THE UNIVERSITY OF MELBOURNE

Faculty of Engineering and Information Technology

Group 29

Graph Data Management Survey

COMP90050

Advanced Database System

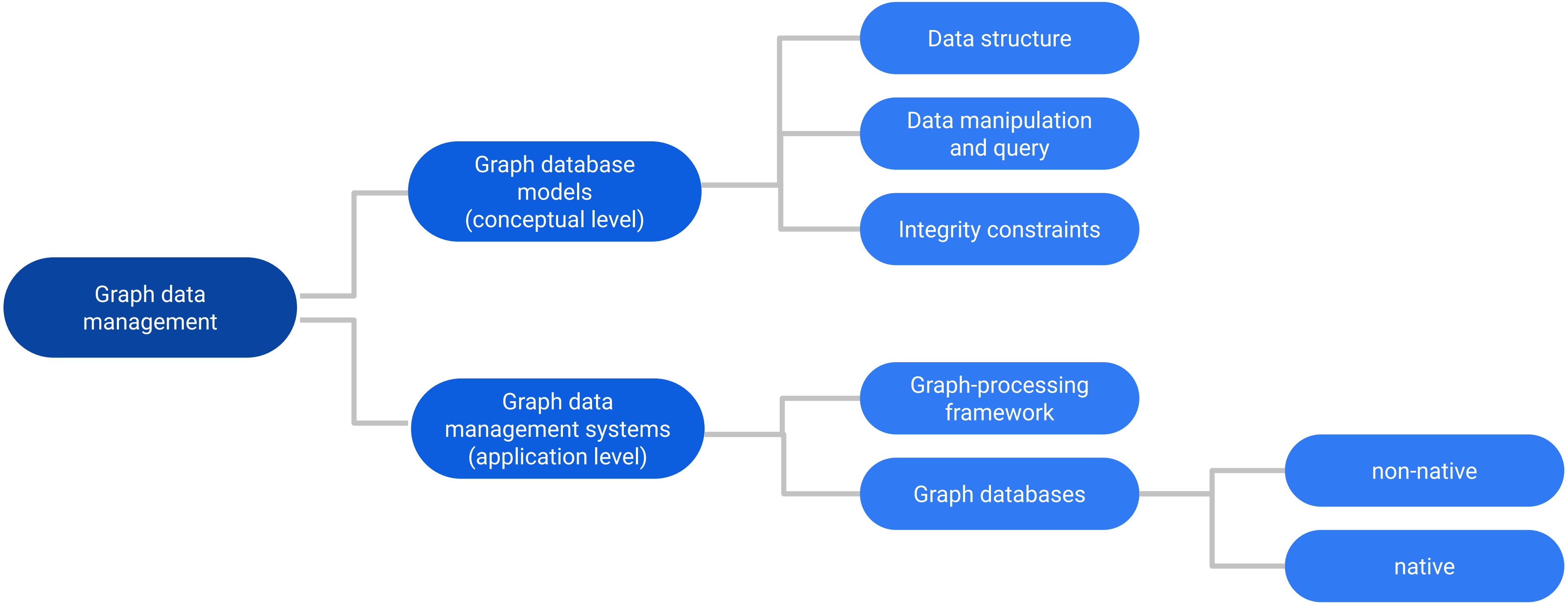
Presented by: Yu Yan, Yuexin Li, Jieyun Peng, Shengqi Ma

Outline

- Introduction of Graph Data
- Graph Queries
- Graph Indexing
- Graph Database System
- Application
- Conclusion
- Reference

Introduction of Graph Data management

THE UNIVERSITY OF MELBOURNE


Graph data management
Graph database models (conceptual level)
Data structure
Data manipulation and query
Integrity constraints
Graph data management systems (application level)
Graph-processing framework
Graph databases
non-native
native

Graph Queries

1. **Adjacency Queries**
    - Node/edge Queries
        - Adjacent node/edge
    - Neighbourhood Queries
        - K-Neighbourhood

