Introduction to Graph Databases

by

Angela Ebirim

Back-end developer, PDS

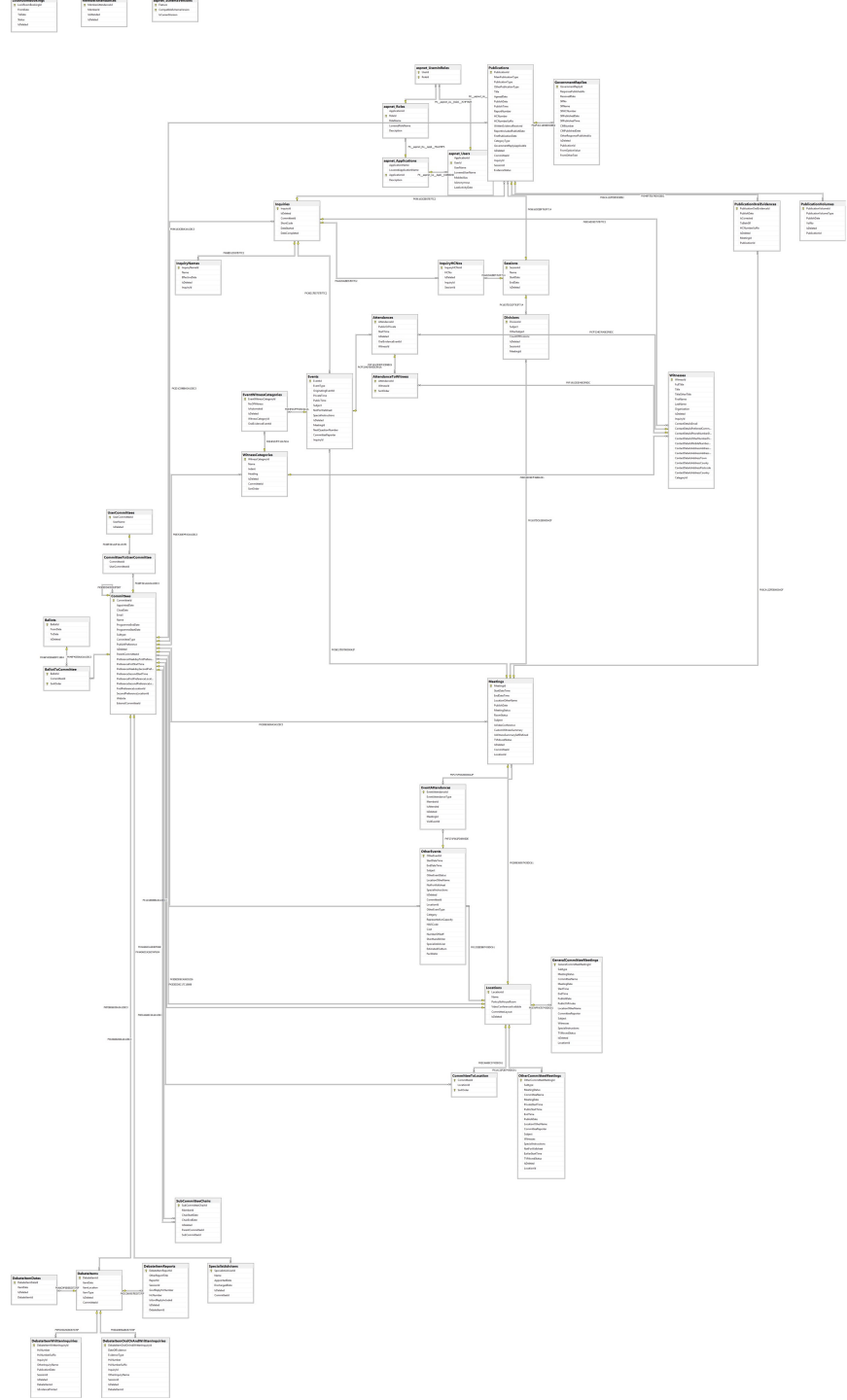

dreamstime.com

What is a database ?

