudos.

#### 4. Visão Geral dos Estudos Selecionados

O estudo de [Gobert 2020] propõe a aplicação de operadores genéricos para evolução de esquemas a partir de um modelo conceitual unificado existente para um SGBD híbrido. Um operador pode ser um objeto do domínio como um tipo de entidade, atributo, relacionamento ou identificador. A linguagem de modelagem unificada *TyphonML*, que serve como ponto de partida para o desenvolvimento de SGBDs híbridos, foi utilizada para combinar modelos relacionais e modelos de dados NoSQL, permitindo que os usuários definam as entidades conceituais que o ambiente híbrido deve gerenciar e especifiquem como essas entidades devem ser armazenadas nos modelos de dados alvo.

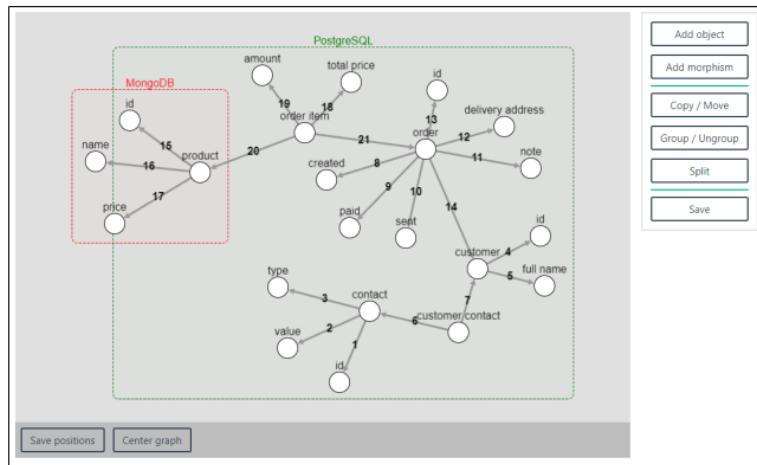
O objetivo do trabalho é, após especificar esses operadores, projetar um *framework* que utiliza a TyphonML como base e é capaz de propagar essas operações para todos os artefatos da aplicação impactados. A Figura 2 apresenta o resumo dos componentes do framework de evolução proposto, onde o modelo conceitual se conecta aos operadores de evolução que podem incluir operações como adicionar, remover ou alterar atributos e entidades. Esses operadores se relacionam com a matriz de impacto responsável por relacionar os operadores de evolução às suas consequências em outros componentes do SGBD. A matriz avalia como cada operador afetará as diferentes partes do SGBD considerando questões como: estratégia de migração de dados, modelo de dados e estratégia de modelagem. A matriz de impacto guia então a aplicação de operações atômicas, que realizam as mudanças específicas em artefatos responsáveis pela gestão dos dados. Cada operador atômico representa uma ação que pode ser aplicada diretamente ao esquema de dados, consultas, atualizações e programas.



**Figure 2. Framework de evolução de esquemas [Gobert 2020].**

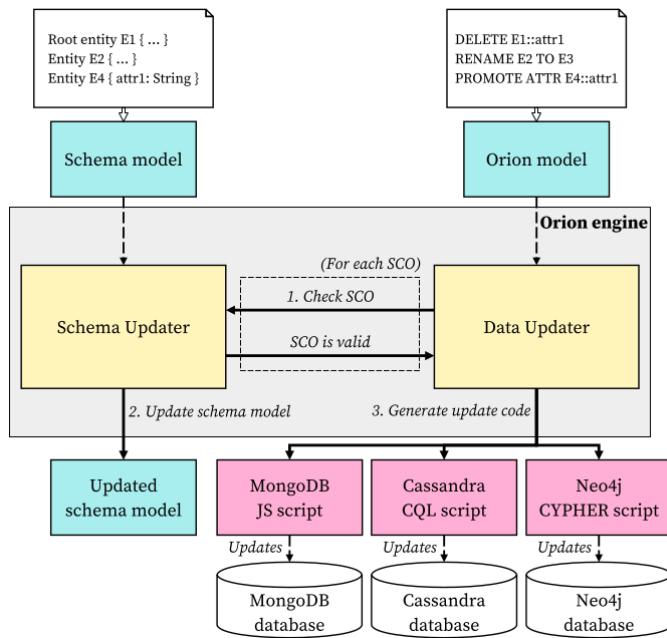
O trabalho de [Bártík et al. 2024] apresenta uma abordagem para o gerenciamento da evolução de dados em um BD multimodelo, onde o desafio está em refletir as mudanças de requisitos do usuário nas estruturas de dados. Ele propõe uma denominada *representação categórica* de dados, que permite modelar e gerenciar dados multimodelo de forma unificada. Essa representação baseia-se na teoria das categorias, onde os dados são organizados como um grafo, com objetos representando entidades, atributos e relacionamentos, enquanto os morfismos definem as conexões entre esses elementos. Essa representação facilita a visualização e a manipulação dos dados combinados, permitindo a integração de diferentes tecnologias (BDs relacionais, documentos e grafos). A categorização dos elementos possibilita ainda a definição de identificadores únicos e propriedades, garantindo consistência na representação dos esquemas.

