baseline CONSULTING
Data Mastered. Value Unleashed.℠

Client INSIGHT

The Baseline Client Newsletter

Fourth Quarter 2007

In this issue...

Fall Line-up: Data Governance

I don't know about you, but I'm happy to see fall again. Sweaters are out of the mothballs, pumpkin spice lattes are back, and Halloween candy has hit the shelves. (But don't leave a bowl of those Dove Bar chocolate mini-squares anywhere in my vicinity. Please.)

One thing that's not changing with the leaves is the need for data governance. I'd mentioned in the last issue of INSIGHT that data governance was a hot topic, and over the summer it warmed up a few degrees. Here's what we've been discussing:

- While data governance should be business owned and managed, it's nevertheless largely IT-driven. This recalls Geoffrey Moore's claim in *Crossing the Chasm* that any IT initiative usually begins with a visionary and in the case of data governance, that visionary is more often than not in IT.
- Confusion persists about the differences between data governance and data management. Data governance is the high-level oversight of data policies and the decision-rights over enterprise information. Data management is the tactical execution of those policies. Ironically, most companies overestimate their data governance readiness while underestimating their data management capabilities.
- Data stewardship is where data governance and data management meet. The role of the data steward should be deliberately defined and carefully positioned.
- Companies shouldn't put data quality and data correction on the back-burner simply because they're planning on implementing data governance. Indeed, we've found that the more mature a data quality effort the higher the organizational readiness for data governance.
- Launching a data governance council as the first step usually backfires. Avoid the "Kickoff and Cold Cuts" approach, in which participants agree over a free lunch that managing data as an asset is a good idea. Much better for a core team of visionaries to actively design data governance before proselytizing its value.
- Data governance isn't a quick fix. It's not a finite project, a research study, or a center of excellence. In order for data governance to truly take hold, it needs to be systemic at the enterprise level.

When in doubt, ask yourself the following question: "If we take on data governance, what's our ability to execute?"

Speaking of which, if you're interested in executing data governance—and delivering authoritative master data—I have some exciting news. I'll be co-chairing TDWI's summit, Master Data Management Insight. The event takes place in Savannah, Georgia, on March 2-4, 2008. If you're interested in learning more, give me a call!

Jill Dyché
Baseline Partner

Eight Steps to Align Business and IT Around Information Delivery

A Baseline Approach

Baseline Consulting uses the *Eight Steps to Align Business and IT* diagram to help its clients understand how alignment can be formalized into planning business intelligence initiatives, application development projects and data integration programs. The "business" hemisphere reflects activities for which the sponsoring business function (e.g., customer support, marketing, finance) is primarily accountable, and the "IT" hemisphere reflects activities for which the IT and data functions are largely accountable.

Who's Responsible?

In the context of alignment, accountability does not mean that the "non-accountable" party is off-the-hook and uninvolved. Nor does it mean that responsibility is tossed from one to the other at Steps 3 and 6. Instead, primary accountability goes to the organization with the greatest expertise for key activities at the given point in the lifecycle.

IT has a "business" strategy, just like every other function in the business. IT's strategy typically includes things like an enterprise architecture, capacity plan, upgrade program, and timetables for extending, supporting and maintaining systems infrastructure required to sustain the strategies and tactics of all other functions within the organization. The IT strategy is what IT brings to the table as a stakeholder in a business initiative—it may put constraints on the business solution, it may imply an interim solution until additional infrastructure can be developed to support a more robust solution, or the business value may be so great that it drives modification of the IT strategy.

Even though IT brings special skill and knowledge to the planning effort, the business sponsor has the final say about the priorities of business needs, how the solution interfaces with business users, its value to the company and how that value will be measured—not IT. By the same token, business stakeholders don't get to dictate the solution approach and foundational technologies. IT has the expertise and perspective to make the best decisions for that part of the lifecycle. However, once development is done it's the business sponsor who is accountable for determining whether or not the solution meets the business requirements agreed upon before development started. Therefore, the business is also accountable for deciding when a project is complete.

