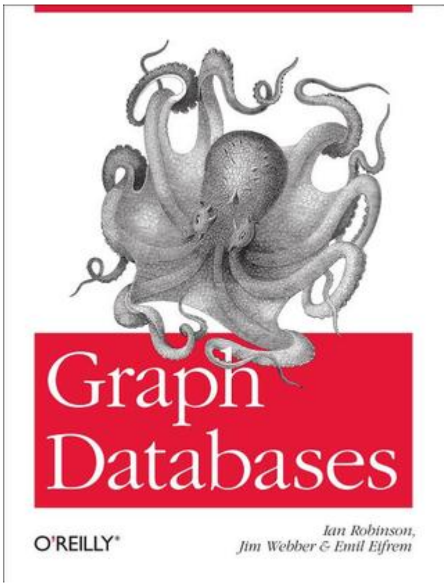

Graph Databases

Ian Robinson , Jim Webber , Emil Eifrem

Graph Databases

Ian Robinson , Jim Webber , Emil Eifrem

Graph Databases Ian Robinson , Jim Webber , Emil Eifrem
Discover how graph databases can help you manage and query highly connected data. With this practical book, you’ll learn how to design and implement a graph database that brings the power of graphs to bear on a broad range of problem domains. Whether you want to speed up your response to user queries or build a database that can adapt as your business evolves, this book shows you how to apply the schema-free graph model to real-world problems.

Learn how different organizations are using graph databases to outperform their competitors. With this book’s data modeling, query, and code examples, you’ll quickly be able to implement your own solution.

Model data with the Cypher query language and property graph model
Learn best practices and common pitfalls when modeling with graphs
Plan and implement a graph database solution in test-driven fashion
Explore real-world examples to learn how and why organizations use a graph database
Understand common patterns and components of graph database architecture
Use analytical techniques and algorithms to mine graph database information

Graph Databases Details

Date	:	Published June 17th 2013 by O'Reilly Media (first published June 1st 2013)
ISBN	:	9781449356262
Author	:	Ian Robinson , Jim Webber , Emil Eifrem
Format	:	Paperback 224 pages
Genre	:	Computer Science, Programming, Science, Technology, Nonfiction, Technical, Reference, Computers, Software

Download and Read Free Online Graph Databases Ian Robinson , Jim Webber , Emil Eifrem

From Reader Review Graph Databases for online ebook

John Warner says

I got the second free edition from neo4j and was not surprised that it is actually an introduction to neo4j and cypher (the query language) with some general graph database principles thrown in.

There was an assumption that I knew relational databases and their theory (which I do but not as a professional) and I was hoping for a book that grew from the ground up building the database theory from its theoretical roots. I know the limits of relational databases so didn't want reminding about them every time a new feature was discussed.

There were chapters I skipped as I am not going to be using the APIs etc - I am making toy databases to extend my knowledge and to see what I can do and maybe use in small projects to help myself and my daily life. So it felt like an industrial strength street-cleaner being used to sweep up a few crumbs.

Readers will learn from this book often if you are not doing it chapters can be omitted safely.

James says

Interesting but feels like an advert for Neo4j, which is unsuprising given the authors. After reading this I feel like every problem I see could be solved with a graph database. I think I need to try it out on a personal project and see how it holds together.

William Anderson says

While not dedicated to them specifically this book serves as a great introduction to Neo4j and Cypher. Overall it is a highly accessible introduction to graph databases that then shifts into giving architectural approaches for designing how your graph data is structured.

Rob says

Got it as a free book on neo4j's website. It is centered around neo4j, dont expect too much on graph databases in general. It is basically a collection of their blog posts. Generally, fairly informative and supported with decent examples but can be repetitive as it has not be edited too much to be read as a book.

Chen Qin says

I read early release.
So far, I felt graph database do have way to relief "join pain" in term of its native graph modeling.

What may be interested to hear is how partition of graph with billions of heterogeneous nodes cloud be made so that most heavily used query could be optimized from low bandwidth communications or etc.

Philipp says

Note: You can get the 2nd edition for free from the authors here: http://graphdatabases.com/

A good introduction to Neo4j, less to graph databases in general. It goes from the basics, to using the query language Cypher, to some use-cases, internal technologies, and a quick primer on graph algorithms (Dijkstra's algorithm, A*).

Very American, the tone is overenthusiastic to the point of being hilarious:

> [...]graph databases lack the kind of schema-oriented data governance mechanisms we're familiar with in the relational world. But this is not a risk; rather, it calls forth a far more visible and actionable kind of governance.

Translated: It's OK to give children live hand-grenades. But this is not a risk; just give them some rules for handling explosives.

It's similar overenthusiastic in the benchmarks, and there's an absolutely unholy code-example showing how to extract all nodes from a range of dates with no comment that a whole page of code with several MATCHes for a date1 <= x <= date2 is a *bit* too complicated. Then again, the authors made the database and work for the company that makes all of its money with Neo4j. At times the book feels more like a company white-paper.

The later chapters get more interesting - testing using Java, technical details of the implementation, algorithms, architecture patterns, caching etc. I would have liked more technical details but these are out there (and there are a few useful paper and book recommendations, here too!)

Recommended for: People who want to get their feet wet with graph databases, but don't want too much of the technical details

---

Xanan says

The first two chapters introduce graph databases and compare their modeling approach with Relational and NoSql modeling, illustrating the advantages of the former. Relational database knowledge is required to understand the discussion.

Chapter 3 goes deeper into graph database modeling by providing examples that illustrate sample graphs and basic creation and query statements with Cypher (a declarative graph query language used by Neo4j and other graph databases).