    ? Challenge: Sparse Data

<u>*Application: Recommendation Systems, Information Retrieval, Six-Degrees-Of-Separation Theory…</u>

2. **Pattern Matching Queries**
    - Subgraph Isomorphism
        - NP(nondeterministic polynomial time complete)-complete
    - Graph Isomorphism Problem
        - Unknown Computational Complexity

<u>*Application: Bioinformatics, Pattern Recognition…</u>

Graph Queries

3. Reachability Queries (Connectivity)

- Reachability of information
- Characterized by path and traversal problems
- Expressed by *Regular Path Query (RPQ)*
- Less Informative than *Pattern Matching Queries*

? **Challenge**: Computational Complexity-----NP-Complete Problem

*Application: Shortest Path, Social Networks…

4. Analytical Queries

- Summarization Queries
- Have aggregate functions

*Application: Graph Mining/Analysis, graph-processing frameworks

Graph Indexing

General Indexing : path-based => graph-based

- ○ improving query efficiency by means of data mining
- ○ issue of increased index size should be tackled

- **GraphGrep (path based)**
  - constructed through enumeration of paths
- **G-Index (graph-based)**
  - constructed through enumeration of fragments
- **FG-Index (graph-based)**
  - constructed through BuildIndex algorithm
  - an optimization in the aspect of candidate verification

Graph Indexing

Multi-dimensional Data : MBR => Graph Closure

- improving pruning rates
- can support more types of graph queries

Semi-structured Data : bisimilarity => k-bisimilarity

- reducing index size and simplifying the searching
- require validation when query path is longer than k

- **R-Tree (MBR)**
  - using spatial objects to perform similarity queries
- **Closure-Tree (Graph Closure)**
  - using pseudo subgraph isomorphism for pruning
  - can support both similarity and range queries

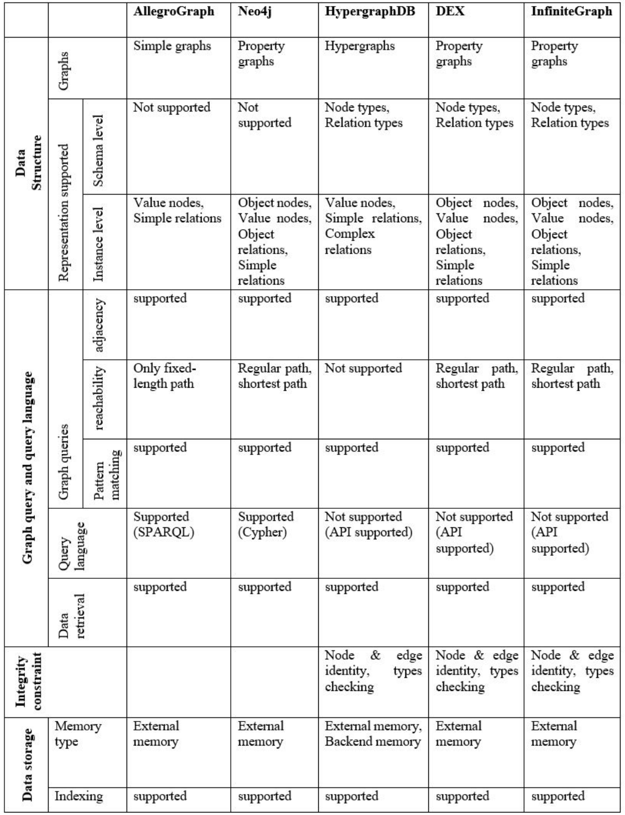
- **DataGuide and 1-index (bisimilarity)**
  - precisely encode all paths in the data graph
- **A(k)-index (k-bisimilarity)**
  - relaxes the equivalence condition via k-bisimilarity
- **D(k)-index (k-bisimilarity)**
  - adaptive to query load and source data

