



white paper

The DBI Difference

A streamlined approach to business architecture modeling helps align operational integration challenges with technology solutions.

DBI Consulting, Inc.

6200 Shingle Creek Parkway, Suite 400

Minneapolis, Minnesota 55430

Office: 763 561-4990

Fax: 763 561-9022

www.dbiconsulting.com



Symptoms or root causes?

As an executive, do you feel that:

- You are trailing competitors in product costs or time-to-market?
- You lack the ability to detect and prevent fraud and/or corporate malfeasance?
- You don't have access to the information you need to make critical business decisions?
- Your business agility is hampered by your technology infrastructure?
- You're not getting the expected return on your IT investment?

If you answered “yes” to any of these questions, you may be blaming data or information integration. Perhaps your operations can't readily share information across their boundaries; or maybe they can't roll up disparate information into the enterprise-level views you need to make the right strategic and tactical decisions.

So what do you do? Many executives look to technology to solve these challenges. Their reasoning: if the problem is information, then information technology must hold the answer.

In reality, data integration is often only the symptom of an underlying problem – inadequate operational integration. Unfortunately, by addressing the symptoms (we need better and more integrated information) and not the underlying problem (we need to better integrate our operations), you risk making information problems worse.

How? By cobbling together increasingly elaborate technology fixes that result in brittle, inflexible IT infrastructures. Many businesses have built such complex infrastructures to integrate data and systems that it's impossible to find any one individual who even understands them. Rather than solve information problems, such infrastructures hold the business hostage to technology imperatives. Soon technology begins to drive business processes, not the other way around. Who's in charge? The answer is often IT. Do operational integration problems get better? The answer is often no.

In most cases, there's a better way of looking at the underlying operational integration problem – one that isn't exclusively focused on technology. It's almost a cliché that “technology must be aligned with the business.” Yet the traditional systems- or technology-focused approach typically

has little to offer in terms of accomplishing this alignment. In contrast, DBI's business architecture approach addresses business problems more directly and provides the methodology for deriving technology solutions from your existing business model – instead of forcing you to change the way you do business to fit the mandates of a particular technology.

Consulting flavors

Consultants often come in one of two flavors. The first and predominate flavor is technology-focused consultants who examine IT infrastructures to determine areas for improving the flow of data from one system to another. This type of analysis is often done with only cursory attention paid to individual business units and the business processes IT systems support.

Instead, these approaches tend to focus on the abstract benefits technology brings and the abstract business processes they are meant to improve. If these benefits are compelling enough, individual business units will naturally benefit. If business process changes are required to conform to the demands of new technology, the price is considered acceptable in light of the expected benefits.

If this first flavor of consulting can be described as a technology-comes-first approach, the second flavor of consulting focuses only on business processes and tends to place technology in a secondary position. Consultants in this camp may deploy a variety of methodologies to “model” the business at a high level, but such work often ends up as “shelfware,” since it rarely translates business requirements into workable technology solutions.

Business architecture falls between the two approaches. As its name implies, business architecture focuses on understanding the building blocks or components of a business and how they fit together to create its value chain. Both technology and business modeling are part of the approach, but unlike the two contrasting approaches described above, business architecture attempts to span the technology and business domains rather than focus on one or the other. In many cases, it's a more efficient and effective way for developing technology solutions, especially operational integration solutions, that effectively solve business problems.

Business architecture is “architectural” in that it identifies at every level in a business's value chain:

- What are the important components?
- What services do components provide?
- How are components combined to provide higher-level value?

At the same time, business architecture is business-focused in that its solutions are based on an architectural understanding of the business problem(s) they seek to address.

Figure 1 shows the two spaces (or domains) in which most consultants operate. The bottom oval represents the domain of those who take a systems- or technology-driven approach to business problems. The top oval is the business domain, where business processes are often modeled independent of the technology on which they depend.

The middle oval in Figure 2 encapsulates the business architecture approach, which attacks business problems from the inside out rather than from the top down or bottom up. Instead of focusing almost exclusively on technology or high-level business processes, business architecture approaches operational integration challenges by modeling the business problem and the solution goal in tandem.

It does so by applying some business architecture tools (see below) to structure the business challenge in such a way that a potential solution can be envisioned earlier in the process. However, instead of attempting to model all the business processes of a large enterprise – an exhaustive and time-intensive endeavor – the business architecture approach is more analogous to fast-track design and construction in the building industry: The basic architecture of the business challenge is established (the blueprint), a solution architecture is sketched out (the technology required) and work begins. A few case studies from DBI’s own work experience illustrate the difference this approach can make.