- A database is simply a container for disparate types of information
- A bookcase shelf can be called a database because it has the following characteristics:
  - It a system that makes it easy to search, select and store information
  - Each shelf is made up of books(tables) which hold different types of data which can be represented in a tabular form (rows and columns)
  - Each book is made up of pages which equate to database fields
  - You can use the information stored in the back of the book (index) to perform a search for a particular topic of interest and return a record set or the data you want (probably be in the form of a page number reference)

Relational databases

- These are databases built as software applications and are the workhorses for repositories of information. They have been in use for the last 30 years and are still heavily used today
- A highly structured data model is created initially usually in a tabular form and then data is applied to fit that model
- Every relational table will have a unique field (primary key) e.g **OrderId** when first created and a foreign key (a field in the table that is a primary key in another table)
- Foreign keys are important when retrieving a record set that involves cross referencing of multiple tables.
- The more tables you have to traverse to get a record set, the more JOINS you have to perform, the more intensive in terms of memory and compute power and the longer it takes to execute a query

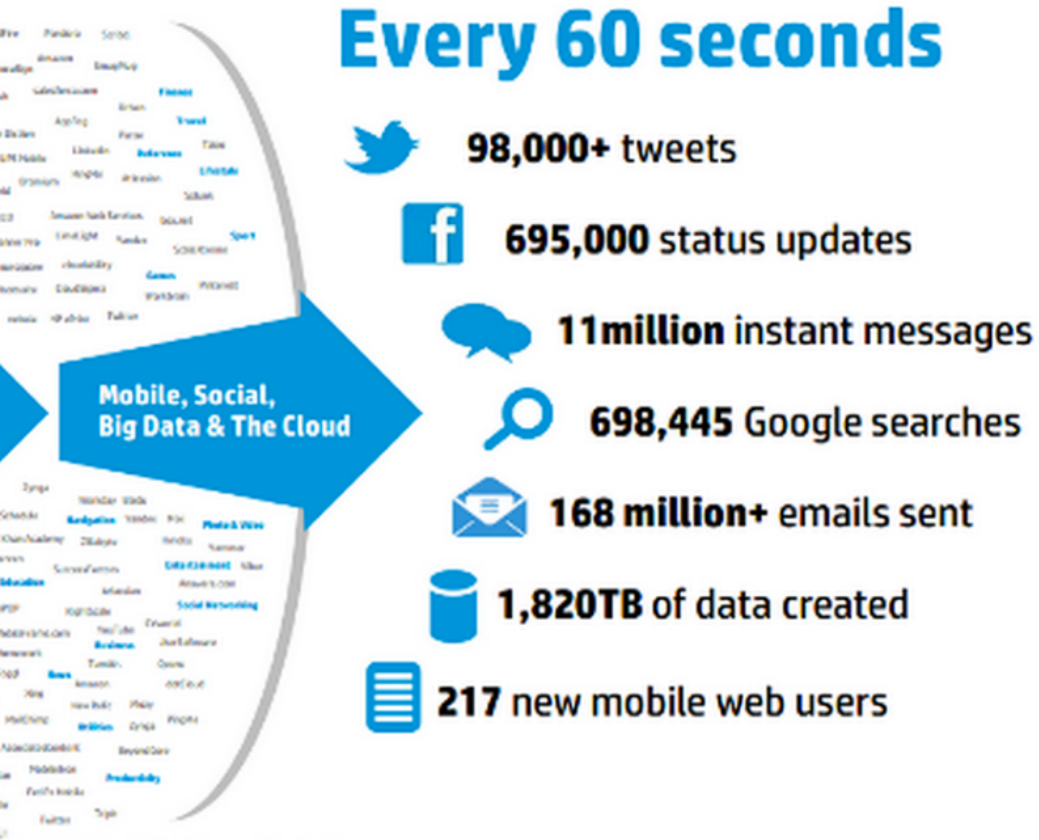


Trends in Data

- According to Eric Schmidt, Chairman, Alphabet (parent company of Google) ”Every two days we create more information than we did up until 2003 approximately **2 Exabytes** (2 billion GB) or 250 million 8GB Samsung smartphones”
- Global internet traffic is currently about **110 Exabytes** a month
- 90% of the world’s data has been created in the last 2 years alone
- Forecasts suggest that annual data creation will hit approximately **45 Zetabytes** (1 Zetabyte = 1000 Exabytes) by 2020