Graph data management system

- **Graph-processing frameworks**
  - meet the needs of large-scale graph data
  - use parallel-processing strategies
- **Graph databases**
  - persistent management of graph data
  - efficient in single node solutions
  - <u>non-native graph databases</u>
  - <u>native graph databases</u>
    - AllegroGraph
    - Neo4j
    - HypergraphDB
    - DEX
    - InfiniteGraph

Comparison of graph databases

- **Data structure**
  - Data graphs
  - Representation capability
- **Graph query and language**
  - Graph queries
    - Pattern matching
    - Reachability
    - Adjacency
  - Query language
- **Integrity constraint**
- **Data storage**
  - Memory type
  - Indexing



			AllegroGraph	Neo4j	HypergraphDB	DEX	InfiniteGraph
Data Structure	Graphs		Simple graphs	Property graphs	Hypergraphs	Property graphs	Property graphs
	Representation supported	Schema level	Not supported	Not supported	Node types, Relation types	Node types, Relation types	Node types, Relation types
		Instance level	Value nodes, Simple relations	Object nodes, Value nodes, Object relations, Simple relations	Value nodes, Simple relations, Complex relations	Object nodes, Value nodes, Object relations, Simple relations	Object nodes, Value nodes, Object relations, Simple relations
Graph query and query language	Graph queries	adjacency	supported	supported	supported	supported	supported
		reachability	Only fixed-length path	Regular path, shortest path	Not supported	Regular path, shortest path	Regular path, shortest path
		Pattern matching	supported	supported	supported	supported	supported
	Query language		Supported (SPARQL)	Supported (Cypher)	Not supported (API supported)	Not supported (API supported)	Not supported (API supported)
	Data retrieval		supported	supported	supported	supported	supported
Integrity constraint					Node & edge identity, types checking	Node & edge identity, types checking	Node & edge identity, types checking
Data storage	Memory type		External memory	External memory	External memory, Backend memory	External memory	External memory
	Indexing		supported	supported	supported	supported	supported

Applications of Graph Data Management

- **Biology**
  - Biological network data
  - Systems Biology Graph Extender

- **Reinforcement Learning**
  - Decision Making
    - Automation
    - Adaptivity
  - April
    - Storage structure selection
    - Index selection
    - Query optimization

- **Cloud Environments**
  - X10
  - ScaleGraph
  - Acasia

Applications of Graph Data Management & Comparisons

- **Social Networks**
  - Big Social Graph
  - Knowledge-based system

- **Healthcare Systems**
  - Healthcare graphs
    - “3NF Equivalent Graph” (3EG) transformation
  - An ensemble system
    - Decrease database complexity
    - Increase data accessibility

- **Comparisons between Relational Data Management and Graph Data Management**
  - Storage
  - Format of data
  - Complexity of queries and databases

Conclusion

- Graph data management is increasingly popular in both industries and academia. However, it faces many challenges in solving data problems.
- Design of graph data management system has uncertainties and difficulties due to a large variety of graphs in different forms and shapes.
- Performance of the system worsens with large size and high dimension graphs and complex relationships.
- Query overhead is hard to predict since it’s closely dependent on the solution of mathematical issues.
- Trade-off between storage and performance enhancement of indexing should be considered.

Reference

A. Cuzzocrea and I.-Y. Song, “Big Graph Analytics,” Proceedings of the 17th International Workshop on Data Warehousing and OLAP - DOLAP ’14, 2014, doi: 10.1145/2666158.2668454.

Angles, R., Gutierrez, C. (2018). An Introduction to Graph Data Management. In: Fletcher, G., Hidders, J., Larriba-Pey, J. (eds) Graph Data Management. Data-Centric Systems and Applications. Springer, Cham. https://doi.org/10.1007/978-3-319-96193-4_1

B. A. Eckman and P. G. Brown, “Graph data management for molecular and cell biology,” IBM Journal of Research and Development, vol. 50, no. 6, pp. 545–560, Nov. 2006, doi: 10.1147/rd.506.0545.