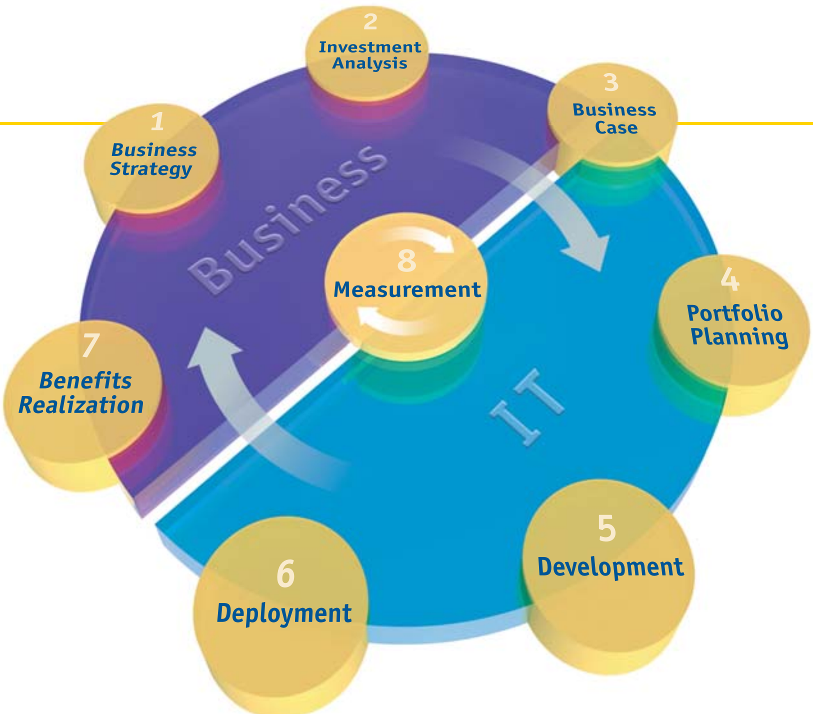
Aligning Priorities

There is an interesting co-responsibility that best practice companies have adopted: Portfolio planning and management. A portfolio is a collection of applications identified by and driven by the business. Each application is profiled just enough to identify dependencies among applications but not sufficiently to proxy for scoping and requirements. The portfolio is important in allocating scarce IT resources to development and enhancement projects based on their relative value across the enterprise.

Portfolio management is a formal repeatable process for prioritizing, adding, deleting and modifying applications. Responsibility for facilitating regular review and execution of the process may reside with IT, working alongside an enterprise data management function, and collaborating with Finance because of the relationship with budget, or with a cross-functional program manager—typically associated with an enterprise data warehouse or business intelligence program.

Regardless of who facilitates the process, portfolio management is cross-functional. The resulting priorities represent a weighted assessment of value, risk and cost to the business, including IT considerations. Portfolio management results in a list of business applications for development and deployment that the enterprise has agreed is from current highest to lowest priority. Applications lower on the list may change priority over time, but IT has marching orders for the first few efforts.

As a cross-functional activity, each stakeholder is accountable for accurately representing the value, benefits and risks of the applications identified by his department. The relevant stakeholders are held accountable for overstating value or failing to adequately anticipate business risks. IT is accountable for failing to adequately anticipate technical risks for both the project and the operational application. A best practice portfolio management process represses the ability to divert resources to the "squeaky wheel" or make decisions based on political intrigue. Similarly, the process allows sponsors on both business and IT sides to marginalize those who try to game the system for their own benefit, holding them accountable for articulating high-impact business requirements. Instead, best practice portfolio management leverages and enhances the communication between the business and IT that started with planning.



A Word about Measuring Value

In most models, measurement is represented once at the end of the cycle. When the metrics selected are relevant, they provide actual performance data for comparison with performance forecast during planning, and imply opportunities for continuous improvement and enhancement. The *Eight Steps* diagram, however, shows two measurement efforts at Steps 7 and 8. Separating measurement has allowed Baseline clients to explore investment and business plan performance as a topic separate from measuring the efficiency and effectiveness of the cycle itself. In practice, they're part of the same continuous cycle.

Realizing Benefits

Benefit realization focuses on measuring how well goals and objectives from the plan were met, and how closely actual performance matched the forecast used to justify investment. Underlying assumptions include:

- ❖ The sponsor selected metrics that are relevant and sufficient to profile the performance of the target business activity.
- ❖ Current benchmarks that will be compared with the post-implementation performance are relevant, complete and accurate.
- ❖ A reliable mechanism for gathering performance data is in place or is included in the project plan.

In most cases, measurement of benefits realization will continue in subsequent years and create performance history that can be used for future forecasts and modeling. Occasionally, there will be some short-term measurement used to demonstrate financial performance—return on investment or payback projections for example—or one-time gains. For example, a project that enables business process redesign might be measured for anticipated changes in employee productivity for a specified period after deployment; it won't be an ongoing business performance measurement.

