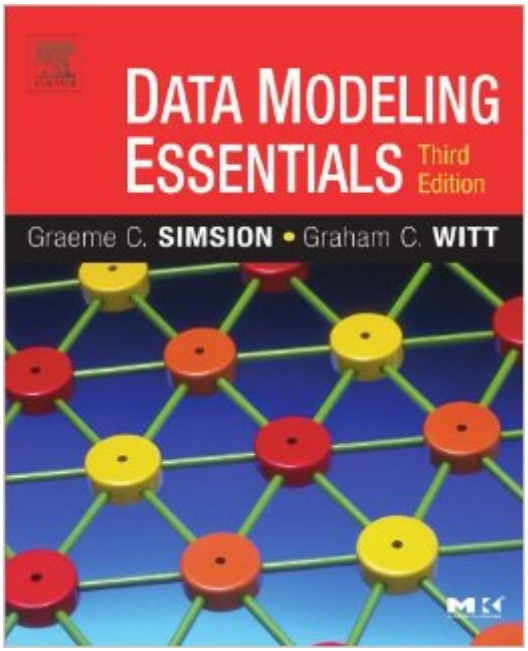
# Data Modeling Essentials, Third Edition



## Synopsis

Data Modeling Essentials, Third Edition provides expert tutelage for data modelers, business analysts and systems designers at all levels. Beginning with the basics, this book provides a thorough grounding in theory before guiding the reader through the various stages of applied data modeling and database design. Later chapters address advanced subjects, including business rules, data warehousing, enterprise-wide modeling and data management.The third edition of this popular book retains its distinctive hallmarks of readability and usefulness, but has been given significantly expanded coverage and reorganized for greater reader comprehension. Authored by two leaders in the field, Data Modeling Essentials, Third Edition is the ideal reference for professionals and students looking for a real-world perspective. Â  Thorough coverage of the fundamentals and relevant theory.Â  Recognition and support for the creative side of the process.Â  Expanded coverage of applied data modeling includes new chapters on logical and physical database design.Â  New material describing a powerful technique for model verification.Â  Unique coverage of the practical and human aspects of modeling, such as working with business specialists, managing change, and resolving conflict.Â  Extensive online component including course notes and other teaching aids (www.mkp.com).UML diagrams now available! Visit the companion site for more details.Click here to view a book review by Steve Hoberman!

## Book Information

Paperback: 560 pages

Publisher: Morgan Kaufmann; 3rd edition (November 18, 2004)

Language: English

ISBN-10: 0126445516

ISBN-13: 978-0126445510

Product Dimensions:  7.5 x 1.3 x 9.2 inches

Shipping Weight: 2.6 pounds (View shipping rates and policies)

Average Customer Review:  4.2 out of 5 starsÂ Â See all reviewsÂ (34 customer reviews)

Best Sellers Rank: #305,129 in Books (See Top 100 in Books)  #54 inÂ Books > Computers & Technology > Programming > Software Design, Testing & Engineering > Structured Design  #121 inÂ Books > Computers & Technology > Networking & Cloud Computing > Network Administration > Storage & Retrieval  #229 inÂ Books > Computers & Technology > Databases & Big Data > Data Processing

## Customer Reviews

Excellent practical introduction to data modeling in the relational paradigm using entity-relationship techniques. Detailed and instructive discussions weighing the relative merits of alternative models for scenarios. Positions data modeling within the context of developing information systems for business. Real-world, messy examples of the kinds of problems and errors that can arise-some of them a bit contrived, but usually to make a good point. A number of respectable sources footnoted, but unfortunately no bibliography.Proposes evaluation criteria for measuring model quality. Admits conflict among these criteria-all desirable attributes of a model cannot be optimized simultaneously. Trade-offs must be made. Recognizes the limits of data modeling: "Don't try to solve every problem by developing a conventional data model (p. 265)."Emphasizes that data modeling, although often confused with analysis, is not analysis. It is design. There is no one correct model for every scenario. Advocates using creativity to propose multiple alternative models before selecting a solution. Establishes the role of the data modeler by analogy with that of a residential architect.Interestingly, goes on to say that the distinction between analysis and design is important-without ever drawing it. Does not describe data "analysis," if such a thing even exists.Differentiates between data model and database design. Mainly because the paradigm used to represent the data while modeling it with the database customer (relational tables & columns, in this case) might differ from the paradigm that the database uses to represent it (network or hierarchy, perhaps). More recently, it has become common to model a solution with customers using the object paradigm and to implement it with database software using the relational paradigm. The paradigms need not always differ, but when they do, a translation is required before building the database.Addresses not just how a data model works, but also how to build one, including the people to involve, the inputs to consult, and the sequence of tasks. Suggests various approaches, including top-down (entity-relationship modeling from scratch), bottom-up (using existing documents), and the customization of existing models and model fragments.Covers the five normal forms of relational data, not omitting the limits of normalization and the assumptions on which it is based. Contrasts normalization with entity-relationship modeling as "bottom-up" versus "top-down," the former emphasizing technical soundness and the latter emphasizing business suitability. Admits that normalization is usually performed explicitly only as a final check after entity-relationship modeling-if at all. Examples show importance of normalization.Numerous interesting observations on type hierarchies and generalization.Notes compromise between representing business rules with specific data structures and accommodating business change with generic data structures: the more rules are represented in data structure, the more susceptible is that structure to future change. Unstable rules are better represented in program code or in data values-both easier to change than

