

DATA ENGINEERING AND ANALYTICS

Data Warehousing and Dimensional Modeling

Level: Practitioner • 2 days (expandable to 3) • Virtual, In-person

Overview

The tools of analytics change constantly; the shape of a good analytical model barely changes at all. Star schemas, facts, and dimensions have outlived three generations of platforms because they solve a permanent problem: business people ask questions in terms of measures and the things that describe them, and a model built that way answers those questions fast and stays understandable. The hard part is the modeling judgment: choosing the grain, handling change over time, and resisting the urge to model the source system instead of the business.

This is a hands-on, practitioner course. It builds dimensional modeling from the ground up: first why warehouses exist and how they differ from transactional databases, then the core star schema pattern, then the harder design decisions around dimensions and facts, and finally how models get loaded and how they fit modern cloud and lakehouse platforms. In keeping with a less-but-deeper philosophy, we go deep on the design skill that transfers across every platform rather than surveying vendor features. Every module ends with a modeling or SQL lab, and each module builds on the one before.

Who Should Attend

- BI developers and analysts who build or maintain reporting databases
- Data engineers who load warehouses and want to design the models, not just fill them
- Database developers and architects adding analytical design to a transactional background

Prerequisites

- Solid SQL: joins, aggregation, and GROUP BY (see *SQL Querying and T-SQL Fundamentals* if you need a foundation)
- Familiarity with relational database concepts; *Relational Database Design* covers the transactional side of this coin
- No prior warehousing experience required

What You Will Learn

- Explain why analytical and transactional databases are designed differently
- Design a star schema from business requirements, starting with a deliberate choice of grain
- Design dimensions that handle change over time, including slowly changing dimension types
- Choose the right fact table type (transaction, periodic snapshot, accumulating snapshot) for a business process
- Build the ETL patterns that load and maintain a dimensional model
- Judge where dimensional modeling fits on modern platforms, from SQL Server to cloud warehouses and lakehouses

Course Outline

Day one: why warehouses, and the star schema

- Why Data Warehouses Exist
 - OLTP versus OLAP: why one database rarely serves both well
 - Warehouse architectures: staging, the warehouse itself, and the serving layer
 - Kimball's approach in one picture, and the vocabulary the rest of the course uses
 - Lab: take a set of business questions and show why the source system answers them badly
- The Star Schema
 - Facts and dimensions: measures versus the context that describes them
 - Grain: the single most important decision in any dimensional model
 - Star versus snowflake, and why simpler usually wins
 - Lab: design a first star schema from a business process description
- Designing Dimensions
 - Surrogate keys and why natural keys are not enough
 - Slowly changing dimensions: type 1, type 2, and when history matters
 - Hierarchies, date dimensions, and conformed dimensions across business processes
 - Lab: design a customer dimension with type 2 history and a reusable date dimension

Day two: facts, loading, and the modern context

- Designing Fact Tables
 - Transaction, periodic snapshot, and accumulating snapshot facts, and how to choose
 - Additive, semi-additive, and non-additive measures
 - Degenerate dimensions, factless facts, and other patterns you will actually meet
 - Lab: extend the model with a second fact table at a different grain
- Loading the Warehouse
 - ETL for dimensional models: staging, lookups, and surrogate key handling
 - Processing slowly changing dimensions and late-arriving data
 - Incremental loads and auditing what was loaded when
 - Lab: write the SQL that loads a type 2 dimension and its fact table incrementally
- Dimensional Modeling in the Modern Stack
 - Cloud warehouses and lakehouses: what changes, and how much of Kimball survives (most of it)
 - Wide tables, semantic layers, and where BI tools like Power BI meet the model
 - When not to build a warehouse: honest alternatives for small teams
 - Lab: adapt the course model to a gold layer for a BI tool and defend the design choices

Extended Version

The three-day version keeps the same gradient and adds depth and a full design cycle:

- Harder modeling cases: many-to-many bridges, ragged hierarchies, and multi-currency measures
- A deeper treatment of ETL automation and testing for warehouse loads
- Performance and indexing considerations for dimensional models on SQL Server and cloud platforms

- A capstone that designs, defends, and loads a complete multi-process dimensional model from a realistic business brief