

NoSQL and SQL Data Modeling

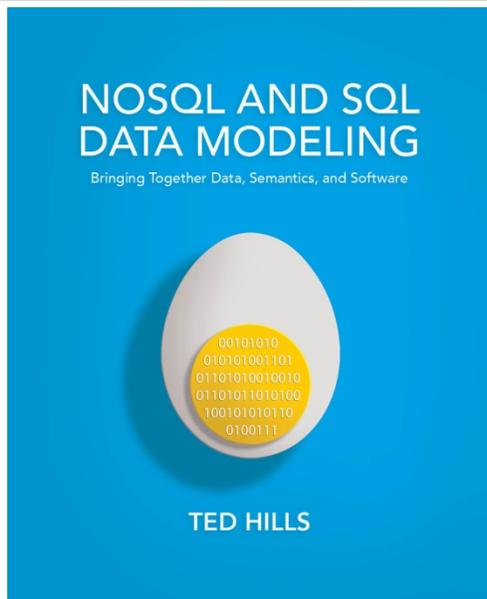
Bringing Together Data, Semantics, and Software

How do we design for data when traditional design techniques cannot extend to new database technologies? In this era of big data and the Internet of Things, it is essential that we have the tools we need to understand the data coming to us faster than ever before, and to design databases and data processing systems that can adapt easily to ever-changing data schemas and ever-changing business requirements. There must be no intellectual disconnect between data and the software that manages it. It must be possible to extract meaning and knowledge from data to drive artificial intelligence applications. Novel NoSQL data organization techniques must be used side-by-side with traditional SQL databases. Are existing data modeling techniques ready for all of this?

The Concept and Object Modeling Notation (COMN) is able to cover the full spectrum of analysis and design. A single COMN model can represent the objects and concepts in the problem space, logical data design, and concrete NoSQL and SQL document, key-value, columnar, and relational database implementations. COMN models enable an unprecedented level of traceability of requirements to implementation. COMN models can also represent the static structure of software and the predicates that represent the patterns of meaning in databases.

I believe that this is a breakthrough modeling technique – and it is technique, not just notation. COMN provides notation to handle all of the constructs that E-R techniques don't do well, and it steps up to the problem of linking physical and conceptual models. . . . I'm convinced that COMN is the future of data modeling.

—Dave Wells, BI and Analytics Educator and Consultant, Infocentric



This book will teach you:

- the simple and familiar graphical notation of COMN with its three basic shapes and four line styles
- how to think about objects, concepts, types, and classes in the real world, using the ordinary meanings of English words that aren't tangled with confused techno-speak
- how to express logical data designs that are freer from implementation considerations than is possible in any other notation
- how to understand key-value, document, columnar, and table-oriented database designs in logical and physical terms
- how to use COMN to specify physical database implementations in any NoSQL or SQL database with the precision necessary for model-driven development

Author: **Ted Hills**

ISBN, print ed. 9781634621090

ISBN, Kindle ed. 9781634621106

ISBN, ePub ed. 9781634621113

ISBN, PDF ed. 9781634621120

<http://www.TechnicsPub.com>

Ted Hills has been active in the Information Technology industry since 1975. At LexisNexis, Ted co-leads the work of establishing enterprise data architecture standards and governance processes, working with data models and business and data definitions for both structured and unstructured data. Ted has always been an active researcher, with interests in software and data integration, data modeling notations, and improving the expressivity of languages while keeping them type-safe.