

Kimball Design Tip #40: Structure Of An Analytic Application

By Bill Schmarzo

A comprehensive analytic application environment needs to support a framework that moves users beyond standard reports. The environment needs to proactively “guide” users through the analysis of a business situation, ultimately helping them make an insightful and thoughtful decision. The goals of this analytic application lifecycle are to:

- Proactively “guide” business users beyond basic reporting
- Identify and understand exceptional performance situations
- Capture decision-making “best practices” for each exceptional performance situation
- Share the resulting “best practices” or intellectual capital across the organization

We break the process into five distinct stages:

1. PUBLISH REPORTS -- provides standard operational and managerial “report cards” on the current state of the business.
2. IDENTIFY EXCEPTIONS -- reveals the exceptional performance situations (both over- as well as under-performance) to focus attention.
3. DETERMINE CAUSAL FACTORS-- seeks to understand the “why” or root causes behind the identified exceptions.
4. MODEL ALTERNATIVES -- provides a backdrop to evaluate different decision alternatives.
5. TRACK ACTIONS -- evaluates the effectiveness of the recommended actions and feeds the decisions back to both the operational systems and data warehouse (against which Stage 1 reporting will be conducted), thereby closing the loop.

Let’s walk through each of these stages in a bit more detail to understand their objectives and impact on our data warehouse architecture.

Stage 1: PUBLISH REPORTS. Standard operational and managerial reports are the starting point for the analytic applications lifecycle. These reports look at the current results versus plan or previous periods in order to provide a report card on the state of the business (e.g., “Market share is up 2 points, but profitability is down 10%”).

Data warehouse requirements in the PUBLISH REPORTS stage focus on improving the presentation layer and includes presentation technologies such as dashboards, portals and scorecards. While many data warehouse implementations successfully deliver reports, they stop at the PUBLISH REPORTS stage and declare success.

Stage 2: IDENTIFY EXCEPTIONS. This stage focuses on the identification of “what’s the matter?” or “where are the problems?” phase of the analysis. The focus of this stage is to identify the exceptions to normal performance as well as the opportunities. Most business managers have asked the data warehouse team to replicate a stack of reports in the data warehouse, when in reality, they really

want the parts of the reports marked with highlighters and yellow stickies. The exceptions stage is essential in helping users wade through the deluge of data to focus on the opportunities offering the best business return; those deserving the most attention.

The IDENTIFY EXCEPTIONS stage implies new capabilities such as broadcast servers which distribute alerts to users' devices of choice based upon exception "triggers," and visualization tools to view the data in different more creative ways such as trend lines, geographical maps or clusters.

Stage 3: DETERMINE CAUSAL FACTORS. This stage tries to understand the "why" or the root causes of the identified exceptions. Identifying reliable relationships and interactions between variables that drive exceptional performance is the key.

Successfully supporting users' efforts in this stage will require our data warehouse architecture to include additional software such as statistical tools and/or data mining algorithms (i.e., association, sequencing, classification, and segmentation) to quantify cause-and-effect.

Stage 4: MODEL ALTERNATIVES. In this stage we build on cause-and-effect relationships to develop models for evaluating decision alternatives.

The ability to perform what-if analysis and simulations on a range of potential decisions is considered the final goal when following a typical analytic applications lifecycle. We hope to successfully answer strategic questions such as: What happens to my market share, revenue and units sold if I achieve a greater than 10% price differential versus my top 2 competitors? Or, what are the impacts upon my inventory costs if I can achieve 5% sales forecast accuracy instead of my usual 10%? Our data warehouse architecture will need to accommodate additional technologies in the MODEL ALTERNATIVES stage including statistical tools and data mining algorithms for model evaluation, such as sensitivity analysis, Monte Carlo simulations, and optimizations (goal seeking).

Stage 5: TRACK ACTIONS. This stage tracks the effectiveness of the decisions recommended by the previous stage. Ideally, we can enable a "closed loop" process and feed the recommended actions back to the operational system. The effectiveness of the decisions should be captured and analyzed in order to continuously fine-tune the analysis process, business rules and models.

The TRACK ACTIONS stage places additional demands on our data warehouse architecture. We need to enable effective closed loop capabilities back to the operational systems and the data warehouse. We also recommend enhancing existing dimensional models or building performance management tracking data marts to record and track the results of specific business decisions to determine which decisions worked and which ones didn't. Emerging technologies applicable in this area include broadcast servers which will soon enable users to respond with recommended actions from their device of choice (e.g., e-mail, PDA, pagers, WAP phones) not just deliver alerts.

This Design Tip is only a "teaspoon sip" introducing the structure of an analytic application. A more comprehensive column on this topic will appear in Intelligent Enterprise magazine in a few weeks.