Metrics usually vary based on the company's size, vertical industry and culture. Some are very specific—for instance, reduction in duplicate addresses—while others, like Return on Investment (ROI) are broad and often contextual. Some examples include:

- ❖ ROI: Often companies use pre-facto ROI estimates as a determining factor in project approval, and then use those estimates to measure success once initial implementation is complete.
- ❖ Consensus of need: Determining whether an application should be prioritized often depends on the breadth of organizational need. This translates into adoption levels once the application is deployed.
- ❖ Data availability: Identifying if the data is missing or difficult to obtain, irrespective of how important an application is. After deployment, this can translate into a data usage measurement.
- ❖ Strategic impact: Applications like churn analysis or customer dashboards can avail new information that translates into nothing less than higher customer satisfaction scores or increased share of wallet. Measuring such improvements can justify an application's funding long after the initial release.

Note that realization of benefits may measure objectives for the business, IT or both. Business metrics for a marketing intelligence application designed to increase conversion of customers to an up-sell offer might include "conversion ratio" and "number of cancellations." IT metrics for the same application would come from the service level agreement (SLA) negotiated with Marketing, and might include "conversion transaction data latency" and "untrapped bad addresses."

(continued on page 8)

What Does a Data Warehouse Architect Do, Anyway?

by Randall Anderson and Robert Stone

The term "architect" has many and varied interpretations, which is precisely why it's a bad idea to have more than one of them on your project. The exception is when the word is preceded by an area of expertise.

In her book, *e-Data: Turning Data into Information with Data Warehousing* (Addison Wesley, 2000), Baseline Partner Jill Dyché describes the role of the data warehouse architect:

> *The term "architect" has many and varied interpretations, which is precisely why it's a bad idea to have more than one of them on your project. The exception is when the word is preceded by an area of expertise. Your database architect, network architect, or application architect—if they are responsible for tangible deliverables—are all doing useful things. It's the architect without a modifier who can be dangerous, because the vagueness of the title circumscribes the title. A roving linebacker has no place on a data warehouse development team.*

So what does a data warehouse architect do, anyway?

Reconciling Business Value with Technical Implementations

The job of data warehouse architect combines breadth and depth of many different skill sets related to the definition, design, development, administration and maintenance of corporate data assets. This involves communication with business as well as technical stakeholders. All architecture processes must reconcile business value with technical implementation requirements. A successful data warehouse architect will have experience in business analysis, infrastructure design, data administration, database administration, application design, and business analytics.

The best data warehouse architects are those who are adept at engaging business users, while keeping their finger on the pulse of the BI market. A key role of a data warehouse architect is to provide structure and process to the data warehouse environment; thus, knowledge of products, trends, and emerging capabilities is key. Without a solid architectural framework, different groups in the company would invent—and likely reinvent—a multitude of designs, implementations, and platforms that are inefficient as a holistic system and result in increased costs and development times.

Key goals of the data warehouse architect are:

- ❖ Provide a future state map to help guide both business and IT in the development and growth of the data warehouse.
- ❖ Provide a roadmap that optimizes resources and increases the ease of use and the reuse of data as an asset.
- ❖ Provide standards, principles, and guidelines to help optimize use of the data.
- ❖ Provide consulting and project support to help implement the architecture.

Baseline Consulting often works with data warehouse architects on projects that require a high level of specialization in detailed design and implementation knowledge. During the diagnostic stage, data warehouse architects contribute to the technical assessment of their Business Intelligence (BI) environment. During the planning and design phases, architects recommend the deployment environment in terms of the optimal combination of technologies and platforms, as well as performing more specific tasks of operating system sizing, I/O performance estimation, and database design. During implementation, a data warehouse architect will work with the infrastructure team to define and assist in the implementation of complex technologies such as MDM hubs or software such as BI applications and data quality tools. The architect will also provide problem resolution support for ETL developers experiencing database error situations.

Always cognizant of technology evolution and improvement opportunities, the data warehouse architect maintains knowledge of IT architecture standards such as Service Oriented Architecture (SOA), popular database management systems (DBMS), ETL products, data profiling tools, OLAP products, and BI best practices. This broad knowledge is required to not only help define IT's future state map, but also to help provide a roadmap for the growth of the data warehouse to scale along with business innovation.