Business-Driven Architecture Methodology

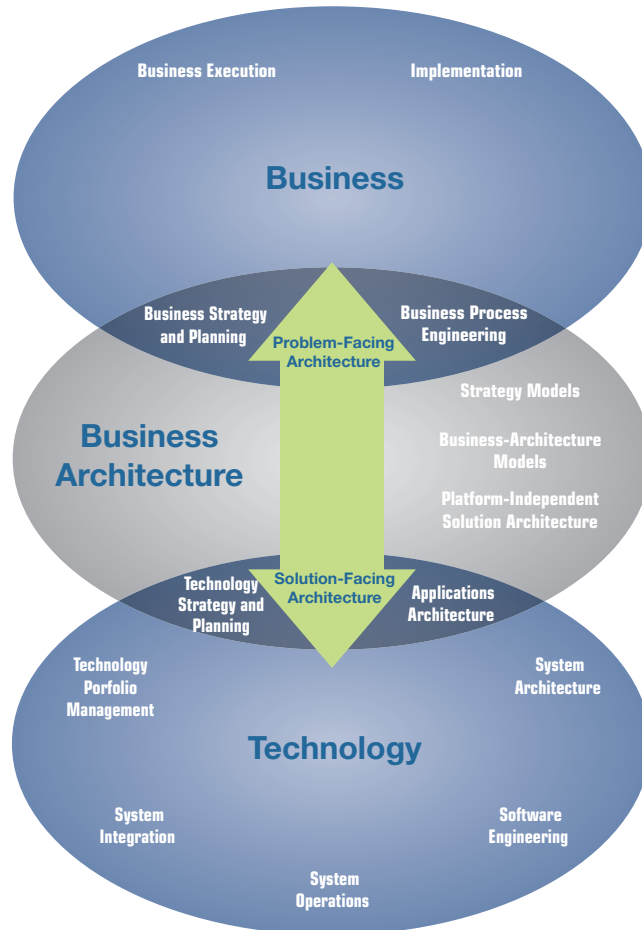


Figure 1: Consulting domains

Example 1: Simplifying data sharing using a federated information systems model

A leading financial services company had struggled for years with what it perceived as a data integration problem. Business units within the company couldn’t get data from other business units when they needed it and had difficulty interpreting it once they did. As the company grew, the number of these system connections expanded exponentially. As a result, a simple change to a database table in one system could produce unpredictable problems that would ripple across the organization. In one instance where several existing systems were to be replaced with a vendor package, it took 120 days just to find and test existing data dependencies in these systems.

Decision-makers in the company concluded that the solution to this data integration problem would invariably involve more technology. DBI's business architecture analysis came to a somewhat different conclusion. DBI discovered that individual business units had a great deal of operational autonomy, including implementing independent business processes, customized IT systems and proprietary vendor software solutions. Within this context, data issues tended to surface when information flowed across business unit boundaries.

This operational autonomy was one of the strengths of the overall business, however, and this wasn't about to change. Instead, DBI proposed that a federated information system could solve the data integration challenge and have relatively little impact on how the autonomous business units operated. In this model, heterogeneous, loosely coupled IT systems communicate via agreed-upon standards. Different units in a business retain their data formats and technology platforms and continue to operate semi-autonomously.

To make the model work for this business, DBI needed to identify the business occurrences, or events, which created or modified important information that then needed to be made available to the enterprise. The company then had to establish agreed-upon standards to communicate this information automatically, completely and unambiguously – not a trivial task, but far easier than adopting another layer of technology and forcing extensive business process changes.

DBI focused on determining which business processes generated business events and defining what information about the events was needed. This approach eliminated a lot of unnecessary detail and instead focused on two things: which business events the process generated (produced), and what information the process itself needed (consumed) in order to execute properly. Once this was done, it was fairly easy to link the information systems, thus solving the information-sharing problems of the past.

Example 2: Enabling product standardization and flexibility with mass customization

In a second example, a leading issuer of mortgage-backed securities had problems getting new product offerings to market quickly, resulting in significant competitive disadvantage. Product terminology and structure were not standardized among business units or the systems that supported them, making it difficult to introduce or even to evaluate the impact of new or modified product offerings on existing business processes and systems. Even slight variations on existing offerings were implemented as new stand-alone products, requiring many people to get involved and myriad changes to be made to IT systems. As a result, responding to the market was slow and onerous, with “new product” introduction routinely taking six to nine months.

