



"turning data into dollars"

Tom's Ten Data Tips – June 2006

### Data Warehousing

Data Warehousing was an innovation from the 90's that promised to change the data landscape for good. How far have we come? Many vendors have entered the marketplace because it makes sense to bring together data from throughout the organization, and this will continue to make sense in the future.

How large the Data Warehouse market will grow nobody knows yet. But for sure it is still growing fast, and currently is estimated at 4,5 billion dollar per year (IDC).

#### 1. Why Do Data Warehouse Projects Run Into Scope Creep?

To quote Bill Inmon (guru and author of several great books on Data Warehousing) "Traditional projects start with requirements and end with data. Data Warehousing projects start with data and end with requirements."

As soon as the project gets under way, users will find new applications, and with it will come new requests for data. Interestingly, these projects often are justified by moving Q&R work away from the 'data people'. What we've seen is that the first thing that happens as soon as the project delivers is that *more* requests for special queries are submitted to these same 'data people'. This may appear to undermine the initial business case but actually signals the onset of value creation from the DWH project.

#### 2. Star Schema Versus Entity Relation Model?

There has been enormous debate in the community about the merits of different data models. At the risk of over simplifying: ER models tend to have better performance (processing time) for the end user, and are often perceived as "easier" to understand by end users. Drawbacks are that ER models require more disk space, and, because of the intrinsic redundancy in the data, have consistency problems from a maintenance perspective.

Having said this, the practice seems to be that often some combination of the two is unavoidable in the practical setting, despite preferences (ER or Star) of the chief architects. Overall, Star models seem to have gained the most ground.

### 3. The Importance Of A Data Warehouse Business Case

Much has been written about the business case for a Data Warehouse. What goes in to a *good* business case? IT savings are ubiquitous in DWH business cases. The important point is to *not* limit this to 'pure' savings, but to connect to primary business processes as much as possible. As an example, faster turnaround cycles for list selections are fine (when quantified in hourly rates), but it is even better if the revenue from more customer acquisitions that follow from these selections can be tied in. Not only will the relation to revenue growth rather than savings make for a more balanced business case, more important is the intrinsic business buy-in that results from a direct connection to the company bottom line.

These days, changes in legislation (in particular Sarbanes-Oxley) play a major role in justifying business cases. This may be either through a higher company valuation for its transparent information gathering, or, less sleepless night for the CEO, which is of course priceless...

### 4. Why Do Data Warehouse Projects 'Never' Go Wrong?

Actually, Data Warehouse projects do sometimes fail. But, they fail so rarely, that it is actually very hard to believe... Especially after having talked to so many disgruntled end-users. And there are many ways a Data Warehouse project can go wrong. Delivering on time, data administration issues, and unavoidable data quality issues in feeding systems.

Corporate politics (see Tip 7) are probably the best explanation for this phenomenon of near 100% success rates on DWH projects. In my experience, the reason why a failure or 'semi-failure' can go unnoticed is either because senior management is not aware, or, let's say "unmotivated" to talk about mispending of company funds. As a result, not enough is learned. Maybe we as consultants have a stake in this as well, as this assures the industry plenty of ongoing business...

J

### 5. What Is Different About Warehousing Web Data?

Kimball & Merz (2000): "Although this clickstream data in many cases is raw and unvarnished, it has the potential of providing unprecedented detail about every gesture made by every human being using the Web medium". The subatomic nature of clickstream data poses unique challenges. There are fewer built in feedback mechanisms to ensure data quality, compared to other data streams. The relation between user mouse clicks and server log records is not as tight as in "traditional" transaction processing due to technical

issues like proxy servers and caching. Because of these differences, IT people need to adapt to the web process flow, rather than having the process adapt to IT needs as is common for most other DWH interfaces.

## 6. Which Data Should Be Loaded In The Data Warehouse?

The data that enter the DWH ultimately determine its place in the organization. A "let's load all data, to be safe"-attitude is a sure fire way to derail your DWH project. Choices as to what should and should not be included need to be made early on, to keep the project manageable.

After proven success of the delivered, deployed, and profitably exploited DWH, there *always* will be funding somewhere to include previously ignored interfaces. Given the anticipated lifecycle of the DWH, it makes perfect sense to consciously exclude certain sources. The choice as to what data to include needs to be driven by business considerations, and in particular reference to the company bottom line. If it can't be shown how data will be put to use profitably, they stay out! See also tip #3.

## 7. Data Warehousing & Company Politics

Data Warehouses have an impact on the company bottom line. Hence, they are likely candidates for turf battles, and are also at risk of becoming "small change" in budget allocation negotiations. None of these considerations benefit corporate long term goals. Managing a DWH project is hard enough as it is, and budget issues shouldn't make it any harder than it already is.

Because DWH investments are in the present and revenues lie in the future, it is even more important to secure funding through a sound business case and buy-in from the appropriate (high) management level. See also Tip #3.

Access to data means power, and talking about power is one of the greatest management taboos, still around. Sensitive as they are, even budgets are more readily discussed...

## 8. Data Warehouse Projects Traps

Some commonly recurring 'roadblocks' on the path to timely delivery of a Data Warehouse project:

- ETL processes have eaten up so much time (and *still* need "babysitters"), that little if any time is left to develop applications needed to *exploit* the DWH

- Some data are needed, but turn out not to be unavailable, or not in a timely fashion
- Maintenance required for tuning, indexing, and backup and recovery is severely underestimated
- Different ways of calculating the same phenomenon lead to different results, and nobody is able to conclusively explain the difference(s)
- The data that is loaded (and recombined) turn out to contain previously unknown inconsistencies in the source systems, the 'classic' data quality issues that trip DWH projects
- Meta data were lacking, and developers spend inordinate amounts of time finding out what a field really 'means'

### 9. DWH Hardware And Software Go Hand In Hand

In Data Warehousing, it is *not* about hardware, and *not* about software: it is about *the perfect integration of these two*. Those who begin their project from either end, will pay dearly for this mistake. Reasons are:

- in terms of price/performance, new, pre-integrated hardware-software combinations are taking the lead
- from a project management perspective, you never want to be caught between vendors when a proposed solution doesn't work as expected
- database tuning and indexing is very important and a hugely complex job, necessarily left to specialists (in-house trained)

### 10. Performance Is Key

Although I don't often find technology factors to be this important, in Data Warehouse acceptance, no other factor will be as important as *performance*. As size increases over time, this factor becomes even more important. There are three reasons for this:

1. performance has a huge impact on the development speed (initial load is always *very* time consuming), and hence the overall maturity of the DWH at delivery time
2. performance can make or break end-user acceptance, in particular the *predictability* of performance
3. performance has a tremendous impact on end user productivity, the ultimate driver of the business pay-off