The Data Warehouse Build: From Idea through Sign-off and Beyond

In a typical data warehouse creation or major upgrade, the data warehouse architect will execute the following processes:

- *Feasibility.* The architect reviews the desired objectives, resources and timeframes to determine if the planned initiative or project can be successful. This requires a review of the business drivers as well as the current technical environment.
- *Assessment.* If the project is feasible, the architect reviews the planned environment to determine what process and infrastructure gaps exist and makes recommendations. This is coordinated with the roadmap to make sure the project fits into the long term growth plan agreed to by both business and IT.
- *Design.* The architect ratifies the project design in terms of architecture changes, application extensions, network configuration, security, and backup/recovery considerations.
- *Development.* The architect regularly reviews development progress to ensure that development standards are followed and documented for future projects, and that issues are tracked that might affect—or be affected by—the current IT infrastructure.
- *Deployment.* The architect reviews all issues related to changes from the architectural design perspective as well as any implementation specifics.
- *Sign-off.* The architect contributes to the final project review and verifies the implementation of the overall design.

Success Story: Solving Load Performance and Introducing Best Practice

Baseline recently worked with a client who was concerned about load performance on its data warehouse. Each day, they loaded and stored a complete copy of the current day account records and the previous year-to-date daily balance records The source system was not able to reflect transactions that adjusted a previous balance. In order to accurately reflect the current state of year-to-date daily balances, every record had to be reloaded. Additionally, the warehouse was required to store all source records in order to support an investigation if questions arose about when a particular balance may have changed.

Baseline and the data warehouse architect identified the following requirements:

- Historical daily balances can be adjusted.
  - The application system has no mechanism to reflect changed data.
  - The daily load is a complete history of daily account balances.
- The data warehouse users must be able to locate specific accounts for investigation.
  - The source feed is a daily flat file.
  - Flat files are archived each day.
- The frequency of format changes for the source feed is quarterly.

With requirements in hand, we recommended a single year-to-date image of daily account balances to improve load performance. This would be accomplished by matching the previous source feed with the new source feed and loading only the "deltas" or changes. The result: predictable load times and simplified queries.

We also recommended saving the source files as XML documents. Since the format of the source load files may change as often as quarterly, the XML document approach provided flexibility in searching for accounts further back than the last format change. A load date and time was added to the current date information contained in each record and placed in a searchable directory. An investigation into the balance history for an account would be a cross-query operation.

In this situation the data warehouse architect had to step away from the details of data warehouse operations and maintenance that the rest of his team grappled with. By thinking in broader terms, the architect implemented a solution that complied with the known requirements and stabilized a growing issue with load performance. He introduced the adoption of change data capture concepts—a data warehouse best practice that will be applied to similar situations in the future.

(continued on page 8)

To Build or Buy? How to Get a Data Model of Your Very Own

by Fernando Martinez-Campos

In the past dozen years or so, data modeling has gone in and out of fashion. Once the purview of "geeks"—data modelers were known to hole up in meeting rooms, constructing cryptic graphics on blackboards consisting of squares and arrows—data modeling was part of the IT development lifecycle. No one on the business side really understood what they were for, and after a while both models and their modelers seemed to disappear without a trace.

Data modeling is back in fashion. The rise of enterprise data warehousing, master data management, and customer relationship management initiatives have driven companies to seek the "single version of truth" about their customers, products, and financial assets, among other business areas. And as technology platforms ebb and flow, the data model becomes the de-facto touch point for consolidated information that reflects the business.

Some of our clients have data models that, after several years of shelf-ware status, are being dusted off, reviewed, and enhanced. Others are looking to vendors or consultants to build data models as natural and extensible project artifacts, while still others are hoping to leverage vendor data models built expressly for a single application or software product. As it has for software products themselves, the "build or buy?" question has come home to roost on behalf of the data model.

The Packaged Data Model

In the last two decades vendors and consulting companies have worked with clients to refine data architectures so that corporate data reflects business requirements. Such efforts usually involve the creation of what's known as a "packaged" data model. Such data models represent the vendor's cumulative experience of modeling many different companies' requirements and portraying the common data denominators used in that particular industry. The data models are enhanced and refined as new requirements emerge. Some models have an enterprise-wide focus, representing business needs across organizations and systems—for example, a financial services data model reflecting core banking functions. Other models represent specific application domains for an industry, such as the ARTS (Association of Retail Technology Standards) data model in retailing.

The complexity and specialized resources associated with data modeling have caused many companies to look to packaged data models that reflect the needs of their particular industries. A major benefit of these packaged data models is speed. They allow a company to get up and running faster by delivering a tangible data representation that can be used as a basis for further modeling work. Packaged data models also provide a blueprint of how data should be structured logically for a business, ensuring that business questions can be answered with the model, ultimately leading to a more robust database structure to support query and reporting needs. Most companies recognize that the packaged data model is a starting point, requiring additional work and customization to support specific business needs.

