JIC JORNADA DE INICIAÇÃO CIENTÍFICA E TECNOLÓGICA XV EDIÇÃO
Integridade Científica e Combate à desinformação
20 a 22 de outubro
uffs.edu.br/jic
UNIVERSIDADE FEDERAL DA FRONTEIRA SUL

# PROPOSTA DE UM METAMODELO FEDERADO PARA DATA LAKES

**HALLYSON TAVARES CRUZ[1, 2], GEOMAR SCHREINER[3], DENIO DUARTE[4]**

## 1 Introdução

O cenário atual da tecnologia é marcado por um crescimento exponencial no volume de dados, gerado por uma vasta gama de aplicações centradas em dados. Para gerenciar essa demanda, diversos modelos de bancos de dados surgiram como alternativas ao tradicional modelo relacional, notavelmente os da família NoSQL, que oferecem maior flexibilidade e escalabilidade. Essa diversidade resultou em um ambiente de "persistência poliglota", onde uma única organização pode utilizar múltiplos tipos de bancos de dados — como Grafo, Documento, Chave-Valor, Colunar e Relacional — para atender a diferentes necessidades.

Apesar das vantagens, essa heterogeneidade introduz um desafio significativo: a ausência de uma visão unificada da estrutura dos dados. A natureza com esquemas flexível ou inexistente (schemaless) de muitos sistemas NoSQL, embora benéfica para o desenvolvimento ágil, deixa a análise e a compreensão dos dados armazenados mais complexa. Essa diversidade impede a criação de um mapa coeso dos dados de uma organização, dificultando a integração e a falta de compreensão dos dados.

A falta de um esquema unificado acarreta consequências práticas, como a dificuldade na integração de dados entre sistemas, a complexidade na formulação de consultas que abrangem diferentes fontes e o risco de inconsistências nos dados. Além disso, a capacidade de avaliar a qualidade dos dados é prejudicada, e a compreensão geral do modelo de dados por parte de analistas e desenvolvedores torna-se uma tarefa árdua e propensa a erros.

Para superar esses desafios, propõe-se o desenvolvimento de uma ferramenta que opera sobre um metamodelo unificado, capaz de representar de forma coesa os esquemas extraídos

[1] Graduação em Ciência da Computação, Universidade Federal da Fronteira Sul, campus Chapecó, contato: hallysoncruz4@gmail.com
[2] Grupo de Pesquisa: Inovação e Desenvolvimento Tecnológico.
[3] Doutor, Universidade Federal da Fronteira Sul.
[4] Doutor, Universidade Federal da Fronteira Sul, **Orientador**.

de múltiplas fontes. Embora existam abordagens para extração de esquemas de bancos de dados individuais, como os orientados a documentos ou a grafos, percebe-se uma lacuna na existência de ferramentas que se proponham a unificar os resultados dessas extrações distintas. O presente trabalho visa preencher essa lacuna, apresentando uma ferramenta que mapeia e consolida múltiplos esquemas heterogêneos em uma representação única e padronizada, facilitando a análise entre diferentes sistemas de bancos de dados.

## 2 Objetivos

O objetivo geral deste trabalho é o desenvolvimento de uma ferramenta, denominada Polyschema, capaz de unificar esquemas extraídos de múltiplos bancos de dados heterogêneos em uma representação única e coesa.

## 3 Metodologia

A metodologia partiu da análise de ferramentas e padrões de esquema existentes para os principais paradigmas de banco de dados. Para as fontes de Grafo e Documento, foram utilizadas, respectivamente, as ferramentas de extração GPFuse (CRUZ et al., 2023) e JFUSE (BANHARA et al., 2022). Para Chave-Valor, a abordagem se baseou na saída de uma ferramenta para Redis (SCHNEIDER et al., 2024), enquanto para o modelo Relacional, o próprio script DDL (Data Definition Language) serviu como fonte direta do esquema. Como não foram encontradas ferramentas de extração para o modelo Colunar, foi desenvolvida uma ferramenta própria para este fim, utilizando o HBase como sistema de gerenciamento de banco de dados de referência. A análise conjunta destes cinco formatos de entrada foi fundamental para a definição dos requisitos do metamodelo unificado.