the structure of a production database. Cites frequency of both over-generic and over-specific models.Makes the important point that data models represent not the real world, but rather WHAT WE KNOW about it. Some data models quite properly assert that a person might be neither man nor woman-because a business might not know the gender of every person in which it has an interest. Personally, I would go a little further by adding that a model represents only what we CARE to know.Marring the otherwise valuable discussion of type hierarchies is their misapplication to modeling the various roles in which persons and organizations might act. A role may by nature be assumed and abandoned without changing identity. Using a subtype to represent it forces the subtype's instances to become and then to "unbecome" instances of the subtype as they change their roles-an obvious absurdity. We would indeed venture too far into the spirit world to claim that one might cancel membership in Homo Sapiens while retaining membership in Mammalia for the purpose of exercising at some later date the option to reincarnate as a chimpanzee!Points out necessity of asymmetry in implementation of recursive many-to-many relationship. Debunks some previously asserted "rules" regarding relationships. Discusses transferability of relationships and uses this concept in discussing one-to-one relationships, foreign keys in primary keys (weak entities), and time-dependent relationships.Interesting details on attributes that many similar books skip-particularly in the section on attribute generalization.Sadly accepts the notion that all of a model's codes might be implemented very nicely in one big table. This idea is an abomination. It impedes the evolution of "code entities" into non-trivial entities. It complicates enforcement of referential integrity. The suggestion of views for isolating cohesive subsets of the big code table defeats the very data-driving that code tables are built to enable.Also errs in proposing Code as a proper supertype for a "code entity." Code is a meta-entity. It represents nothing in the domain of the data model. In that domain it is not a supertype of anything. It would make as much sense to say that each thing is a type of Word because it has a word to describe it. It is valuable to recognize the common processing shared by many codes, but that commonality does not by itself imply a supertype.Good exposition of the option to use data structure, program code, or data value to enforce a business rule.Advises representing rules in the entity-relationship diagram using features for which there is "little intention of actually implementing (p. 269)." Type hierarchies are particularly recommended in this regard-even if they are not valid partitionings. Certainly, there are rules dependent on the values of attributes, but let's not make each attribute the basis of a subtype partitioning just to permit their graphic depiction! Advocates graphic depiction for communication with business customers even though diagrams are notoriously difficult for business customers. Diagrams are best suited to DBAs and programmers, but they are the very ones who wish not to

see them cluttered with unimplemented constructs!Quibbles and quips notwithstanding, a good book on one of my favorite subjects.

Simsion and Witt's _Data Modeling Essentials_ has been a classic on my data management bookshelf since the first edition. Now in the 3rd Edition, this work has become even more valuable and useful on data management projects. The fact that the authors continue to enhance and expand their work is a real asset.This work is targeted at both students and experienced information technology professionals....and, of course, any data modeling book that manages to quote from Led Zeppelin's "Stairway to Heaven", Stephen Covey's _7 Habits of Highly Effective People_, Bob Dylan's "Brownsville Girl", and even Jack Kerouac must be a good read, right?Let's start with what I really like about this book:1) _Essentials_ starts at the beginning "What is a Data Model" and works its way through entities, attributes, subtypes, ERDs, normalization and all the basics through to fairly advanced topics such as the use of surrogate keys, transformations, designing for performance, time dependence and advanced normalization. Simsion and Witt make this trek in a balanced and clearly-explained manner. This is no _...For Dummies_ type work - it is a true professional level book that consistently targets the whys, why-nots, how-much and when-to-stops of data modeling.2) Along the way, the authors refer to multiple methods, notations, and tools, while sticking with a single notation throughout. I much prefer data modeling books like _Essentials_ that use the most common notation in modeling, as these books are more useful in a variety of contexts over those that use more obscure notations. I can see how this edition has updated references to tool features and modeling support.3) _Essentials_ includes discussions that are, more often than not, left out of technical works in the data management field. For instance, most of the topics include references to myths, trade-offs, and real world issues. The authors' willingness to explore these topics is, in my opinion, a sign of maturity of this book. So many technical texts in database design completely ignore the trade-offs in tuning, simplifying design, and working with external constraints, etc., but the authors jump right in and give their opinions on what is best.4) This book contains a substantial amount of material on the development of physical models and databases. Many data modeling books treat this area lightly and I find the authors' thoroughness in this area a really strength. Many logical data modelers struggle with turning beautiful designs into working databases and _Essentials_ does a great job of explaining the trade offs in a non-DBMS-specific manner.This 3rd edition expands in these areas to become a true professional's guide to data modeling.What I didn't like about Essentials:1) While the majority of the work uses contemporary terminology and notation, there are still some terms with currency issues. For instance, when describing process models, the

