

NoSQL – Graph databases

Michal Barla

NoSQL systems

- MapReduce framework
- Key-value storage
- Column-oriented storage
- Document storage
- Graph databases

NoSQL systems

- MapReduce framework
- Key-value storage
- Column-oriented storage
 - Cassandra, Scylla, ...
 - Fast, variable column spaces, many many columns
- Document storage
- Graph databases

NoSQL systems

- MapReduce framework
- Key-value storage
- Column-oriented storage
- Document storage
- **Graph databases**
 - Completely different, but also NoSQL
 - Forget about aggregates, embrace graphs

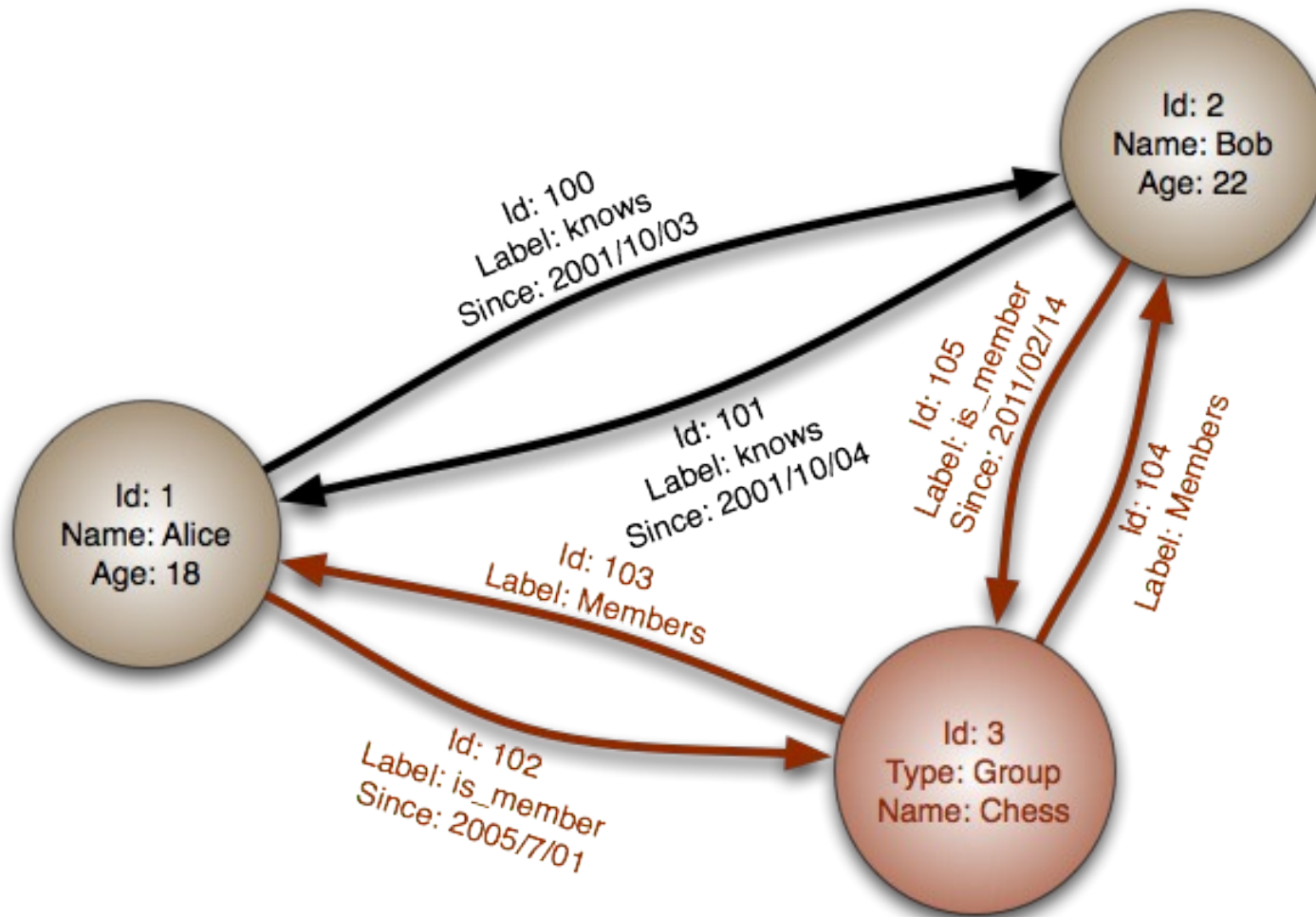
Graph databases

- Data model: nodes & edges
- Nodes have properties (attributes)
 - e.g, ID
- Edges have properties
 - label, type, role... anything

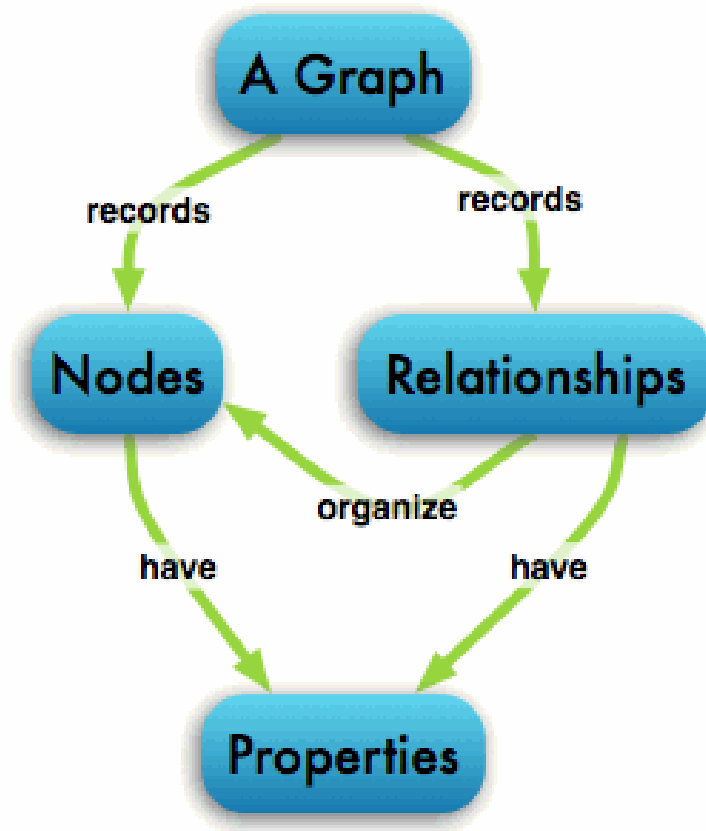
Graph databases – when?

- If we have data with many different types of relationships
- If we cannot say which relationships are relevant,
 - Maybe we want to mix them in a query
- Link-rich domains (social networks et al.)
- Routing, dispatching, ...
- Recommendations