DEFINING BIG DATA

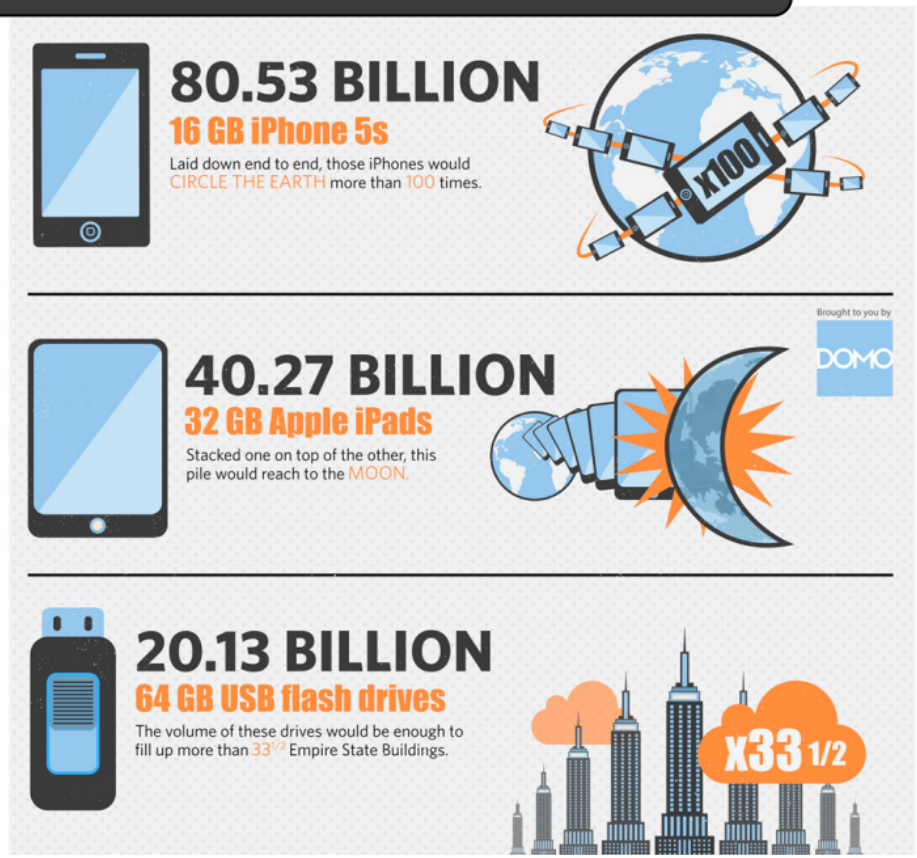
HOW BIG IS BIG?



Every 60 seconds

- **98,000+** tweets
- **695,000** status updates
- **11million** instant messages
- **698,445** Google searches
- **168 million+** emails sent
- **1,820TB** of data created
- **217** new mobile web users