O trabalho utiliza a *MM-Evocat*, um protótipo para realizar a gestão da evolução de BDs multimodelo pelos usuários administradores de BD e validar o gerenciamento de vários SGBDs como o PostgreSQL, Neo4j e MongoDB. Testes envolveram a migração de dados entre modelos relacionais e de documentos, simulando um cenário de pedidos de clientes como mostra a Figura 3, que apresenta a interface gráfica da *MM-Evocat*.



**Figure 3. Ferramenta MM-Evocat [Bártik et al. 2024].**

A proposta de [Chillón et al. 2024] é voltada à evolução de esquemas de forma genérica para modelos de dados NoSQL do tipo documento, colunar e grafo. Ela define uma taxonomia de operações de mudança de esquema com base em um modelo de dados unificado denominado *U-Schema*, que pode representar tanto esquemas conceituais ou lógicos, seguindo uma arquitetura de SGBD Polystore. Ela utiliza também a linguagem *Orion* que permite a definição de mudanças de esquema em vários SGBDs NoSQL. Uma visão geral da arquitetura da proposta é mostrada na Figura 4.



**Figure 4. Arquitetura da proposta [Chillón et al. 2024].**

A arquitetura da proposta possui 2 componentes: *Atualizador de Esquema* e *Atualizador de Dados*. Quando um *script* Orion é processado, cada operação de mudança de esquema (SCO) passa por um fluxo de 3 etapas. Primeiro, o Atualizador de Dados envia

uma solicitação ao Atualizador de Esquema para validar se uma operação de evolução de esquema pode ser aplicada com segurança. Essa verificação inclui a compatibilidade do BD e a consistência estrutural do esquema. Caso a validação seja aprovada, o Atualizador de Esquema modifica o modelo U-Schema para refletir a mudança no esquema. Em seguida, o Atualizador de Dados gera comandos específicos para cada SGBD, aplicando as modificações nos dados armazenados. Esse processo garante que mudanças no esquema sejam acompanhadas de atualizações consistentes nos dados.

## 5. Análise dos Estudos Selecionados

A Tabela 2 apresenta uma comparação dos estudos selecionados. A análise comparativa dos estudos é realizada com base nas respostas às questões de pesquisa da revisão sistemática, conforme segue.

**Table 2. Comparação dos Estudos Selecionados.**

Estudo	Arquitetura	Modelos de Dados
[Gobert 2020]	Híbrido	Relacional e NoSQL
[Bártík et al. 2024]	Multimodelo	Relacional, Documento e Grafo
[Chillón et al. 2024]	Polystore	Documento, Colunar e Grafo

Com relação à questão de pesquisa **Q1**, o trabalho de [Bártík et al. 2024] não detalha os fatores que influenciam a evolução de esquemas, enfatizando mais a gestão da evolução dos dados e a manutenção da integridade. Já [Chillón et al. 2024] destaca a diversidade dos modelos de dados, a necessidade de prever e suportar variações estruturais, assim como a automação da evolução do esquema. Por fim, [Gobert 2020] menciona requisitos funcionais e não funcionais, heterogeneidade de dados, duplicação de dados, facilidade de integração e experiência da equipe como fatores influenciadores para a evolução de esquemas.

Com relação à questão de pesquisa **Q2**, quanto às arquiteturas propostas suportarem a evolução de esquemas de forma flexível e escalável, [Bártík et al. 2024] propõe uma representação categórica unificada e operações de modificação de esquema. [Chillón et al. 2024] sugere o uso de um modelo unificado e uma taxonomia de operações de mudança de esquema. Já [Gobert 2020] defende o uso de modelos conceituais unificados, operadores genéricos de evolução e frameworks de evolução. Vale observar também que cada abordagem foi aplicada a uma arquitetura diferente de persistência poliglota.

Por fim, com relação à questão de pesquisa **Q3**, [Bártík et al. 2024] lida com modelos de dados relacional, documento e grafo a partir de uma abstração unificada denominada representação categórica. [Chillón et al. 2024] lida com diversos modelos NoSQL e o modelo unificado U-Schema. Por fim, [Gobert 2020] considera modelos de dados relacionais e NoSQL, tendo como base a linguagem de modelagem unificada TyphonML para a definição de operações de evolução de esquemas.