Para o desenvolvimento da ferramenta, foi utilizada a linguagem de programação Python. A arquitetura foi estruturada em três estágios distintos: (i) uma camada de *parsers* especializados, responsáveis por traduzir cada formato de entrada; (ii) uma representação intermediária, implementada com *dataclasses* para garantir uma estrutura de dados bem definida e tipada; e (iii) um gerador de esquema, que converte a representação intermediária para o formato final do metamodelo. Internamente, a manipulação de dados nos *parsers* fez uso de dicionários. Por serem implementados como tabelas *hash*, os dicionários garantem uma busca eficiente, o que é fundamental para a performance do processo de análise e unificação dos esquemas.

Para a validação da ferramenta e de seus parsers, foi utilizado o banco de dados público MIMIC-IV (JOHNSON et al., 2021), que armazena informações de pacientes internados em um hospital de Massachusetts. A fim de simular um ambiente de persistência poliglota (*polystore*), o esquema original foi desmembrado e adaptado, com diferentes partes do modelo de dados sendo reestruturadas para os paradigmas de Grafo, Documento, Colunar e Chave-Valor, além de manter parte da estrutura Relacional. Este conjunto de esquemas heterogêneos serviu como a base de entrada para testar a capacidade de mapeamento e unificação da ferramenta em um cenário realista.

## 4 Resultados e Discussão

Como resultado deste trabalho, foi implementada a ferramenta **Polyschema**, que, a partir de um conjunto de esquemas heterogêneos de entrada, gera um esquema unificado que representa a estrutura consolidada dos dados, seguindo a gramática apresentada. A ferramenta opera em um fluxo de duas fases principais: o mapeamento individual de cada esquema de origem para uma representação intermediária e, em seguida, a unificação desses resultados em um único artefato de saída.

A gramática que serve como base é um metamodelo unificado, projetado para ser capaz de representar as particularidades de diferentes paradigmas de banco de dados. Sua estrutura central é a definição de ENTITY, que agrupa propriedades dentro de um bloco tipado que identifica a origem (ex: GRAPH, DOCUMENT). O metamodelo também formaliza RELATIONs e CONSTRAINTs para descrever relacionamentos e chaves. A especificação formal deste metamodelo é apresentada na Figura 1.

```
<Schema>     ::= “SCHEMA” <SchemaName> "{" <Definition>+ "}"
<Definition> ::= <Entity> | <Relationship> | <ConstraintDef>
<Entity>     ::= “ENTITY” <EntityName> ("EXTENDS" <EntityName>)? "{"
                 (<EntityType> "{" (<Property>)* "}") "}"
<EntityType> ::= "DOCUMENT" | "GRAPH" | "COLUMNS" | "KEY_VALUE" | "RELATIONAL"
<Property>   ::= <PropertyName> ":" <Type>  ("{" <Constraint> ("," <Constraint>)* "}")?
<Type>       ::= <At_Type>
                 | "ARRAY" ("[" <Type> “,” "]")* ("(" <Min ":" Max> ")")?
                 | "TUPLE" "{" <Type> ("," <Type>)+ "}"
                 | “OBJECT” (<ObjectName>)? "{" <Property>+ "}"
                 | "MAP" "(" <At_Type> ":" <Type> ")"
                 | "ENUM" "[" (<At_Type> “, ”)+ "]"
                 | "TAGGED_UNION" "(" <PropertyName> ":" <EnumRef> ")" "{" <Case>+ "}"
<At_Type> ::=  "STRING" | "BOOLEAN" | "NULL" | “DATE” |
               "NUMBER" ("(" ("min" "=" <Number>)? "," ("max" "=" <Number>)? ")")?

<Case>          ::= <At_Type> "=>" <Type>
<Relationship>  ::= "RELATION" <RelationshipName>
                    "FROM" <Entity> "TO" <Entity>
                    "(" <Cardinality> ")" “;” "(" <Cardinality> ")" ("{" <Property>+ "}")?
<ConstraintDef> ::= "CONSTRAINT" <NameEntity> “Key” "ON" <Entity> "." (<PropertyName> |
                    "(" <PropertyName> ("," <PropertyName>)+ ")") “IS” ("KEY")
<Constraint>    ::= "REQUIRED" | "UNIQUE" | "KEY" |” OPTIONAL”
<Cardinality>   ::= <Min> ":" <Max>
<Min>, <Max>    ::= "0" | "1" | "N"
<SchemaName>, <EntityName>, <RelationshipName>, <PropertyName>, <ObjectName> := (STRING)*
```