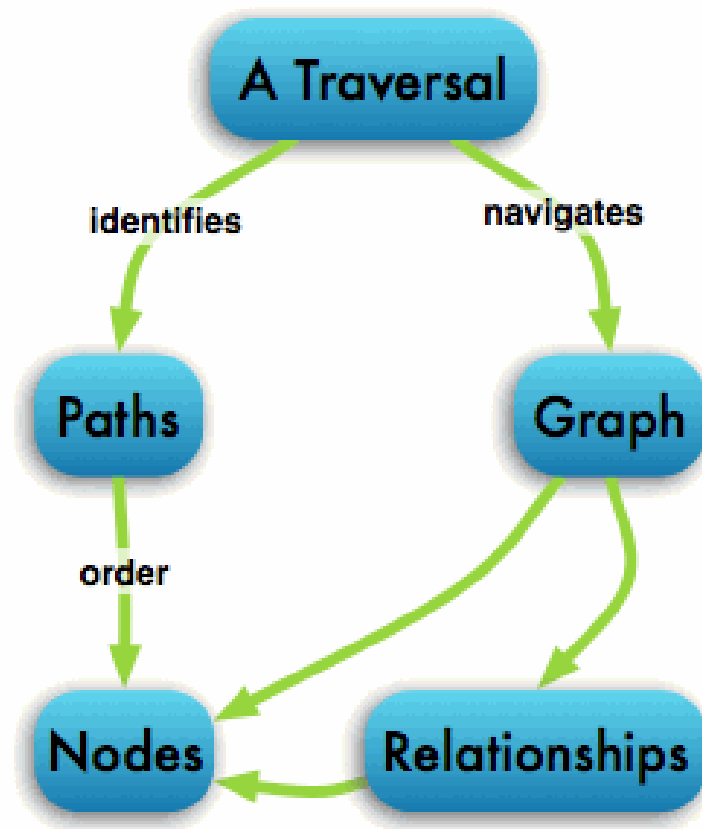
Sample from Wikipedia



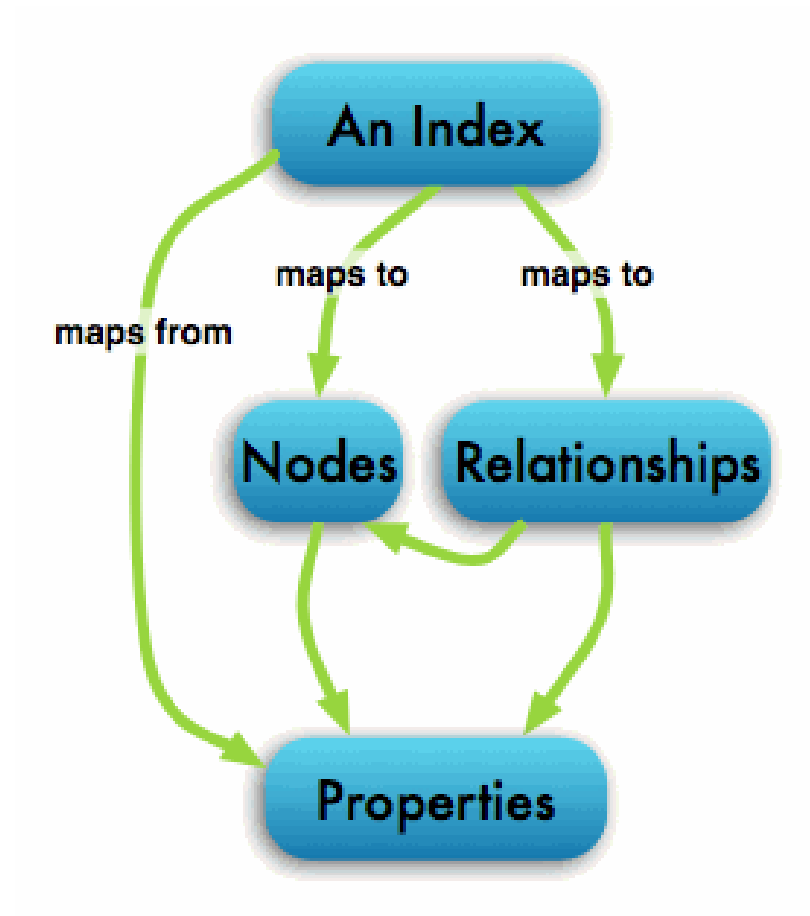
Neo4J – how to explain a graph through a graph



Graph traversal in a graph (querying)



Graph indexing



Neo4j demo

<https://neo4j.com/download/>

- Run `<install-dir>/bin/neo4j console`
- And open your browser on `localhost:7474`

Summary

- Graph databases might be convenient if we have data with many links and questions which can benefit from such links
 - Typed connections
 - Recursive graph traversal
 - Graph algorithms