

✦ EXCLUSIVELY FOR TDWI MEMBERS

FOURTH QUARTER 2009

TEN MISTAKES TO AVOID

In Dimensional Design

By Christopher Adamson



10 TEN MISTAKES TO AVOID

In Dimensional Design

By Christopher Adamson

FOREWORD

In virtually every data warehouse implementation, you can find the products of dimensional design: the star schema, the snowflake, or the cube. Despite this near-universal acceptance, the basic principles of dimensional design are commonly misunderstood and misapplied.

Many mistakes are errors in approach, committed before design work begins. Chief among these is the failure to exploit strategic functions offered by a dimensional model. Others include waiting until the design stage to do dimensional design, then leaving the design work to designers.

Success is often thwarted by common technical errors, such as sacrificing operational detail or taking shortcuts with historic data. Designers also doom their solutions by failing to adapt them to the software tools that comprise the data warehouse.

Avoiding these 10 mistakes in dimensional design will help you ensure successful implementations in any data warehouse architecture, including those advocated by Ralph Kimball and W.H. Inmon.


ABOUT THE AUTHOR

Christopher Adamson is an independent consultant, educator, and author. He works with customers in all industries to develop data warehouse strategies, define and prioritize projects, and design solutions. Chris is a faculty member at TDWI and teaches classes in dimensional design. He is the author of *Mastering Data Warehouse Aggregates* (John Wiley & Sons, 2006) and *Data Warehouse Design Solutions* (John Wiley & Sons, 1998). Chris blogs about data warehousing at www.starschemacentral.com and can be contacted at chris.adamson@oaktonsoftware.com.

© 2009 by TDWI (The Data Warehousing Institute™), a division of 1105 Media, Inc. All rights reserved. Reproductions in whole or in part are prohibited except by written permission. E-mail requests or feedback to info@tdwi.org.

Product and company names mentioned herein may be trademarks and/or registered trademarks of their respective companies.

MISTAKE NINE: REUSING UNIQUE IDENTIFIERS



A dimensional model must have its own unique identifier for each dimension. This allows flexibility in responding to changes in source data. When a customer record is updated, for example, the dimensional model may need to accommodate current and historic versions. The source system's unique identifier will not be enough to distinguish between them.

The customary solution is to design surrogate keys for each dimension. Surrogate key values are usually populated from a sequence and have no business meaning—their sole purpose is to serve as unique identifiers. Several historic versions of the same customer, for example, can be stored in the same dimension table; each will have a different surrogate key value. A separate fact table can use these values to refer to historically accurate versions of the customer when recording facts about sales.

Source systems sometimes maintain change logs. Designers often mistakenly assume this obviates the need for distinct unique identifiers in the dimensional model. In fact, a dimension rarely corresponds to a single entity in an operational system. A customer dimension, for example, may consolidate operational data that appears in a number of source tables, including customers, addresses, industry codes, and so forth. A change log for the customer table may not reflect changes to some of these other tables, particularly if they are linked via association tables. Without its own unique identifier, the dimensional model cannot maintain this history.