

Kimball Design Tip #55: Exploring Text Facts

By Bob Becker

In this Design Tip, we return to a fundamental concept that perplexes numerous dimensional modelers: text facts (also referred to as fact indicators, attributes, details or notes).

Some of you may be rightfully saying that text facts are a dimensional modeling oxymoron. However, we frequently field questions from clients and students about indicator, type or comment fields that seem to belong in the fact table, but the items are not keys, measurements, or degenerate dimensions (see [Design Tip #46](#)).

Generally, we recommend not modeling these so-called text facts in the fact table, but rather attempt to find an appropriate home for them in a dimension table. You don't want to clutter the fact table with several mid-sized (20 to 40 byte) descriptors. Alternatively, you shouldn't just store cryptic codes in the fact table (without dimension decodes), even though we are quite certain EVERYONE knows the decodes already.

When confronted with seemingly text facts, the first question to ask is whether they belong in another dimension table? For example, customer type likely takes on a single value per customer and should be treated as a customer dimension attribute.

If they don't fit neatly into an existing core dimension, then they should be treated as either separate dimensions or separate attributes in a junk dimension. It would be straightforward to build small dimension tables that assigned keys to all the payment or transaction types, and then reference those keys in the fact table. If we get too many of these small dimension tables, you should consider creating a junk dimension. We discussed junk dimensions in Design Tip #48 last year. There are several considerations when evaluating whether to maintain separate dimensions or to lump the indicators together in a junk dimension.

- Number of existing dimension foreign keys in the fact table. If you're nearing 20 foreign keys, then you'll probably want to lump them together.
- Number of potential junk "combination" rows, understanding that the theoretical combinations likely greatly exceed the actual encountered combinations. Ideally we'd keep the size of the junk dimension to less than 100,000 rows.
- Business relevance or understanding of the attribute combinations. Do the attributes have so little to do with each other that users are confused by the forced association in a junk dimension?

Finally, what should you do when the supposed "fact" is a verbose, free-form text field that takes on unlimited values, such as a 240-byte comment field? Profiling the field, then parsing and codifying would make it most useful analytically, but that's almost always easier said than done.

It's been our experience that if the field is truly free-form, it is seldom accessed analytically. Usually these comment fields are only valuable to support a detailed investigation into suspicious transactions on an occasional basis. In this event, you'll want to put the text into a separate dimension rather than carrying that extra bulk on every fact record.