creates enough data to fill

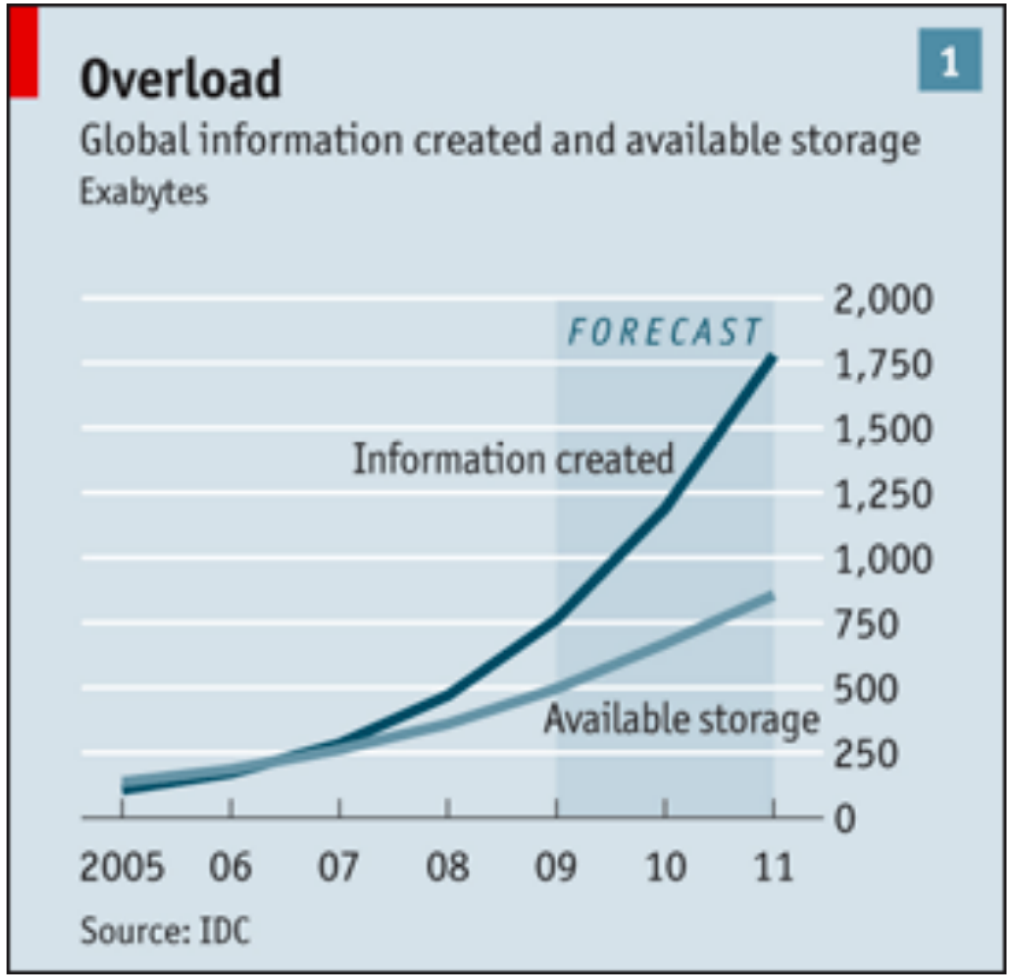


in 1 year!

EVOLUTION OF DATA CREATION

Creation vs Storage

Creation of data completely outstrips current capabilities to store it.

