

A Comparison of Single versus Multi-Vendor Approaches for Tackling Business Intelligence

Business Problem

The Business Intelligence (BI) marketplace is littered with simple “solutions” to complex problems. The complex problems faced by most organizations are: 1) assembling information; 2) ensuring that the information is consistent and accurate; and 3) providing that information in a consumable form. Since 1) and 2) are difficult, most “solution providers” focus on 3).

Today, the most common “solution provider” story is one of self-service reporting and analytics tools that do not require technical skills. In many cases, the vendor has no offering to address organizing information or ensuring that the quality of information sources is good, as very few of them have a comprehensive technology offering. Nonetheless, this self-service story is appealing to uninformed buyers for several reasons:

- 1) People are familiar with Excel, and front-end BI tools represent that they are as easier than Excel
- 2) Personal BI tools support individual desires to maintain control of the underlying data (there is subtle resistance to changing the source of information from personal storehouses of data in Excel to a common set of “company” information)
- 3) The flashy interface of most BI tools creates the illusion that information can be assembled and provided with reasonable quality at limited expense and effort

The above are exacerbated by the longing in most companies by the technical staff to provide a “quick win” that brings value to the business as fast as possible. Ultimately, the dynamic of the vendor community communicating a story that customers want to believe, combined with the customer desire for an expedient response to internal business requests leads to a poor result: acquisition of solutions that have almost no chance of real customer success and increased spending for the business decision maker without adequate demonstrated return.

Introduction

The success of any BI project requires the use of a tool set that facilitates the creation of short-term tactical solutions (i.e., providing information) while building the framework for a comprehensive master data architecture (i.e., assembling information and ensuring adequate data quality.) To be cost effective, work performed during the initial phases must be efficiently leveraged by the master data services to avoid incurring significant reimplementation costs. In addition, data discrepancies introduced by this reimplementation can impact users' confidence in the reported data which may threaten the success of

the entire BI project. Information Builders (IBI) provides an integrated, wide-ranging BI product suite based on a rich, ubiquitous, metadata layer designed to facilitate an evolutionary BI lifecycle. IBI's product architecture is significantly different from competing approaches which use a suite of separately designed products from multiple vendors that require identical software logic to exist in multiple places.

The Evolutionary BI approach

After purchasing BI product suite, few organizations will accept the idea that they must wait a long period of time before the benefits of their sizable expenditure can be demonstrated. As a result, at the beginning of any BI project it is most often necessary to show some quick wins in order to prove that the recent investment in BI technology can deliver on its promise. This involves constructing reports from several sources without completely defining an overall reporting architecture. This is actually a good exercise, since it helps BI engineers gain insight into the organization's data and reporting needs before developing a reporting model based on limited knowledge. In this paper we refer to this as the *Tactical Phase*.

What follows the tactical phase is the *Strategic Phase*, which may begin as soon as enough information is captured to create a preliminary reporting model. During this phase, information is often moved to a reporting structure such as a star or snowflake schema, or it may also involve creating relational models in order to support specific applications. For the strategic phase to be effective, it must efficiently leverage all preceding work and be flexible enough to accommodate new dimensions and attributes as they become available. Note that both the tactical and strategic phases may coexist forever, the former being used as a way to quickly respond to users' demands while providing a pathway forward.

The purpose of this paper is to identify the differences between a multi-vendor approach and the IBI single vendor approach.

BI Implementation using Components from Multiple Vendors

Consider what happens after a company buys a BI reporting tool from one vendor and an ETL (Extract Load Transform) or data quality tool from another. The execution of the tactical phase frequently means designing a set of reports or dashboards that require disparate information existing across a business to be synchronized and displayed. In order to accomplish this in the short term, most BI reporting products provide the ability to access multiple heterogeneous data sources and create reports as long as the information has appropriate keys to associate the data and, the data is of the same grain. However, this is rarely the case in most organizations. As companies expand, merge, and globalize, corporate technology continues to become more diverse. Most organizations contain a large, complex set of diverse applications supported by databases that are seldom integrated and are not necessarily modeled to facilitate BI reporting. Also, many data sources may come from third party vendors who supply data from outside the organization. Still the demand for quick results exists and must be satisfied.