Caveat Emptor: Try Before You Buy

The main consideration when purchasing a packaged data model is to determine how well it fits the way you run your business. There will probably be data relationships in the model that do not reflect the way your company operates. There may be specific business processes or organizations that are not represented by the data model. In addition, your business processes may use the data in a different manner. Since most models will require some customization, the key point is: How much customization needs to be done—and could the customization effort require more work than starting from scratch?

Most data models look impressive at a glance: Lots of boxes representing entities, relationships, and attributes. Creating data models is a labor-intensive endeavor. But that doesn't mean that all of a packaged data model will be apt or even useful to your company. In addition to determining the level of customization required for the packaged data model, you and your colleagues will need to evaluate at a detail level what parts of the model can be used in your situation. The packaged data model will likely be incomplete in some ways and overkill in other ways.

Here are some of the key steps for evaluating packaged data models.

Step 1: Define a list of initial entities and map them.

Gather your existing business requirements and distill the "person, place, or thing" types that would wind up as major entities in the data model. If no business requirements exist, gather all your sources, both operational and analytical, that reflect entities in your business. You may not yet have defined or normalized your data model, but identifying core data groupings might help define high-level entities in the model.

- List your master data structures.

  In other words, take an inventory of the "nouns" in your organization, otherwise known as your reference lookup data: Product, Organization, Customer, Accounts, and Financial Hierarchies. These are all "things" that play a role in your business processes and can end up as entities in a data model.

- List your major transactional data structures.

  Sales, Shipping, Payment Events or any major transaction in your business will also be entities in the model.

- Map your list of entities to the packaged model list.

Now that you've identified your entities, map them to the packaged data model to see if (and where) they are in the model. Identify other "things" in your business that are not represented in the model. These gaps represent areas that the vendor may not have developed in its model. This exercise will also help you discern entities in the model that may not be relevant to your company's business—the "overkill" we mentioned earlier.

A brokerage company we work with evaluated a Financial Services enterprise data model in the hopes that many parts of the model could be used in house. After mapping the model entities to brokerage business entities, it was found that the only correlations were in the area of reference data: Organization, Customer, and Product. About 85 percent of the core brokerage entities representing Trades, Rule Sets, Pricing, and Portfolio Management where not represented in the model at all. Our client decided to design these in-house since the Financial Services enterprise data model did not truly represent the brokerage business. But they still found that having the basic structure of the data model was much better than starting with a blank slate.

Step 2: Develop resource estimates.

You'll now need to determine the level of effort that will be required in order to develop the model from scratch. If the model is built in-house, you can gauge how much time and what skills will be needed to complete it. This activity is often the deciding factor for most companies, since it forces the issue of availability of in-house skill sets.

Note that data modeling is a specialty and while many people can understand a data model, relatively few have the skills to develop (let alone customize) a robust business-focused model. A mistake many companies make is re-deploying business analysts or programmers into the data modeling role, the assumption being that if the staff member can talk to end users or write Java Script, she must be able to build a data model. Data modeling usually involves intensive training on conventions (e.g, so-called third normal form, 3NF, or relational modeling versus star-schema or dimensional modeling) and modeling methodologies. Simply "assigning" a data modeler is a decision that's fraught with risk and overinvestment. You might need to consider hiring a full-time data modeler from the outside, or engaging a consultant to initiate some knowledge transfer before going it alone.

Step 3: Estimate the effort necessary to enhance the model.

If a packaged model is indeed your starting point, what level of customization needs to be done? An important consideration is what part of the model will need to be revised or even "undone" in order to accurately reflect your company's data needs.

The following questions summarize the issues to consider when making the "build versus buy" decision:

- Do we have the in-house skills and resource availability to build our own custom model from scratch?
- How long will it take to enhance the packaged data model?
- If we acquire the packaged data model, what data areas may still need to be developed?
- Does our master data fit the way master data is used in the packaged data model?
- Does the model capture the major detailed attributes of our core transactions?
- Does the model support our existing business processes?

(continued on page 9)

Eight Steps to Align Business and IT Around Information Delivery

(continued from page 3)

Whatever is measured is agreed upon before solution design and development ever begins; the metrics and performance thresholds are specified during the requirements phase. The more communication between business and IT stakeholders through the planning process, the crisper the metrics, and the more likely there will be a meaningful correlation between business and IT measurements.

Measurement

Whereas benefits realization measures business value associated with a project or program, Step 8 focuses on the business value of the planning lifecycle itself. The purpose is to capture data for analysis that can help identify opportunities for continuous improvement.