examples use Data Flow Diagramming notation, something that is not quite a common as it used to be and can be perceived as dated. On the other hand, what did the authors choose to call the boxes on a data model? "Entity Classes", in deference to what object modelers chose to call these boxes. The authors believe that this deference will improve communications between modelers. I don’t agree. Having borrowed a term from the object crowd, how does the book refer to modelers? "E-R modelers", a term that is rare and dated. And in many places, instead of referring to data models, they are called "E-R models". Data modeling tools are referred to as "documentation tools" or "CASE tools" - these are also dated terms. Perhaps in the 4th edition we will see a complete updating of terminologies and notations.2) As a textbook, this work recommends approaches that are not suitable for novice modelers. For instance, the authors recommend the use of dummy rows and special dummy words in databases to avoid Nulls, the use of multi-valued attributes (not columns, attributes), etc. Of course these things happen in the real world, but to recommend them in a text without sufficiently covering the down sides of these approaches is going to get a few newbie modelers in hot water.3) As a professional guide, the definition of "Logical Model" as a model that is DBMS-specific is not a well-accepted definition and will cause confusion when professionals work with others who define a Logical Model as a model that is DBMS-independent.4) I believe that the introduction of Normalization in Chapter 2 is premature. Many normal forms can be `met’ by following good modeling practices. If these practices were introduced in an appropriate manner, the authors could then show how these practices support normalization.5) As I have said in reviews of other data modeling works, I hold text book examples to a higher standard. _Essentials_ uses an entity and relationship naming standard that is overly prone to errors and misunderstandings: infinitive-based verb phrases with a "be" form in the reverse relationship. This leads unfortunately weak relationship names, such as those in figure 10.3 Insurance Model:a. Policy Type may classify Policy / Policy must be classified by Policy Type (using may and must based on optionality)b. Policy must involve Person Role in Policy / Person Role in Policy must be for PolicyI’m not sure how to interpret these. Why is "involve" the reverse of "be for"? What does the term "be for" really mean, anyway? What does "be of" mean?What if I don’t want to introduce cardinality in my business sentences? I’d get sentences such as "Person Role in Policy be for Policy". What business user wants to work with a model that has assertions such as that? What does the relationship that is named "nominate" on one side and "be party to" on the other really mean? This sounds like I may just be nitpicking, but I continue to find this be-form and infinitive verb style prone to errors and I wish authors would give up on it in textbooks. If the authors can’t make it work, how will the students?OverallWhile I’ve mentioned a handful of things I didn’t like in this work, I still highly

recommend it. I especially appreciate the approach to topics that most authors shy away from. This is a substantial work - it has goodies for new modelers, intermediate modelers, and advanced modelers. _Data Modeling Essentials_ is my number one recommended how-to data modeling book. It is the perfect balance of theory and practice, giving the reader both the foundation and the tools to deliver high-quality data models.Disclaimer: I was a pre-publication reviewer for this work and received compensation, including copy of the book, for providing a review based on my data modeling experience. I receive no compensation from the publisher related to sales or promotion of this book.

The most valuable book I own about data modelling.Covers all the basics one needs to know if they are going to talk about data modelling and what it really means and what is involved. And if you are expected to actually do the data modelling, even better, it provides coverage on all the things you should include, or at least consider, as well as some insights on how you are going to show the value and importance of being able to model your data.Even a seasoned modeller like myself wants to refer to a solid piece of reference material to ensure I'm doing the right thing and that I'm not forgetting anything.I was very happy to see a section on Conceptual Data Modelling as I find myself spending more time in this space getting the business to recognise that 'they' own this model, and they should identify and define all their business attributes here. That way when the database, or data interface, needs to be built the logical model can be created using these conceptual models as a reference point... should be less argument on what that column in the database really means!So many more highlights... but maybe I'm just a fanatic about data... it's my coffee table book.Thanks Graeme and Graham.

Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data (Data-Centric Systems and Applications) AACN Essentials of Progressive Care Nursing, Third Edition (Chulay, AACN Essentials of Progressive Care Nursing) Chocolate Modeling Cake Toppers: 101 Tasty Ideas for Candy Clay, Modeling Chocolate, and Other Fondant Alternatives Atmospheric and Space Flight Dynamics: Modeling and Simulation with MATLABÃ Â® and SimulinkÃ Â® (Modeling and Simulation in Science, Engineering and Technology) The Model's Bible & Global Modeling Agency Contact List - An Insider's Guide on How to Break into the Fashion Modeling Industry Modeling Agency Tips: Get Listed with Fashion Modeling Agencies and Find Your Dream Job Data Modeling Made Simple: A Practical Guide for Business and IT Professionals, 2nd Edition Modeling Risk (Third Edition) Mastering Data Modeling: A User-Driven Approach Data Modeling: A Beginner's Guide UML and Data Modeling: A Reconciliation