**Figura 1**. Gramática do metamodelo unificado.

O principal resultado deste trabalho é a capacidade da ferramenta Polyschema de traduzir esquemas de formatos distintos para o metamodelo unificado. O exemplo abaixo exibe um trecho de um script SQL DDL que serviu como uma das entradas para a ferramenta.

```
CREATE TABLE admissions (
    hadm_id INT NOT NULL PRIMARY KEY,
    subject_id INT NOT NULL,
    CONSTRAINT fk_admissions_patients
    FOREIGN KEY(subject_id) REFERENCES patients(subject_id)
);
```

O trecho de código a seguir demonstra como a ferramenta processou a entrada anterior e a integrou no esquema final. A comparação entre os dois exemplos evidencia a transformação realizada: a tabela admissions foi convertida para uma RELATIONAL ENTITY admissions; a coluna hadm_id INT NOT NULL PRIMARY KEY foi mapeada para hadm_id: NUMBER { REQUIRED, KEY }; e a FOREIGN KEY foi transformada em uma RELATION explícita. Esse processo é aplicado a todas as fontes, permitindo a consolidação de múltiplas visões em um único artefato.

```
SCHEMA UnifiedPolySchema {
    ENTITY admissions {
        RELATIONAL {
            hadm_id: NUMBER { REQUIRED }
            subject_id: NUMBER { REQUIRED }
        }
    CONSTRAINT admissionsKey ON admissions.hadm_id IS KEY
    RELATION fk_admissions_patients FROM admissions (subject_id) TO
    patients (subject_id)
    }
```

A principal contribuição desta abordagem é a capacidade de criar uma visão coesa e centralizada de um ambiente de persistência poliglota. Ao preservar a tag de origem de cada entidade, o esquema unificado não apenas descreve os atributos, mas também mantém o contexto de seu paradigma original. Essa representação consolidada serve como uma documentação clara do modelo de dados, facilitando a análise de redundâncias e a identificação de pontos de integração entre sistemas distintos.

## 5 Conclusão

O presente trabalho atingiu o objetivo de desenvolver a ferramenta Polyschema, que unifica esquemas de diferentes paradigmas de banco de dados — Relacional, Grafo, Documento, Chave-Valor e Colunar. A ferramenta supera a limitação da análise isolada de cada fonte de dados, provendo uma visão consolidada que facilita a compreensão e a integração em ambientes de persistência poliglota.

## Referências Bibliográficas

JOHNSON, A. et al. **MIMIC-IV v1.0**. 2021. PhysioNet. Disponível em: https://physionet.org/content/mimiciv/1.0/. Acesso em: 1 fev. 2025.

SCHNEIDER, A. M. et al. **Schema Extraction on Key-Value Databases**. Chapecó: Universidade Federal da Fronteira Sul, 2025.

BANHARA, N.; SCHREINER, G. A.; DUARTE, D. **Extração de esquemas de fontes heterogêneas semiestruturadas: uma abordagem probabilística**. Chapecó: Universidade Federal da Fronteira Sul, 2022. (Projeto de Pesquisa - Código: PES-2022-0312).

CRUZ, H. T.; SCHREINER, G. A.; DUARTE, D. **Extração de Metadados de Banco de Dados Semiestruturados**. Chapecó: Universidade Federal da Fronteira Sul, 2023. (Projeto de Pesquisa - Código: PES-2023-0182).

**Palavras-chave:** Metamodelo; Persistência Poliglota; Esquema.

**Nº de Registro no sistema Prisma: PES-2024-0260**

**Financiamento:** Universidade Federal da Fronteira Sul

UNIVERSIDADE FEDERAL DA FRONTEIRA SUL