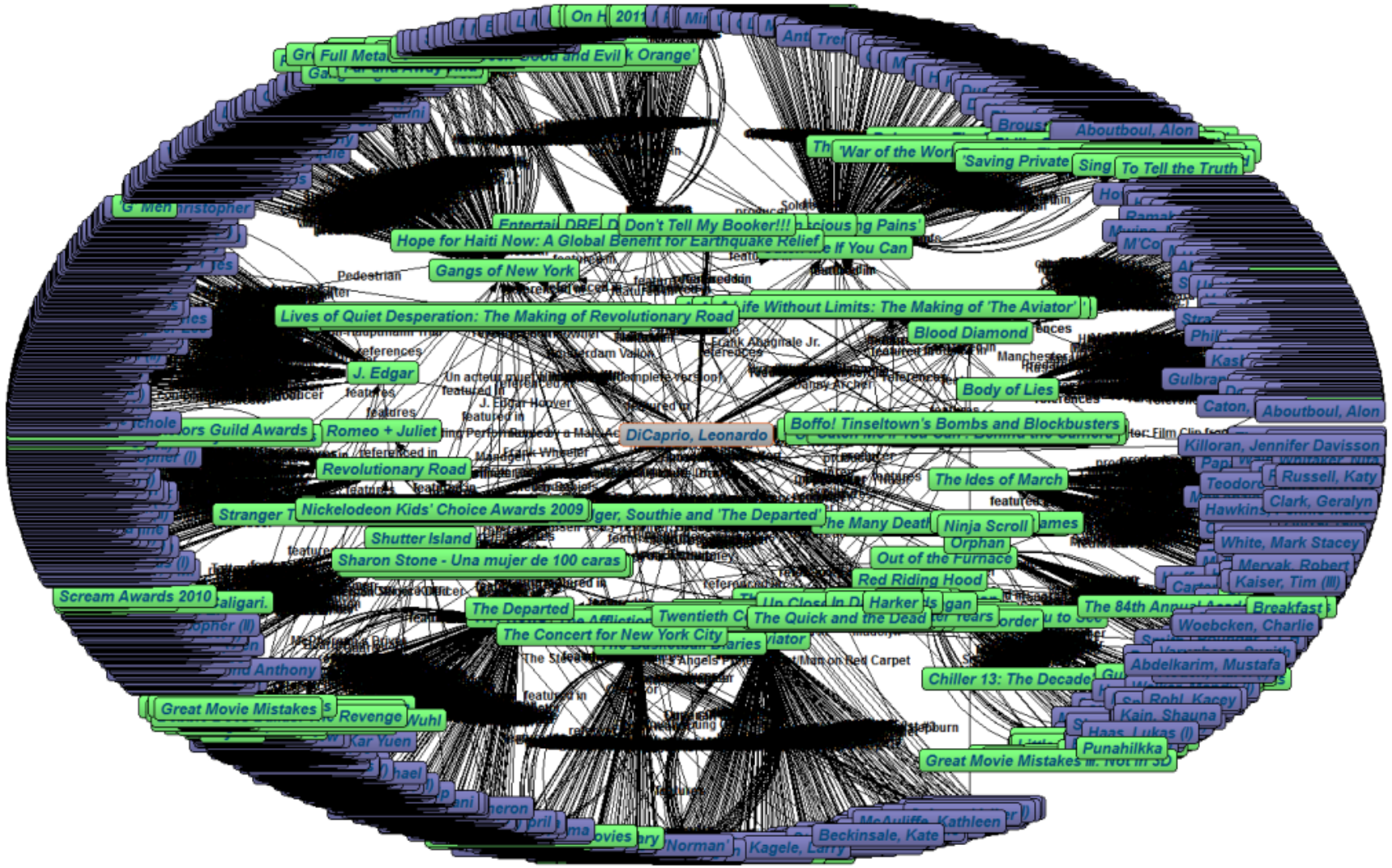


From: http://www.economist.com/node/15557443

Challenges for relational databases in this interconnected world

- Data is more connected than ever e.g. blogs, personalized IoT data (Amazon Alexa Echo), tagging, social networks, instant messages, Google mail
- Developers unable to accommodate connected, unstructured data with ad-hoc relationships inside relational databases
- Relational database schemas usually fixed at it's creation and because of that, unable to adapt to rapidly changing business requirements
- Attempts to adapt to this landscape require frequent schema changes which relational databases aren't designed for
- As data becomes more complex (query example: who are my friends of friends of friends of friends of friends?) results in more joins and a negative impact on database performance

The Connected World of Leonardo DiCaprio



Enter Graph Databases!

What are Graph Databases ?

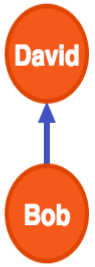
- It is a database that uses graph structures to represent and store data as nodes/vertices and edges (relationships). Edges directly link data together and allow them to be retrieved in one query
- Graphs are mathematical constructs used to model relationships between objects. Smallest representation of a graph is a start node, end node with a relationship between them
- Examples of graphs include pages of a website (nodes) interconnected by links or edges, Tube map, Wikipedia, human genome project, Twitter data
- Highly flexible; emphasis on relationships; reflect the real world; can organically evolve in line with changing business requirements; next evolution of relational databases

