

# Bancos de Dados em Grafos

Gustavo Estrela de Matos  
*Instituto de Matemática e Estatísticas*  
*Universidade de São Paulo*  
São Paulo, Brasil  
gestrela@ime.usp.br

Hector Montenegro Terceros  
*Instituto de Matemática e Estatísticas*  
*Universidade de São Paulo*  
São Paulo, Brasil  
hector@ime.usp.br

**Resumo**—Vamos fazer este por último...

**Index Terms**—component, formatting, style, styling, insert **Isso aqui eu não entendi direito, não sei como fazer aqui.**

## I. INTRODUÇÃO

Grafos são objetos matemáticos que podem ser usados como estruturas de dados em diversas aplicações computacionais. Sistemas de Bancos de Dados baseados em grafos são considerados bancos de dados NoSQL [1] e podem ser aplicados em diversos contextos, como o de aplicativos da web; áreas industriais de transportes, telecomunicação e comércio [2]; e também em áreas de pesquisa, como em bioquímica [4], biologia molecular [5] e web semântica [6].

Um grafo pode ser definido como uma dupla  $G = (V, E)$ , em que  $V$  é um conjunto de vértices (ou nós) e  $E$  é um conjunto de arcos, que conectam dois nós. Podemos representar um arco  $e \in E$ , por uma dupla  $e = (v_i, v_j)$  com  $v_i \in V$  e  $v_j \in V$ . No contexto de bancos de dados de grafos, usualmente um nó representa uma entidade modelada, e arcos representam relacionamentos entre essas entidades. É comum que os nós recebam rótulos que estão associados ao tipo de entidade modelada, e também um conjunto de propriedades em formato chave-valor, capaz de armazenar atributos da entidade. Além disso, os arcos do modelo também podem receber rótulos que identificam o tipo de relacionamento que é modelado, e um conjunto de propriedades que representam atributos do relacionamento.

Bancos de dados baseados em grafos costumam ser aplicados em contextos em que os dados de interesse possuem relacionamentos complexos ou simplesmente quando boa parte da informação está contida nos relacionamentos. Nestes casos, um banco de dados relacional pode ser inadequado ou ineficiente. Considere, por exemplo, a relação `FABRICA (id_fabricante, id_produto)`. Em uma consulta em que se deseja saber as informações dos produtos fabricados por uma fábrica, no modelo relacional, é necessário fazer uma junção com a relação que armazena os dados dos produtos, enquanto no modelo baseado em grafos, basta percorrer os arcos do relacionamento `FABRICA` que estão ligados ao nó que representa a fábrica de interesse. Além de ser mais eficiente para algumas consultas, sistemas baseados em grafos podem ter consultas mais expressivas, capazes de representar relacionamentos complexos entre os dados [3].

Assim como outros bancos de dados NoSQL, os bancos de dados em grafos são muito utilizados em aplicações que precisam armazenar um grande volume de dados. Para atender a este requisito, alguns bancos de dados baseados em grafos permitem armazenamento distribuído. Este tipo de solução precisa implementar particionamento de dados, além disso, um sistema distribuído deve providenciar maneiras de responder a consultas que acessam informações de vértices alocados em diferentes máquinas.

Neste artigo, nosso principal objetivo é apresentar os conceitos fundamentais e as principais soluções para implementar bancos de dados baseados em grafos distribuídos, e também sistemas para o processamento distribuído de grafos. Ao longo deste trabalho, vamos... **informações a serem adicionadas no futuro, com o que vamos colocar de fato no artigo.**

## II. METODOLOGIA

- Como escolhemos os artigos que lemos para criar esse artigo? - Qual tipo de grafos estamos focados em tratar? (Os que a gente achou... acho que é maioria usado pra rede social??)

## III. CONCEITOS FUNDAMENTAIS

- A. *Graph DBMS versus Graph Analytics Systems*
- B. *Notação e representação de grafos (?)*
- C. *Cortes de grafos*

## IV. PARTICIONAMENTO DE GRAFOS

## V. SISTEMAS GERENCIADORES DE BANCOS DE DADOS EM GRAFOS

## VI. GRAPH ANALYTICS SYSTEMS

### A. *Pregel*

## REFERÊNCIAS

- [1] "NoSQL Databases", <http://nosql-database.org/>.
- [2] "Neo4j Customers", <https://neo4j.com/customers/>.
- [3] M. Hunger, R. Boyd. "RDBMS & Graphs: SQL vs. Cypher Query Languages" Neo4j Blog. Mar. 2016. <https://neo4j.com/blog/sql-vs-cypher-query-languages/>
- [4] N. Swainston et al., "biochem4j: Integrated and extensible biochemical knowledge through graph databases," PLOS ONE, vol. 12, no. 7, p. e0179130, Jul. 2017.
- [5] F. Olken, "Graph Data Management for Molecular Biology," OMICS: A Journal of Integrative Biology, vol. 7, no. 1, pp. 75–78, Jan. 2003.
- [6] B. McBride, "Jena: a semantic Web toolkit," IEEE Internet Computing, vol. 6, no. 6, pp. 55–59, Nov. 2002.