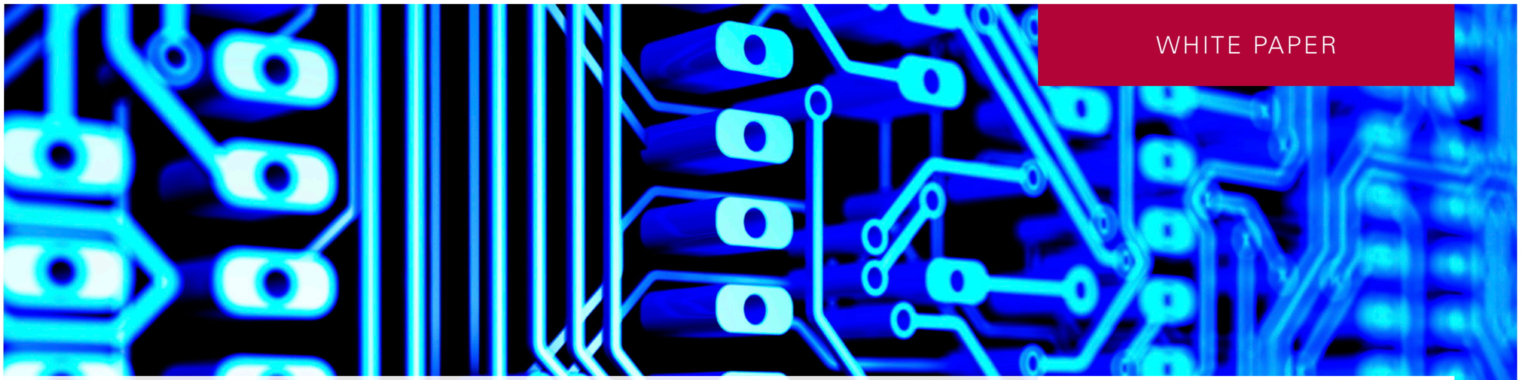


WHITE PAPER

Nimble Data Architecture: What ‘Getting Data Integration Right’ Can Do for Your Business

LIAISON®

Executive Summary

Your business succeeds or fails based on its performance in the physical world of creating and delivering products or services to meet customer demand but a critical component at the foundation of your success may exist out of view, in the abstract world of your company’s data architecture. With this in mind, it pays to ask yourself a fundamental question: Do you and your IT staff have a clear picture of the abstract layer of your business and whether it is contributing to your overall performance? Architecture in the physical world organizes space in ways that either facilitate or hinder the work or personal activities of people. Architecture in the world of data can play a similar role. Though data architectures invariably are designed to streamline business activity, they and the business can evolve to the point that business is hindered, rather than helped. A company’s product delivery, customer satisfaction, and financial performance can suffer because of outmoded data architecture.

For most companies, data architecture develops in stages. Often it is the sum total of minor, incremental changes made during a series of specific data integration projects. Unfortunately, many data integration projects can constrain data architecture instead of enhance it. If your IT staff can maintain a good architecture or significantly improve a problematic architecture, the abstract layer of your business can contribute to your success in the physical world of meeting customer requirements.

In this white paper, we describe how executive and IT management can get and maintain control of their company’s data architecture to help meet business objectives. We describe the relationship between data integration projects and data architecture concepts and practices. We describe how the business planning and IT development processes that direct and leverage data integration projects depend on tooling. We describe breakthrough ways in which data architects, business analysts, programmers, and business users can collaborate to address their organization’s pressing business challenges.

Your business has a data architecture. It will either help or hinder you. By adopting a strategic approach to your data architecture requirements and by controlling tactical data integration projects, you will better understand and control your architecture so your business can succeed.

Bad Data: Real-World Characteristics of Architecture Gone Awry

Even if each piece of data in your business is accurate, all the data combined may not be a business asset. Your data overall might be unwieldy and unresponsive to change. While good data can be easy to work with, bad data can be hard to work with. Whether your data is “good” or “bad” in this sense is a result of how you have integrated the sources and consumers of information in your company and how you have created a data architecture.

Data integration is the art of wrestling things not designed to work together into working together. Data integration projects often occur because of time-sensitive business initiatives like the acquisition of one business by another. Data integration projects have defined funding, start dates and – ideally – end dates.

Data architecture is the art of exposing and benefiting from the underlying conceptual order in data. Data architecture initiatives often occur because data integration projects are falling behind or are too expensive. Data architecture guides decisions people make so that each person can contribute to the health of the organization.

Data architecture in effect exists so that data integration projects can be done efficiently. However, the reality at many companies is that data architecture is the combined result of tactical decisions made during past data integration projects. Those past projects were focused on solving specific problems, not at making future projects more efficient, in fact, they often make future projects inefficient. Thus, to gain future efficiency through good architecture, existing data integrations may need to be redone.

The needs of architecture and projects are often at odds until the architecture is mature enough – good enough – to make data integration projects more efficient. They are often at odds until the ideas behind the architecture become part of the common, shared culture of the organization. Mature architecture will provide clear and consistent usage of data formats, even if it uses many techniques behind the scenes to produce that result. This clarity facilitates data validation, transformation, security, ease of understanding and use, as well as many other benefits.