C.-W. Chung, J.-K. Min, and K. Shim, "APEX: An adaptive path index for XML data," in Proceedings of the 2002 ACM SIGMOD international conference on Management of data, 2002, pp. 121-132.

H. He and A. K. Singh, "Closure-tree: An index structure for graph queries," in 22nd International Conference on Data Engineering (ICDE'06), 2006, pp. 38-38: IEEE

H. Wang et al., “April,” Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Oct. 2020, doi: 10.1145/3340531.3417422.

J. Cheng, Y. Ke, W. Ng, and A. Lu, "Fg-index: towards verification-free query processing on graph databases," presented at the Proceedings of the 2007 ACM SIGMOD international conference on Management of data, Beijing, China, 2007. Available: https://doi.org/10.1145/1247480.1247574

M. Dayarathna and T. Suzumura, “High-Performance Graph Data Management and Mining in Cloud Environments with X10,” Computer Communications and Networks, pp. 173–210, 2017, doi: 10.1007/978-3-319-54645-2_7.

M. Yannakakis, “Graph-Theoretic Methods in Database Theory,” in Proceedings of the 9th Symposium on Principles of Database Systems (PODS). ACM Press, 1990, pp. 230–242.

P. Braun, A. Cuzzocrea, C. K. Leung, A. G. M. Pazdor, and K. Tran, “Knowledge Discovery from Social Graph Data,” Procedia Computer Science, vol. 96, pp. 682–691, 2016, doi: 10.1016/j.procs.2016.08.250.

P. Ciaccia, M. Patella, and P. Zezula, "M-tree: An efficient access method for similarity search in metric spaces," in Vldb, 1997, vol. 97, pp. 426-435.

Q. Chen, A. Lim, and K. W. Ong, "D (k)-index: An adaptive structural summary for graph-structured data," in Proceedings of the 2003 ACM SIGMOD international conference on Management of data, 2003, pp. 134-144.

R. Angles, "A Comparison of Current Graph Database Models," 2012 IEEE 28th International Conference on Data Engineering Workshops, 2012, pp. 171-177, doi: 10.1109/ICDEW.2012.31.

R. Kaushik, P. Shenoy, P. Bohannon, and E. Gudes, "Exploiting local similarity for indexing paths in graph-structured data," in Proceedings 18th International Conference on Data Engineering, 2002, pp. 129-140: IEEE.

T. Milo and D. Suciu, "Index structures for path expressions," in International Conference on Database Theory, 1999, pp. 277-295: Springer.

Y. Park, M. Shankar, B.-H. Park, and J. Ghosh, “Graph databases for large-scale healthcare systems: A framework for efficient data management and data services,” 2014 IEEE 30th International Conference on Data Engineering Workshops, Mar. 2014, doi: 10.1109/icdew.2014.6818295.

R. Giugno and D. Shasha, "Graphgrep: A fast and universal method for querying graphs," in 2002 International Conference on Pattern Recognition, 2002, vol. 2, pp. 112-115: IEEE.

X. Wang and W. Chen.(2020) Knowledge Graph Data Management: Models, Methods, and Systems. In: U, L., Yang, J., Cai, Y., Karlapalem, K., Liu, A., Huang, X. (eds) Web Information Systems Engineering. WISE 2020. Communications in Computer and Information Science, vol 1155. Springer, Singapore. https://doi.org/10.1007/978-981-15-3281-8_1

X. Yan, P. S. Yu, and J. Han, "Graph indexing: a frequent structure-based approach," presented at the Proceedings of the 2004 ACM SIGMOD international conference on Management of data, Paris, France, 2004. Available: https://doi.org/10.1145/1007568.1007607

THE UNIVERSITY OF MELBOURNE

Faculty of Engineering and Information Technology

Thank you!