A common approach used to satisfy this demand is to create logic inside the BI reporting application that normalizes part of the data so that reports can be generated. This may involve reformatting fields, creating

externally maintained correlation tables, or embedding custom SQL queries inside the application. If the initial dashboards are well received the thirst for more information will inevitably grow, resulting in more dashboards and much more custom logic. As time goes on, the demand will become difficult to meet.

At some point, the need for a more comprehensive database of correlated information will become obvious to the organization, i.e., the strategic phase starts. At this point, the ETL tool is introduced to create a separate database, or warehouse of information so that reporting can be done more flexibly. This involves taking the embedded code that was used to correlate the various data sources out of the existing dashboards and re-implementing this logic in the ETL tool.

First and foremost, the problem with this approach is that it represents wasted effort. Given that good BI implementers are expensive and difficult to find, their time must be allocated efficiently. In addition, this approach often results in multiple versions of truth leading to questions about the validity of the warehouse and its overall usefulness. No matter how careful the implementers are, they will always find it difficult to replicate reports that have been created using different data pathways through different technologies. Another problem is that validation now has to be done between three sources: the application reporting, the initial tactical dashboard implementations, and the new dashboard implementations. This is a time intensive process that is difficult to maintain by even the most competent professionals. And, if resources are limited the warehouse may never gain traction in the organization and the initial dashboards employing the ad hoc approach will become the de facto BI standard. This will always limit the usefulness of the BI implementation and never allow it to realize its full potential.

Now consider what happens when another data cleansing software component becomes required, which may be from yet another vendor. When you cleanse data you may end up changing its format or actual values. This effort may affect reports that are already in existence causing them to not function properly, especially if they are based on data that has been manipulated to perform inter-database correlation and normalization. And, since the reporting tool is from a separate vendor than the data quality tool, there is no easy way to know which reports will be affected. Also, if the reports and dashboards do not automatically absorb the cleansed data the issue of multiple versions of truth again arises.

The problems introduced by using the multi vendor approach are not the result of poor process, they are the direct result of trying to use tools from disparate vendors to solve the problem of disparate data.

IBI and the Evolution to Master Data

Now, consider the architectural differences between the IBI approach and the multi-vendor method. IBI Reporting (Webfocus/Infoassist), ETL(Data Migrater), Data cleansing and Data Maintenance (DQM) are integrated through a metadata layer, the capabilities of which are considerable. IBI provides adapters for most every data source, with the metadata layer acting as a common syntactical abstraction layer. It allows the developer to join tables, add data fields based on other fields, and embed sub-queries. When encountered with the initial demand of creating reports from various sources, much of the data manipulation can be done in the meta-data instead of the front end BI application itself

(Webfocus/Infoassist). The impact of using this approach on the effectiveness of the BI development process are profound.

Consider what the previous scenario would look like if the IBI toolset were used from the start. The tactical phase of delivering dashboards and reports would be accomplished using existing data sources accessed through Webfocus/Infoassist. Data correlation would be handled, as much as possible, utilizing the capabilities of the IBI metadata layer. The iterative process of creating reports off existing data, validating and refining the reports, will help define where the necessary data components are located, and what kind of data manipulation will be required to effectively use the data. As this phase begins to deliver useful results, as we discussed before, the need naturally arises for a more comprehensive solution.

When confronted with the need for more comprehensive master data and perhaps the creation of data warehouses, the Datamigrator ETL tool would reuse all of the work done in the metadata layer during the tactical phase to create any variety of star schemas or relational databases targeted at a particular analysis or applications. This would eliminate reimplementation problems that would be introduced if code were to be migrated from the BI reporting tool to the ETL process. As a result, development will be faster and more accurate, validation easier and the adoption of the secondary database(s) more likely.

Now consider the issue of data cleansing. Data cleansing using IWAY's Data Quality Services is considerably different than using a multi-vendor approach. First of all, since the Webfocus/Infoassist products are built on metadata, developers can use the IBI impact analyzer determine which applications will be affected by any change in the metadata or its values due to the cleansing effort. Therefore, data cleansing will provide a way forward to better reporting, not a step backward caused by adversely affecting existing applications. In addition, the cleansing process can use any enhancements that have been made to the metadata during the tactical phase as a starting point. These features help to facilitate the adoption of a comprehensive data validation and quality management solution so critical to the overall success of the BI project.