Uma análise comparativa dos trabalhos revela estratégias aplicadas a diferentes arquiteturas de persistência poliglota para o suporte à evolução de esquemas. A maioria das estratégias se baseia em abstrações unificadas de diversos modelos de dados no sentido de facilitar a gestão da evolução do esquema. Em todos os trabalhos são propostas operações de evolução de esquemas e em alguns casos essas operações estão embutidas

em linguagens de evolução de esquemas. Este suporte se materializa em sistemas proprietários [Bártík et al. 2024, Chillón et al. 2024] ou frameworks [Gobert 2020]. Além disso, os trabalhos consideram os modelos de dados relacional e NoSQL, sendo o trabalho de [Gobert 2020] o mais abrangente.

## 6. Conclusão

Em ambientes de persistência poliglota, esquemas em evolução apresentam desafios à consistência, integridade e adaptabilidade de aplicações centradas em dados. Entretanto, após a realização desta revisão sistemática, percebe-se que a literatura ainda é carente de propostas robustas voltadas à evolução automática ou semiautomática da modelagem e/ou distribuição poliglota dos dados, capazes de adaptar-se a mudanças nos requisitos, como alterações no esquema conceitual ou no workload. Além disso, também adaptar os dados existentes a estas mudanças nos esquemas de dados.

Gerar instruções adequadas a cada SGBD considerado, garantir atomicidade, ser capaz de estimar o efeito das modificações antes que sejam executadas são exemplos de questões a serem tratadas para realizar uma gestão eficiente da evolução de esquemas neste contexto. Além disso, a combinação de linguagens formais e visualizações comparativas do estado anterior e futuro do esquema também pode facilitar a compreensão e a execução de um processo de evolução de esquema para os usuários. Esses pontos são exemplos de investigações futuras nesse tema, que pode se aproveitar ainda da literatura existente para a evolução de esquemas específica para determinados modelos de dados, principalmente o modelo de dados relacional.

## References

- Andany, J., Léonard, M., and Palisser, C. (1991). Management Of Schema Evolution In Databases. In *VLDB*, volume 91, pages 161–170.
- Auge, T. and Heuer, A. (2022). Enhanced Inversion of Schema Evolution with Provenance. *arXiv preprint arXiv:2211.13810*.
- Bártík, J., Koupil, P., and Holubová, I. (2024). Modelling and Evolution Management of Multi-Model Data. In *39th ACM/SIGAPP Symposium on Applied Computing (SAC)*, page 3. ACM.
- Chillón, A. H., Klettke, M., Ruiz, D. S., and Molina, J. G. (2022). A Taxonomy of Schema Changes for NoSQL Databases. *arXiv preprint arXiv:2205.11660*.
- Chillón, A. H., Klettke, M., Ruiz, D. S., and Molina, J. G. (2024). A Generic Schema Evolution Approach for NoSQL and Relational Databases. *IEEE Transactions on Knowledge and Data Engineering*.
- Gessert, F., Wingerath, W., Ritter, N., Gessert, F., Wingerath, W., and Ritter, N. (2020). Polyglot Persistence in Data Management. *Fast and Scalable Cloud Data Management*, pages 149–174.
- Glake, D., Kiehn, F., Schmidt, M., Panse, F., and Ritter, N. (2022). Towards Polyglot Data Stores—Overview and Open Research Questions. *arXiv preprint arXiv:2204.05779*.
- Gobert, M. (2020). Supporting Schema Evolution in Hybrid Database Systems. In *Proceedings of the VLDB 2020 PhD Workshop*. VLDB Endowment. Copyright (C) 2020 for this paper by its authors. Copying permitted for private and academic purposes.

- Hausler, D. (2024). Estimation, Impact and Visualization of Schema Evolution in Graph Databases. In *Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems*, pages 123–129.
- Kitchenham, B. (2004). Procedures for Performing Systematic Reviews. *Keele, UK, Keele University*, 33(2004):1–26.
- Kosmerl, I., Rabuzin, K., and Šestak, M. (2020). Multi-Model Databases - Introducing Polyglot Persistence in the Big Data World. *International Convention on Information and Communication Technology, Electronics and Microelectronics*, pages 1724 – 1729.
- Lajam, O. and Mohammed, S. (2022). Revisiting Polyglot Persistence: From Principles to Practice. *International Journal of Advanced Computer Science and Applications*, 13(5).
- Lu, J. and Holubová, I. (2019). Multi-model databases: A New Journey to Handle the Variety of Data. *ACM Computing Surveys (CSUR)*, 52(3):1–38.
- Murazzo, M., Gómez, P., Rodríguez, N., and Medel, D. (2019). Database NewSQL Performance Evaluation for Big Data in the Public Cloud. In *Cloud Computing and Big Data: 7th Conference, JCC&BD, Revised Selected Papers 7*, pages 110–121. Springer.
- Sadalage, P. J. and Fowler, M. (2013). *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*. Pearson Education.