

Kimball Design Tip #44: Don't Be Overly Reliant On Your Data Access Tool's Metadata

By Bob Becker

"Oh, we'll handle that in the tool" is the refrain we sometimes hear from design teams. Instead, whenever possible, we suggest you invest the effort to architect as much flexibility, richness, and descriptive information directly into your dimensional schemas as possible rather than leaning on the capabilities of the tool metadata as a crutch.

Today's business intelligence tools provide robust metadata to support a wide range of capabilities, such as label substitution, predefined calculations, and aggregate navigation. These are useful features for your business community. But, we need to be judicious in the use of features the tools provide. Too often design teams take shortcuts and rely on the data access tool metadata to resolve issues that are better handled in our dimensional models. The end result is business rules that become embedded in the tool metadata rather than in our schemas. We also see design teams utilize tool metadata to provide code lookups and indicator descriptors in a misguided effort to keep their schema smaller and tighter.

The biggest drawback to these shortcuts is the dependence on the end user tool metadata to enforce business rules. If we rely on the tool metadata to implement business rules, every user must access the data via the "supported" tool to guarantee business users are presented with "correct" data. Users that want or need to use another access method are forced to recreate the business rules buried in the tool metadata to be assured of correct results.

As data warehouse developers, we need to protect against situations where business users might see different results depending on the tool they elect to use. Regardless of how they access the data warehouse data, users should get the same high quality, consistent data.

You may be thinking, "Fine, then we'll force all users to access the data warehouse through our supported tool." However, this approach will inevitably fall apart. There are a number of reasons an individual may need to access the data warehouse through some other means, bypassing the supported tool and therefore any business rules enforced via its metadata. These scenarios may not exist in your organization at the time you are developing your schema, but rest assured; one of them will arise during your watch:

- An IT professional (perhaps you) may elect to use SQL directly against the data warehouse data to resolve a complex query or audit data.
- Your organization may develop analytic applications based on custom written SQL-based queries directly against the data warehouse.
- Statistical modeling tools and/or data mining tools may need to directly access the data warehouse data.
- A sophisticated user armed with Microsoft Access (or another non-supported tool) may be granted direct access to the data warehouse.
- There may become a need to supplement the data warehouse with multidimensional "cubes" drawn directly from the data warehouse.
- Your organization may select another end user tool, yet not replace the current tool.

None of this should be construed as an argument against leveraging the capabilities of your data access tool. Rather the key is that when in doubt or confronted with a choice, we prefer the design choice that places a capability as close to the data as possible to assure that the capability is available to as wide an audience as possible.