The most universal example for this kind of measurement involves comparing project forecast with actual performance. What was the variance between the forecast and actuals? A variance greater than ten to fifteen percent in either direction is a common threshold for improvement; but only analysis can zero in on what is or isn't working well enough, which planning assumptions are or are not holding, and how they can they be changed to more accurately predict performance.

A less obvious measurement for continuous improvement gets at the quality of communication between business and IT. The better the communication and acceptance of responsibility during planning, the fewer surprises upon deployment, and the more expected the results. Just as with improving the accuracy of forecasting, identifying the best actions to take for improving communication, and therefore alignment, requires more than just performance data. It requires analysis and a commitment on the part of all stakeholders to make the process work better.

Adaptation Is Required

Keep in mind that the Baseline's *Eight Steps to Align Business and IT Around Information Delivery* is an archetype designed to promote discussion. It can be customized according to the company's sanctioned governance or steering committee processes. Indeed, the process in any given organization would differ depending on the presence or absence of a formal business planning and budgeting process, formal data management and governance infrastructure, project definition and management methodology (e.g., requirements, scoping and solution design precede development activities), and enterprise performance measurement standards.

If your company does not have a formal business planning process or an illustrated representation of it, Baseline's *Eight Steps* may help you get started. The simple division of accountability within the planning-development-deployment-measurement cycle—i.e., showing that the business is accountable for the areas where they have expertise (what) and IT is accountable for the areas where they have expertise (how)—can begin to build alignment into planning business initiatives and technology-enabled projects.

What Does a Data Warehouse Architect Do, Anyway?

(continued from page 5)

In Summary

The data warehouse architect can be characterized as the gatekeeper who ensures that the data warehouse is developed in a planned and systematic way. The architect has a key role in ensuring the business objectives and business requirements are being met currently and will be met in the future. Once an organization has clarified their information requirements for a business intelligence or master data management application, the data warehouse architect oversees all of the elements necessary to deliver the solution. If Information Architects provide the 'WHAT', the Data Warehouse Architect provides the 'HOW'.

About the Authors

Randall Anderson is a senior consultant with Baseline Consulting and an experienced IT and database architect. He provides data warehouse design and development services as well as organizational readiness assessments, and has worked with a diverse set of clients including OSF Healthcare, Quintiles, and RBC Centura.

Robert Stone is a managing consultant at Baseline Consulting, specializing in enterprise data warehouse and business intelligence application platforms. Robert has worked with a variety of clients across industries, specializing in large systems integration and architecture projects. He has worked with high-profile clients such as CheckFree, Microsoft, and BellSouth.

(continued from page 7)

Whether you expect to build or buy your data model, the people costs are usually the most expensive investment. Not only is the cost of the headcount an issue, but the risks of engaging the business community at your company—often for the first time—should not be underplayed. The purchase cost of the data model might turn out to be the cheapest investment you make!

Other Features to Consider

The model notwithstanding, there are other model "support" features that should also be considered when evaluating a data model—or deciding to go it alone from scratch.

Metadata. Packaged data models usually offer a generic definition of each entity and attribute. However, the precise meanings of data elements vary from company to company. To reduce ambiguity with generic definitions you need meanings that apply to your business. The data modeling effort is not truly complete until you have accurately and precisely defined data at the attribute level.

Physical Database Design. Most packaged data models are logical models built in third normal form to depict the business relationships of the entities and attributes. A physical database design leverages the logical model structure and optimizes it to support aggregations and calculations for reporting. Most physical database designs used for reporting are star schemas. You may need to develop your own physical database design depending on the package vendor's breadth of offerings, as well as your own reporting requirements. Some software application product suites come with built-in physical database designs (sometimes referred to erroneously as "data models," just to confuse their customers). These should be mapped back to the packaged data model to ensure completeness and accuracy of definitions.

Transformation Rules. Once the data modeling tasks are done, you still need to define business rules and cleansing requirements of the source data. Analysis needs to be done to map sources to target databases, profile the source system data, and migrate it from the source system. Source-to-target mapping and data transformation rules usually fall within the purview of the ETL tool you'll use to load the data from the source to the target system.

Forewarned is Forearmed

There is great promise with packaged data models as they are refined over time. Every year new models enter the market, and they can substantially jumpstart your data warehouse or master data management (MDM) efforts by providing a foundation for how data should be defined, connected, and structured on behalf of the enterprise. If you do your homework, the packaged data models may greatly improve both your analytical and operational implementations.

