

Graph Databases: The New Way to Access Super Fast Social Data



Until the NOSQL wave hit a few years ago, the least fun part of a project was dealing with its database. Now there are new technologies to keep the adventuresome developer busy. The catch is, most of these post-relational databases, such as MongoDB, Cassandra, and Riak, are designed to handle simple data. However, the most interesting applications deal with a complex, connected world.

A new type of database significantly changes the standard direction taken by NOSQL. Graph databases, unlike their NOSQL and relational brethren, are designed for lightning-fast access to complex data found in social networks, recommendation engines and networked systems.

[Pancake](#) [1], for example, which is Mozilla's next-generation browser project, uses a graph database to store browsing history in the cloud, since the web is just one big graph.

Graph theory dates back to 1735, when Leonard Euler solved the [Seven Bridges of Königsberg](#) [2] problem by devising a topology consisting of nodes and relationships to answer the then-famous question, "Is it possible to trace a walk through the city

Graph Databases: The New Way to Access Super Fast Social Data

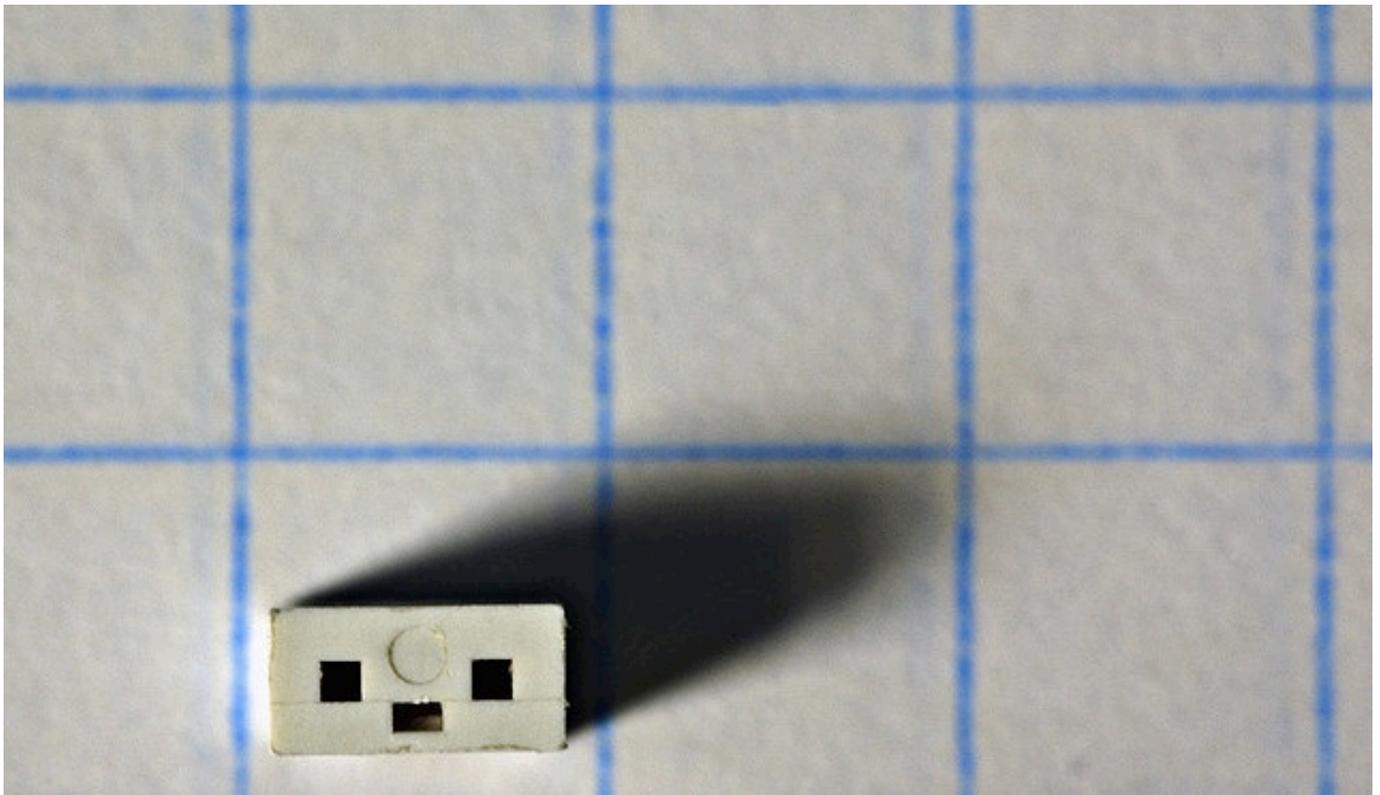
Published on Wireless Design & Development (<http://www.wirelessdesignmag.com>)

that crosses every bridge just once?” Graph theory has since found many uses, but only recently has it been applied to storing and managing data.