In addition, when traders purchased mortgage loans for securitization, trade-time decisions in response to customer requests were captured ad-hoc using spreadsheets and email. This was necessary because existing systems couldn't support variations from standard product definitions. As a result, downstream business units charged with validating purchased loans routinely had to manually incorporate trade-time information into their business processes.

DBI's business architecture analysis showed that the business problem was the lack of product standardization and flexibility. DBI's business architecture solution involved determining the commonality across product offerings while allowing for uniqueness. The solution created a mass-customization environment, in which a stable set of product features and rules could be combined in unique ways to yield different, even highly dissimilar, product offerings.

In short, DBI's business architecture approach restructured the way that this company looked at the mechanics of defining, customizing and selling products. This business space architecture led directly to a high-level component blueprint for a mass-customization solution. Business units could operate in much the same way they had done previously, thus preserving their architectural strengths and their overall contribution to the company's value chain. At the same time, operational integration was improved to such an extent that time-to-market problems with new products were largely eliminated.

Example 3: A technology-agnostic perspective

An internal accounting group responsible for a large portfolio of a financial services company relied on highly manual, time-intensive processes for normal monthly accounting. They used a host of Excel spreadsheets and Access databases, resulting in cumbersome and error-prone processes, relics from the days when the company was much smaller.

Month-end accounting activity routinely took six days to complete, forcing an early cutoff date for data input. This resulted in pre-close estimates, which often varied widely from actual closing numbers, necessitating yet more work to investigate and explain the differences. Large accounting errors were routinely found after the corporate accounting system had been updated. This lack of timely and accurate financial information greatly hampered tactical and strategic decision-making.

Various attempts to improve the situation failed for lack of a clear goal. One effort led to very detailed documentation of business processes, data flows, systems, system processes, documents and data stores. This documentation gathered dust, however, because there was no clear connection between the analysis and how it would be used for decision-making.

DBI's business architecture approach revealed that employees in accounting and other areas were unable to conceptually separate business processes (the "what") from the procedures (the "how") that implemented them. From this discovery, DBI concluded that a traditional current-state/future-state analysis would be inadequate. Instead, DBI developed "true-state" business process models that illuminated the actual business processes in place, irrespective of technology. These models showed business processes, their sequencing and the information they produced and consumed, unencumbered by what was or was not "possible" to implement.

DBI's business architecture approach provided a technology-agnostic view of the business requirements, allowing the company to align business process execution and ownership, to drive information requirements from actual internal consumers' needs, and to make "exceptional" business processes "normal." The result was a vastly streamlined accounting process supported by a fully automated solution.

The DBI difference

As these examples show, business challenges are often interpreted by executives as information-sharing and data-integrity problems. However, DBI's business architecture modeling approach often shows these to be symptoms of underlying operational integration issues.

The results can be a more complete understanding of operational integration issues and a streamlined roadmap for operational improvement – streamlined because analysis done in the business problem space uses the same tools and parallels as analysis done in the technology domain. The benefits of DBI's business architecture approach can include:

- **Increased business agility.** You can respond to competitive pressures or business climate change without having to "re-invent" your IT systems and reengineer business processes every time you need to do business differently.



- **Better decision-making.** Operational integration implies the integration of information systems that drive the different operations in your business. When these information systems are integrated, information flows easily between them without manual or middleware workarounds that delay getting operational information in front of decision-makers and make the reliability or validity of information suspect.

IT solutions aligned with business problems. By addressing both business requirements and possible technology solutions in parallel rather than as separate steps, DBI's business architecture approach can help ensure a better fit between business needs and technology.

Increasingly today, consultants emphasize technology solutions to business problems. This frequently amounts to attacking the wrong problems with the wrong tools. By viewing the business problem and the solution in a unified context, DBI's business architecture approach creates solutions that directly address the business problems at hand.

About DBI Consulting

DBI is a business technology consulting firm that specializes in integrating internal business operations. DBI breaks down informational silos between internal systems and departments to establish the corporate-level information relationships critical to company executives. They don't simply patch systems and technologies together; they focus on integrating business operations – so the core business problem is addressed, not just a symptom.