Chapter 4 is a hodgepodge of various topics. The first part deals with issues you may come across with graph data modeling, including fine-grained vs generic relationships, how to model time, and how to model facts (interactions between entities that involve time). It then goes on to describe the pros and cons of running graph databases as a standalone server of embedded in your application and clustering-related issues in Neo4j. It also touches on Neo4j support for applications testing and importing bulk data into a graph database.

Chapter 5 describes 3 real-world use cases where graph databases might be employed and shows Cypher queries that can be used in those scenarios. The chapter also serves to illustrate more complex Cypher queries, which are explain in detail.

Chapter 6 relies on Neo4j to illustrate how a graph database is actually stored on disk, and the different levels of APIs that developers can use to interact with it. This chapter is more technical than the others but not very detailed.

The final chapter contains a pretty detailed illustration of Dijkstra's algorithm for calculating the shortest path between a source and destination node in a weighted graph and also barely sketches the concept of informed search as a way of optimizing the search with the aid of additional information beyond graph edge weights. It concludes with concepts from predictive modeling where the existence of unknown relationships between entities in a social graph can be predicted on the basis of existing (known) relationships.

All in all the book gives a readable introduction to graph databases motivating their purpose in the context of existing alternative technologies, describes data modeling in adequate details.
Basic graph and relational database knowledge is required to fully understand the discussion.
The book uses Cypher for all examples and introduces features and language constructs along the way when needed. It doesn't provide a detailed reference to the language but the presentation is accessible and gives a good starting point.
The book also relies on Neo4j to provide implementation detail and example code besides Cypher queries.
Some advanced concepts are also introduced but barely sketched.

Thomas says

Good overview of graph database concepts, but very focused on Neo4j. I'm sure the basic concepts transfer to other databases, but all code examples are geared toward Neo4j, which the authors are heavily involved in developing.

Mark Seemann says

The title of this book ought to be *Neo4j, with a bit about general graph database theory*. While implying to be a book about graph databases in general, all examples specificly use Neo4j.

Queries are demonstrated using Cypher, Neo4j's query language. Other code examples show how to write custom Java code that uses Neo4j's various APIs.

That said, I didn't mind it much, because I knew that the authors are the creators and core developers of

Neo4j, and that Neo4j is the dominant graph database on the market. I expected the book to have this focus, even before I bought it.

The best about this book is that as soon as it gets technical, the writing and the examples are (with a few exceptions) crystal-clear. I liked those parts. As I wrote above, they are all about Neo4j, but I didn't mind, because I could easily extrapolate general patterns from the examples.

The not-so-good aspect of the book is that there's a lot of text that tells you how great and fantastic graph databases are, and those paragraphs are hard to get through. They're repetitive, the language is lacklustre and clichéd, and they have an overall sermonizing character. It's fairly obvious that there were more than one author involved.

Jan Kroken says

While some of the criticism is fair (it's focused on Neo4j rather than graph databases in general, it's too detailed for non-techies and a tad high level for techies and so on), I still found it to be a very good and informative read.

Atif Rahman says

Great first book on graph databases, very well explained and examples are easy to follow. Its from Neo4J, hence, no mention of TinkerPop and more open standards but rather focus on key unique selling propositions for its own platform. Examples are ofcourse in Cypher. Great book nevertheless and so goes for the database. (scalability in Neo4j is an anti-pattern otherwise)

David says

Just finished reading an early release version of 'Graph Databases' (to be published next month). While perhaps overly focused on Neo4j (not surprising, given the three authors), this is the best reference out there on graph data modeling. That said, I have a few comments:

- Graphs needn't be as schemaless as the authors indicate. I haven't used Neo4j, but they claim that types/labels on nodes are not yet directly supported (p40), so I guess this is one reason why they downplay schemas and ontologies. Remember that in a schema-less system, there is a very real implicit schema, but now it must be implemented as business logic in each application that processes the data.

- I think their modeling rules would need to be adjusted when attempting to naturally fuse disparate data sources at time of graph insert. This is especially true if vertex uniqueness is influenced by event time.

- They claim that "E-R diagrams only allow single, undirected, named relationships between entities" (p32), but surely they've seen many-to-many tables before, which can hold data specific to the relationships themselves.

- The discussion of 'lossy sentiment' (p52) is a frequent graph design gotcha that I would have liked to have seen a bit more formalized. Not sure how to do it offhand, but something like "You lose the explicit pairing between nodes when their connecting link topology is not granular enough" would have made me happy.

- I also disagree with statements like "don't encode entities into relationships. Use relationships to convey semantics about how entities are related" (p54). Good design should allow some links to be more than just named relationships, but less than requiring that they convert into an intermediate node.

- They didn't seem to like RDF triple stores, but didn't know how to properly criticism them. I could help. :-)

- Also, there is considerable advances being made in creating property graphs in column family-based aggregate stores that the authors haven't mentioned.

Comments aside, this is a great book on an important topic, and there's little out there to compete with it. Also, I liked the shout out to William Hartnell. :-)

Tomáš says

This is an OK book. It's more about Neo4j and its Cypher language than about graph databases, but, well, why not. My biggest complain is about the written style: pick up one thing, which is pain for relation databases (recursive joins) and compare it with simple example using graph database. I've missed some real life performance comparisons. I like the idea of graph databases, but this book didn't help me to understand the pros and cons of it.

Norbert Preining says

Well written introduction to graph databases by representatives of the Neo4j database. Clear structure, and the authors try to keep as far as possible independent from Neo4j while still explaining many concepts using it.

David says

This reads like an infomercial. There interesting ideas and material for sure, but I will look for a different reference.