It turns out that graphs are a very intuitive way to represent relationships between data.

Think back to your earliest whiteboard graphing session. Traditionally, the developer would hand this off to a DBA, and if she were lucky, would receive a database one month later and start coding. This is because the relational model is tabular, and it takes both time and expertise to represent non-tabular data in a tabular format.

Graph databases let you represent related data as it inherently is: as a set of objects connected by a set of relationships, each with its own set of descriptive properties. With a graph database, the developer can start coding immediately, because the data stored in the database directly parallels the whiteboard representation.



Development agility is handy, but it wouldn't amount to anything without nose-bleeding speed. A recent benchmark took a “friends of friends” query (which finds all of the immediately adjacent nodes and progresses outward one level at a time) and compared performance between a relational database to a graph database. With a query depth of three, the graph database ran over 150 times faster. With a query depth of four, the graph database was over 1,000 times faster.

The reason for this vast difference in performance lies in how data and relationships are stored inside the database. Native graph databases use a technique called “index-free adjacency.” In simple terms, this means that each data element points

Graph Databases: The New Way to Access Super Fast Social Data

Published on Wireless Design & Development (<http://www.wirelessdesignmag.com>)

directly to its inbound and outbound relationships, which in turn, point directly to related nodes, and so on. This technique allows million of related records to be traversed per second.

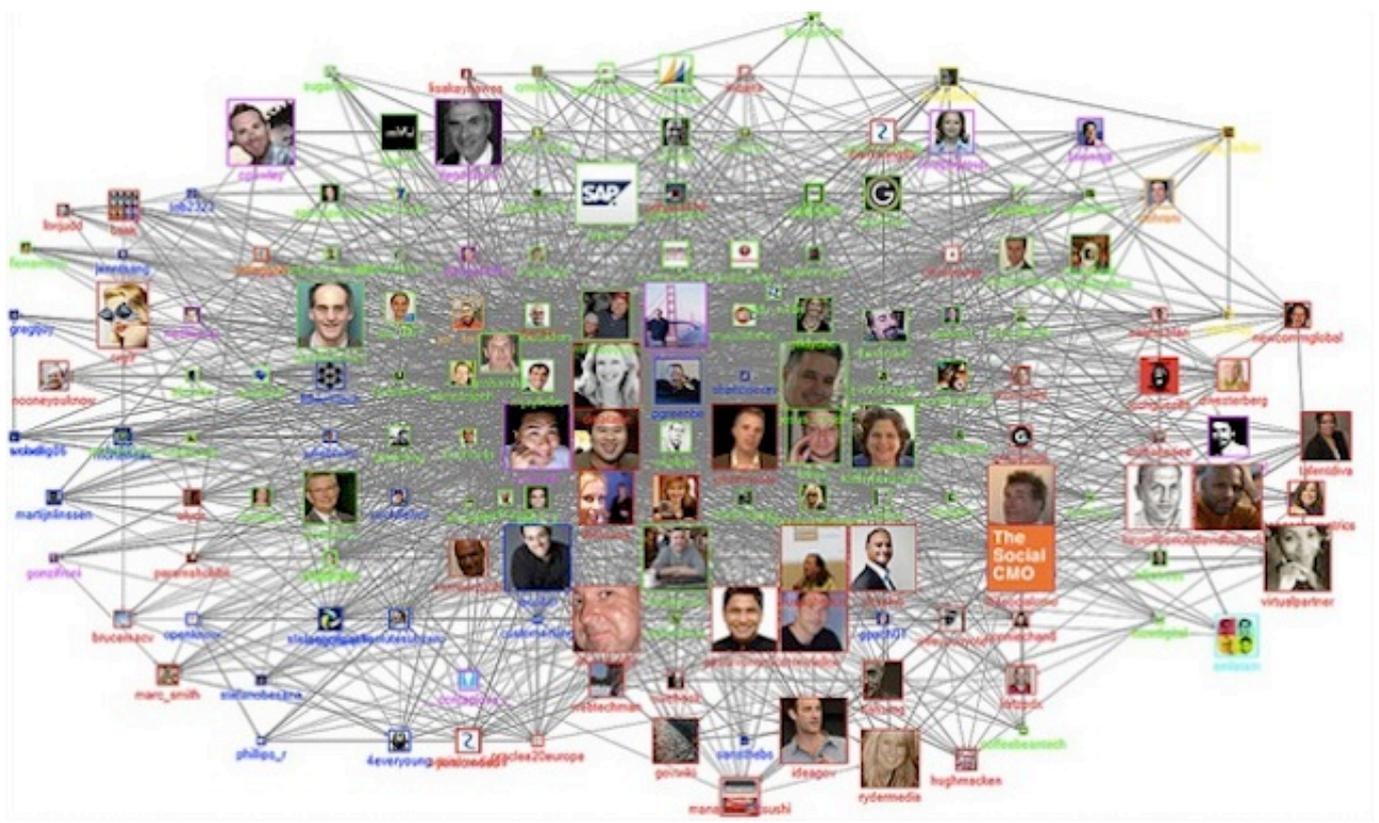
Relational databases, on the other hand, need to carry out a number of steps to determine whether and how things are connected, and then to retrieve related data records. Response times slow down as a relational database grows in volume, which causes problems as a business grows. However with a graph database, traversal speed remains constant, not depending on the total amount of data stored. This allows the database to naturally keep up with one's business as it grows.

