

创新性应用数据建模经验谈 PDF转换可能丢失图片或格式，
建议阅读原文

https://www.100test.com/kao_ti2020/141/2021_2022__E5_88_9B_E6_96_B0_E6_80_A7_E5_c29_141310.htm 笔者从98年进入数据库及数据仓库领域工作至今已经有近八年的时间，对数据建模工作接触的比较多，创新性不敢谈，本文只是将工作中的经验总结出来，供大家一同探讨和指正。提起数据建模来，有一点是首先要强调的，数据建模师和DBA有着较大的不同，对数据建模师来说，对业务的深刻理解是第一位的，不同的建模方法和技巧是为业务需求来服务的。而本文则暂时抛开业务不谈，主要关注于建模方法和技巧的经验总结。从目前的数据库及数据仓库建模方法来说，主要分为四类。来源

：www.examda.com 第一类是大家最为熟悉的关系数据库的三范式建模。通常我们将三范式建模方法用于建立各种操作型数据库系统。第二类是Inmon提倡的三范式数据仓库建模。它和操作型数据库系统的三范式建模在侧重点上有些不同。Inmon的数据仓库建模方法分为三层，第一层是实体关系层，也即企业的业务数据模型层，在这一层上和企业的操作型数据库系统建模方法是相同的；第二层是数据项集层，在这一层的建模方法根据数据的产生频率及访问频率等因素与企业的操作型数据库系统的建模方法产生了不同；第三层物理层是第二层的具体实现。第三类是Kimball提倡的数据仓库的维度建模。我们一般也称之为星型结构建模，有时也加入一些雪花模型在里面。维度建模是一种面向用户需求的、容易理解的、访问效率高的建模方法，也是笔者比较喜欢的一种建模方式。第四类是更为灵活的一种建模方式。通常用于

后台的数据准备区，建模的方式不拘一格，以能满足需要为目的，建好的表不对用户提供接口，多为临时表。下面简单谈谈第四类建模方法的一些的经验。数据准备区有一个最大的特点，就是不会直接面对用户，所以对数据准备区中的表进行操作的人只有ETL工程师。ETL工程师可以自己来决定表中数据的范围和数据的生命周期。下面举两个例子：1) 数据范围小的临时表 当需要整合或清洗的数据量过大时，我们可以建立同样结构的临时表，在临时表中只保留我们需要处理的部分数据。这样，不论是更新还是对表中某些项的计算都会效率提高很多。处理好的数据发送入准备加载到数据仓库中的表中，最后一次性加载入数据仓库。2) 带有冗余字段的临时表 由于数据准备区中的表只有自己使用，所以建立冗余字段可以起到很好的作用而不用承担风险。举例来说，笔者在项目中曾遇到这样的需求，客户表{客户ID，客户净扣值}，债项表{债项ID，客户ID，债项余额，债项净扣值}，即客户和债项是一对多的关系。其中，客户净扣值和债项余额已知，需要计算债项净扣值。计算的规则是按债项余额的比例分配客户的净扣值。这时，我们可以给两个表增加几个冗余字段，如客户表{客户ID，客户净扣值，客户余额}，债项表{债项ID，客户ID，债项余额，债项净扣值，客户余额，客户净扣值}。这样通过三条SQL就可以直接完成整个计算过程。将债项余额汇总到客户余额，将客户余额和客户净扣值冗余到债项表中，在债项表中通过（ $\text{债项余额} \times \text{客户净扣值} / \text{客户余额}$ ）公式即可直接计算出债项净扣值。另外还有很多大家可以发挥的建表方式，如不需要主键的临时表等等。总结来说，正因为数据准备区是不对用户提供接口的，所以我们

一定要利用好这一点，以给我们的数据处理工作带来最大的便利为目的来进行数据准备区的表设计。行业借鉴经验：数据仓库架构经验谈 对于数据仓库的架构方法，不同的架构师有不同的原则和方法，笔者在这里来总结一下当前常采用的架构方式及其优缺点。这些架构方式不限于某个行业，可以供各个行业借鉴使用。首先需要说明的一点是，目前在数据仓库领域比较一致的意见是在数据仓库中需要保留企业范围内一致的原子层数据。而独立的数据集市架构（Independent data marts）没有企业范围内一致的数据，很可能会导致信息孤岛的产生，除非在很小的企业内或只针对固定主题，否则不建议建立这样的架构方式。联邦式的数据仓库架构

（Federated Data Warehouse Architecture）不管是在地域上的联邦还是功能上的联邦都需要先在不同平台上建立各自的数据仓库，再通过参考（reference）数据来实现整合，而这样很容易造成整合的不彻底，除非联邦式的数据仓库架构也采用Kimball的总线架构（Bus Architecture）中类似的功能，即在数据准备区保留一致性维度（Conformed Table）并不断更新它。所以，这两种架构方式不在讨论范围之内。下面主要讨论剩下的三种架构方式。100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com