Simple Directed Graph

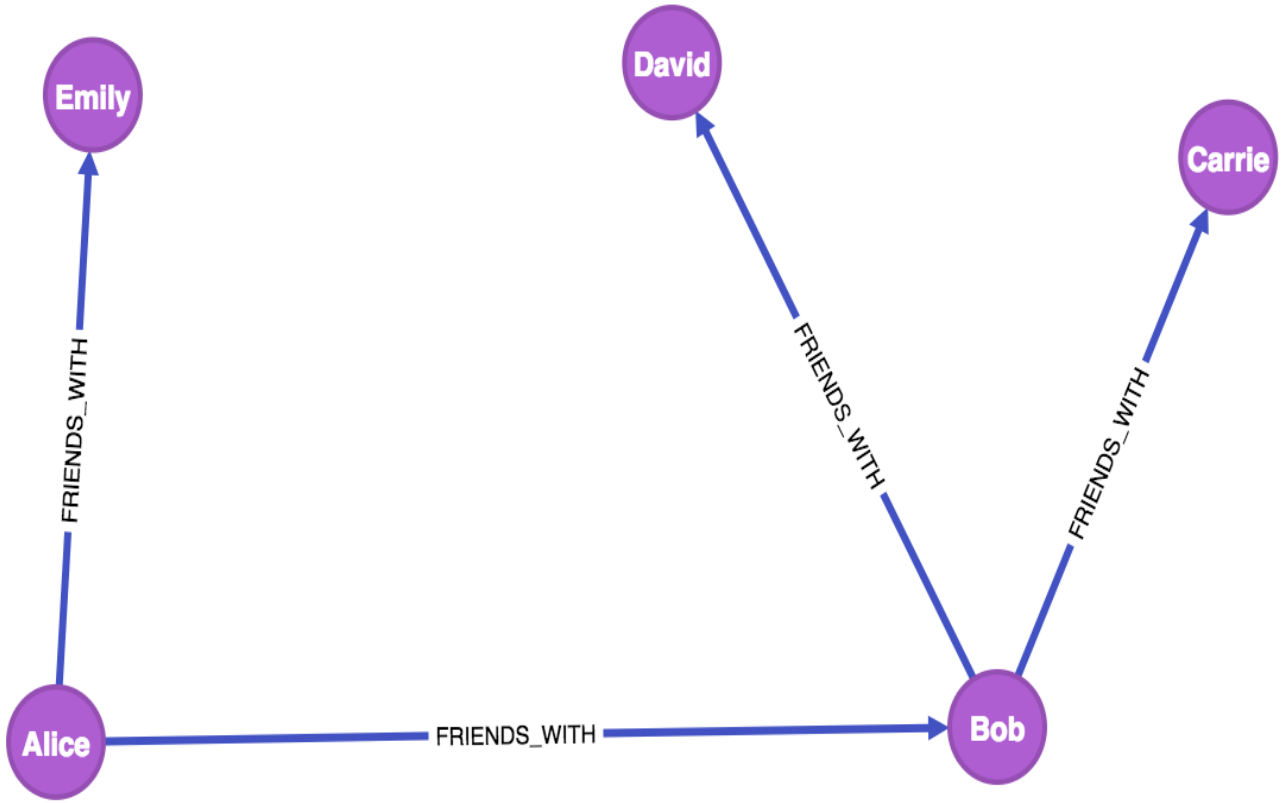
Graph Queries with a Neo4J database

```
CREATE
(a:Person {name: 'Alice'}),
(b:Person {name: 'Bob'}),
(c:Person {name: 'Carrie'}),
(d:Person {name: 'David'}),
(e:Person {name: 'Emily'}),
(a)-[:FRIENDS_WITH]->(b),
(a)-[:FRIENDS_WITH]->(e),
(b)-[:FRIENDS_WITH]->(c),
(b)-[:FRIENDS_WITH]->(d)
```

Examples of Graph Queries



MATCH (b:Person {name: 'Bob'})-[:FRIENDS_WITH]->
(d:Person {name: 'David'}) RETURN b, d



MATCH (a:Person) RETURN a

Introduction to Graph Databases

Any questions ?

Introduction to Graph Databases

Additional material

http://graphdatabases.com/
http://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/big-data-the-next-frontier-for-innovation
https://neo4j.com/
https://neo4j.com/developer/cypher-query-language/
http://www.odbms.org/2015/08/graph-databases-for-beginners-why-a-database-query-language-matters-2/