To understand a graph database, it helps to envision at how data is represented inside a relational database, like Oracle's or MySQL's. Two invariant concepts in the world of relational databases are: 1) the structure of the data is determined ahead of time, and 2) data structures are tabular.

Graph databases differ in that the data *is* the structure. This provides a level of flexibility and resilience that is a great match for today's fast-moving business and agile development methods.

The data representation also differs fundamentally. Graph databases represent data as "things" (or nodes) and relationships between things. This comes much closer to the way we think about complex systems.

Relational databases are great if you're storing tabular data. But surprisingly — or maybe not so surprisingly — much of the real world is not a table. Things can start getting really complex if you try to turn, for example, a biological system, a social network or the web into a set of tables.



Some of the best-known uses of graph databases include [Google's Knowledge Graph](#) [3], [Facebook's Social Graph](#) [4] and [Twitter's Interest Graph](#) [5].

However, graphs aren't just for the Internet giants anymore. Word is starting to get out. A few of the commercial uses that we are seeing with graph databases include:

- **Social Networking and Recommendations:** We've seen a few social network startups begin with relational and learn very quickly that, as they scaled, they needed to move over to a graph database. Most large/successful social networks use graph databases at their core. Graph database provide exceptional power insofar as they can recommendation algorithms.
- **Network and Cloud Management:** A number of telephone companies are using graph databases to model their networks, in support of network optimization activities and to conduct "what if" failure analysis.
- **Master Data Management:** Cisco recently deployed a new hierarchy management system that handles complex master data, such as organization and product. Because of the flexibility and performance advantages over relational, this system is built on top of a graph database.
- **Geospatial:** The "original" graph use case pioneered by Euler remains alive today. Mobile cell analysis, shortest-path analysis and logistics are three such use cases (among many) where graph databases are currently in use.
- **Bioinformatics:** Era7 Bioinformatics uses graph databases to relate a complex web of information that includes genes, proteins and enzymes.
- **Content Management and Security and Access Control:** Adobe's Creative Cloud uses a graph database to manage access to content and the relationships between users, groups, assets and collections. Telenor, one of the world's largest telcos, brought its login time down from minutes to milliseconds by moving the part of its relational system that handled access control over to a graph database.

While we're certainly not predicting the demise of traditional databases anytime soon, we are seeing an increasing number of applications where graph databases are being used to accelerate development and massively speed up performance. Relational databases are great when it comes to relatively static and predictable tabular data.

The complexities and dynamics of the real world, however call, for new methods. This is particularly true when the world is moving at the speed of web, and everybody is racing to get ahead of everybody else. Intricate and complex processes like human behavior, as well as dynamic interconnected systems, such as those found in nature and on the web, tend to be less static and predictable, and are ideal candidates for graph databases.

[Read More](#) [6]

Graph Databases: The New Way to Access Super Fast Social Data

Published on Wireless Design & Development (<http://www.wirelessdesignmag.com>)

September 26, 2012

Source URL (retrieved on 09/21/2014 - 8:59am):

http://www.wirelessdesignmag.com/news/2012/09/graph-databases-new-way-access-super-fast-social-data?qt-digital_editions=0

Links:

- [1] <https://wiki.mozilla.org/Pancake>
- [2] http://en.wikipedia.org/wiki/Seven_Bridges_of_K%C3%B6nigsberg
- [3] <http://mashable.com/2012/02/13/google-knowledge-graph-change-search/>
- [4] <http://mashable.com/follow/topics/social-graph/>
- [5] <http://mashable.com/2012/06/19/interest-graph-marketer/>
- [6] <http://mashable.com/2012/09/26/graph-databases/>