Nowadays, data modeling is not only hot, it's often a selling-point for vendors competing on features and functions. We've seen several MDM vendors argue lately about the need for a data model as part of the core functionality of a customer or product hub solution. Traditional data warehouse vendors tout their "customizable" industry models as part of their product suites. There are even individuals who sell their "off the shelf" models to vendors and client companies for a tidy sum.

Regardless of whether a data model comes from a vendor or is built in-house, you should understand how it will be used. As a business analysis tool? As a mechanism for gathering and validating user requirements? As the foundation for a physical database schema that will represent a BI database? Perhaps all of the above? In any case, understanding the role of the data model will take you a long way in knowing how to build it, where to buy it, and how to leverage it to its fullest once you're ready.

About the Author

Fernando Martinez-Campos is a Senior Consultant with Baseline Consulting, specializing in data integration and management. He has written for DM Review, spoken at TDWI and DAMA events, and has worked at several pharmaceutical companies to help them manage their master data.

Trends in Data Governance

A Baseline QuickRead

In this QuickRead, Baseline Consulting defines Data Governance and provides insight into launching a successful governance program within your organization.

Q What is data governance?

A Baseline defines data governance as a high-level decision-rights and oversight process. It is the organizing framework for establishing strategy, objectives, and policies for corporate data. The important thing is to distinguish between the "what, who, how, and why."

Data governance answers four questions:

- ❖ What decisions need to be made?
- ❖ Who will make those decisions?
- ❖ How will the decisions be made?
- ❖ How will the decisions be monitored?

Baseline takes great care to keep the conversation about data governance at the leadership and asset management level. In so doing, we position data as a corporate asset with executives—it has value, the value can be measured, the asset helps the company achieve its strategic objectives, and the asset requires specialized skills for its continued use.

Data as an asset answers the "why govern" question. The goal of data governance is to enable operational efficiency, scalability, and reliability. Through standardization and integration of data, companies promote data reuse across systems and applications and data sharing among business functions and processes. The end result serves the corporation with better quality decisions, business innovation, market responsiveness, and flexibility.

When viewed in the context of "asset decision-making and oversight", data governance becomes clearly separated from the more tactical data management function which is about execution—implementing data governance policies, data administration, data acceptance criteria, error detection, and data cleansing/correction.

Q What is the first step in launching a successful data governance initiative?

A Baseline's first and foremost advice is: Data governance must be deliberately designed before it's launched. Otherwise you run the risk of getting a bunch of decision makers in a room to hash it out before you really know what "it" is.

Baseline recommends a Data Governance Design Team and uses a simple Data Governance Framework to facilitate the design stage through the following steps:

- ❖ Assess the organization's readiness—do you need what data governance does.
- ❖ Define guiding principles—basic doctrines or rules of conduct.
- ❖ Identify decision-making bodies—cross-functional influencers and deciders.
- ❖ Define decision areas and decision rights—go/no go responsibilities.
- ❖ Identify governance mechanisms—the processes, tools, and metrics.

The framework can be used to educate, generate discussion, allow debate, drive negotiation, and lead to final agreement on how the governance process will operate. The very act of "thinking through" the framework components will lead the design team to identify issues, explore alternatives, test scenarios, and jointly arrive at solutions. This iterative diverge/converge thought process develops ownership and commitment to a governance plan tailored to meet the unique needs of the company and corporate culture.

Companies with successful data governance have usually begun as small core teams focused on an individual application or system and evolved over time to enterprise-level Data Governance Councils focused on enterprise information policies.

Trends in Business Intelligence

A Baseline QuickRead

In this QuickRead, Baseline Consulting looks at current trends in Business Intelligence (BI) and explores some of the most often-posed questions about maturity and BI.

Q What does Baseline consider to be the top three attributes of a mature BI program?

A Several factors determine BI maturity, but ultimate success boils down to the degree of sophistication with which a company is able to leverage analytics to effectively make decisions and take action to run the business. While BI maturity varies from industry to industry, and even department to department within a company, Baseline calls out three key attributes of a mature BI program:

- The data warehouse contains all *relevant* data—as determined through bona fide business requirements gathering—to meet users' metrics for timeliness, usefulness, quality, and relevance to support business analysis. However, it's not just about replicating volumes of detailed data. In a mature BI environment, the data is modeled, managed, and integrated. Users do not have to spend lots of time searching for the data or validating its meaning, lineage, or accuracy.

- End-users are self-sufficient. They have data access "on demand." This means that business users can access metadata, understand the meaning and differences of the data, and launch reports, run queries, and perform complex analysis without involving technical staff. The focus is on the end users and how to support their vastly different analytic needs, skills, toolsets, and data access rights. The BI program enables self-sufficiency with data training, a user support desk, and ongoing communications relating to data quality and business analytics improvements.