Example

First we will consider the multi-vendor approach. A company has customer information housed on two separate databases from two different vendors. Each database contains similar information, however it is structured differently. The company's management wants a report listing all customers, the last time they were invoiced, and the average annual revenue received from each. From this report, management plans to prioritize their future invoicing. Using the tactical approach, a developer examines the two data sources and finds two tables that can be used to link the systems together.

Table 1 - From System-1

CustomerID	FirstName	MiddleName	LastName	CompanyName	CustomerType
1	Orlando	N.	Gee	A Bike Store	445
2	Keith	NULL	Harris	Progressive Sports	556
3	Donna	F.	Carreras	Advanced Bike Components	667
4	Janet	M.	Gates	Modular Cycle Systems	667

Table 2 - From System-2

CustomerKey	CreationDate	CustTypeKey	CustomerName	LastModifiedDate
1	7/9/2000	8	Herriot Computer Products	7/9/2000
2	7/9/2000	4	Rand Innovation Co.	7/9/2000
3	7/9/2000	8	Alde Inc.	7/9/2000
4	7/9/2000	2	Davis Garage Doors	7/9/2000
5	7/9/2000	4	Blackman Co.	7/9/2000
6	7/9/2000	6	SMC, Inc.	7/9/2000
7	7/9/2000	2	Micro Consultants Inc.	7/9/2000

Using the capabilities of the reporting tool the developer writes sub-queries to aggregate data from invoicing records (stored in other tables) to find the last time each customer was invoiced and the average revenue per year that has been received. Then each separate report is consolidated into a single report by appending an ID to the Customerkey and CustomerID fields and matching, where possible the columns from the two tables. In addition, any business logic required to match similar customer types or invoice types would be embedded in the code. Note that the code used to perform the consolidation and aggregation is housed inside the reporting tool and is specific to that tool.

As time goes on and more reports similar to this one continue to be created.

At some point, greater flexibility is required creating the need for a relational or dimensional database/datamart to be created so that the management team can operate in more of a self serve manner. At this time an ETL developer will need to access to same systems and create a model using the same business logic as was used to create the individual reports. However, the ETL developer cannot reuse anything that has been done to date because of the proprietary nature of the products. This is the where things can become very challenging, not only will the ETL developer not be able to reuse exiting code, they will inevitably have great difficulty recreating the business logic so that the existing tactical reports match the newer self-serve reports. This is where deliverables begin to slip and a lack of confidence among users of the system starts to form.

Now consider how differently things would proceed using the IBI toolset. Instead of creating the custom business logic inside of the reporting tool, the IBI methodology would use the metadata layer to implement this logic. Then reports would be created using Webfocus/Infoassist based on the metadata. When the need for self serve became necessary, the DataMigrator tool would be used to create a relational/dimensional using the same metadata as the tactical reports. No reinvention of business logic or transformation logic would be necessary and validation would be much simpler. This makes the transition between phases smoother, less expensive, and less risky.

Conclusion

The synchronization of data across disparate sources to allow access to enterprise-wide information is a complex and often expensive exercise for a corporation to undertake. Given this fact, there is no reason to further complicate matters by using a multi-vendor approach. Disparate vendors only add to the problem of disparate data. Therefore, where possible the use of BI products that are integrated and provide a pathway for an evolution from tactical reporting to Master Data Management increases the chances of success.

In closing, organizing, aggregating, and presenting information is hard work and requires specialized skills. Unfortunately, the need for a business solution is typically cloaked in an expectation that, once the technology and staff are identified and budget is agreed upon, the solutions will be forthcoming. Because there is so much pressure to deliver, the circumstances for the provider of the technology and the buyer of the “solution” tend to create structural challenges in realizing timely benefits from the investment. Moreover, organizing and aggregation often take a secondary role to presentation, which leads to a presentation of disorganized and erroneous data. Further, a technologically piecemeal approach (i.e., technologies from different providers) adds complexity to an already challenged situation. Therefore, the most important elements to an effective BI approach are: 1) ensuring the primary vendor has a comprehensive view of the BI problem, and understands the long term and iterative nature of the sales cycle; 2) maintaining continuity of the customer team constructing the solution. If these two items are managed well, the likelihood of a lengthy and successful endeavor is greatly enhanced.