- Management understands the value of data and has established a culture of "fact-based" decision-making. The data is readily accessible to answer emerging business questions and fulfill new business needs in short order. BI is considered fundamental to strategic business initiatives, as well as both analytical and operational decision-making. It is budgeted and staffed accordingly.

Q What are the risks of not having a prioritization process? How can companies implement this BI best practices?

A Without a defined and sanctioned prioritization process, BI programs fall prey to many maturity inhibitors—business and IT never get on the same page, conflicting interests abound, and ultimately user adoption suffers.

Without a prioritization process, "squeaky wheels" get most of IT's attention and development projects enter never-ending cycles. Single-purpose data get loaded to the data warehouse—of interest to one department and designed for one specific need. Business and IT ask: Who decides what to work on next? Are we spending money on the right things? Should we "pull the plug"? How do we get executive level support? How do we anticipate changing needs and the impact to the BI development schedule?

By imposing rigor in project selection and accountability in project delivery, prioritization raises the overall business value of BI to the enterprise. It provides program and project visibility across the company, and increases data sharing across many different users. Prioritization brings order to the way development projects are sequenced based on dependencies, achievability, cross-functional impact, and business value.

Baseline's BI Application Portfolio℠ service sets the industry best practice standard as one technique for driving BI maturity. It serves as the de-facto development pipeline for the company's BI program. The BI Application Portfolio℠ prepares a development roadmap and leaves you with the methods to sustain ongoing prioritization. It provides a foundation to scope each project deliverable in 90-day increments, calculate ROI, and gather business requirements. Each application in the portfolio contains core business questions and success metrics around the underlying business problems and data needed for a solution.

Upcoming Events

Meet the Baseline Experts

Visit www.baseline-consulting.com for more details.

❖ November 16, 2007. CDI Institute and Source Media CDI-MDM Summit, New York City.
Building the Agile MDM Team with Jill Dyché and Evan Levy.

❖ November 20, 2007. DataFlux Web Seminar.
Zero to Five: A Stepwise Progression for MDM and Data Governance with Jill Dyché and Evan Levy.

❖ November 29, 2007. Initiate Web Seminar.
Data Modeling & MDM: The Facts and the Fiction with Evan Levy.

❖ December 4-5, 2007. Wilshire Data Governance Conference, Orlando.
Bulletproof MDM with Evan Levy.
Data Governance: From Idea to Execution with Jill Dyché.

❖ **Coming March 2-4, 2008:**
TDWI's Master Data Management Insight Summit, Savannah, Georgia.
Baseline Partner Jill Dyché co-chairs this new invitation-only summit designed for senior business and technology leaders to accelerate their knowledge of MDM and tap into the expertise of niche market vendors in the industry.

Learning Center

Baseline Publications

For your copy, please visit us on the Web at:
www.baseline-consulting.com/Featured_WhitePapers.html

❖ The Baseline on MDM: Five Levels of Maturity for Master Data Management by Jill Dyché and Evan Levy
Baseline Consulting presents an MDM taxonomy that separates and describes discrete capabilities, helping you understand your company's "as is" environment to help you accelerate toward your "to be" objectives for master data.

❖ BI Adoption Evolves: The Baseline Pyramid Redux by Jill Dyché
When it comes to business intelligence, there's no such thing as "one size fits all." The key to deploying BI successfully is to understand its evolution, and to meet end-users where they are.

❖ A Data Governance Manifesto: Designing and Deploying Sustainable Data Governance by Jill Dyché
Data governance is a vital component to any master data management initiative, as it defines who owns the data, who establishes policies and who is the decision-making authority when it comes to an organization's various data assets.

❖ Getting a Grip on Data Governance and Data Management by Linda McHugh
Rome wasn't built in a day, and neither is a well-managed data environment, but the examples and exercises in this white paper can help you get a grip on effective management and exploitation of the enterprise data asset.

baseline
CONSULTING
Data Mastered. Value Unleashed.℠

15300 Ventura Blvd., Suite 523
Sherman Oaks, CA 91403

1-818-906-7638

www.baseline-consulting.com

Baseline Consulting is a management and technology consulting firm specializing in data integration and business analytic services to help companies enhance the value of enterprise data and improve the performance of their business. Baseline's proven, structured approaches uniquely position us to help clients achieve self-sufficiency in designing, delivering, and managing data as a corporate asset.

BCI-2007-03