A good architecture helps companies overcome multiple challenges, such as:

- **Heterogeneous IT Environments:** Many companies, especially those that have decentralized IT or that have grown through acquisition, have heterogeneous IT environments where more than one tool (such as an enterprise service bus (ESB) or data modeling tool) fulfills a specific business purpose. Data integration projects are often constrained by the tooling impacting that project, and often staff members have to be retrained in order to begin new projects. Data architecture is often complicated by heterogeneity, which can undercut the ability to understand, control and direct the effects of data integration projects. Heterogeneity complicates analysis and understanding of a company's existing data architecture. Good architecture often has significant hurdles that must be cleared before it provides the global perspective needed to make future integration projects more efficient.
- **Usage of Data Formats is Inconsistent:** Data formats built for various purposes by disparate teams in separate projects often have inconsistent structures, data types, and subtle differences in semantic usage (the way information is defined in a data architecture). These structural differences complicate data integration and semantic analysis. Semantic differences and ambiguity often lead to different interpretations and implementations of data formats by employees across the enterprise. Resolving semantic inconsistencies requires in-depth analysis and increases the complexity of data transformation efforts. These challenges further create inefficiencies in core business processes. Because of the inconsistent use of tools, and projects that do not keep track of information once they end, the seemingly simple tasks of collecting and analyzing data formats is difficult, identifying semantic differences can be much more difficult, and tracking semantic differences in a heterogeneous environment can be prohibitive.
- **Inability to Leverage Previous Usage of Data Formats:** A company's employees often have no way to access data specifications and data transformations already produced to address a similar or identical task. They often cannot perform impact analysis and have difficulty planning new initiatives like creating a canonical data model that describes the data needs of the business. They frequently have to reinvent data representations – often resulting in new structural and semantic variants of the data, complicating future integration projects. Again, the inconsistent use of tools and projects that allow crucial data to be forgotten makes this more difficult.
- **Inability to Govern Changes to the Data Architecture:** Projects may meet short-term needs, but fail to address the long-term needs of a business. Many organizations lack the ability to govern data architecture changes as they are being developed and deployed. As business requirements evolve, the data architecture and supporting business processes compromise the company's ability to make future changes - again, heterogeneity is complicit.
- **Difficulty Foreseeing Long-Term Architectural Consequences of Decisions:** The organization lacks the tooling and flexibility to avoid taking actions that produce adverse effects. Faced with time pressure to complete a large task, ill-advised data architecture decisions are made because the tools to foresee the architectural implications of those decisions don't exist. Inaccessible data format definitions, translation definitions, process definitions,

and heterogeneity all complicate this. The common theme of these productivity drags is bad data. Bad data is data that is not available, is in an unusable format or is inconsistently used. Finding ways to overcome these problems requires an information architecture that provides the tools the business needs and that is the focus of this paper.

Bad Data: Real World Results of Data Integration Gone Awry

Like most maladies, bad data problems exhibit particular symptoms. The following examples illustrate families of symptoms and each malady is drawn from the experiences of multiple companies.

IT Budget Consumed by Maintenance

When testing and repairing the consequences of a data integration change dwarfs the cost of the change itself, the business suffers. If there are fragile data flows and systems that can’t be changed, the cost of planning and making an enhancement elsewhere can be extraordinary. Because data integration changes are inevitable, companies must find a way to manage these costs without consuming all or most of the IT maintenance budget.

Point-to-Point Data Mapping

For an environment with few systems that need to communicate, creating and maintaining point-to-point data flows and data transformations can be highly efficient. For instance, when an application has a new requirement and needs to share additional data, few data transformations need to be updated and the data architecture is minimal. However, as the number of systems increases, so do the number of connections and data transformations. Point-to-point architectures start easily, but in the end maximize the cost of change for companies with complex data requirements.

Inflexible Midpoint

When a data flow has an intermediate step with a data format that cannot change, innovation is stifled. For instance, if a publisher has multiple subscribers that strictly validate messages, adding information to the data format the publisher sends requires either creating a new publisher or changing all the subscribers. This can be worked around by using an unused portion of the publisher’s messages for the new information. Over time, as new information is required that does not have a natural place in the data format, the new data is forced into the data format in unnatural places and the data format becomes progressively more difficult to use. This incremental semantic corruption of the intermediate data format takes on a life of its own, requiring the company to devote significant resources just to manage the inconsistency in how data elements are used.

Communication Breakdown

Data integration takes two dissimilar things and translates between them. Good translation requires understanding both components. Many organizations address this by a process in which data architects, business analysts, and others create translation specifications and then send the specifications to mappers. These specifications are often untested spreadsheets or text documents. This processes results in documents that often have errors discovered either by a mapper or later during a testing cycle. Resolving these errors requires rework and additional discussions between the mappers, data architects, business analysts and QA testers. One common approach to the problem is to add more mappers and testers, which are often overseas. Addressing a communication problem by scaling out to separate locations introduces additional communication problems. Knowledge and communication distance increases as the number of communicating entities increases. This is similar to the communication gaps caused as the number of point-to-point mappings increase.

Data Integration Starts Before Key Facts are Known

Data integration projects often begin with incomplete information and without success criteria. For example, the mapping process may start before the data formats are fully understood, and the true mapping requirements may only be discovered and proven during end-to-end quality assurance (QA). Once QA begins, the team will discover that the data format definitions or data transformation requirements were not completely understood. As a result, the mapper must open the map, become familiar with it, identify the problem, correct the problem, unit test, and pass the map back to QA. In some situations, the map must be rebuilt. At a minimum, this is a time-consuming process that results in rework and additional risk, especially when specifications are not kept current as changes are made.

Strategic initiatives like acquisitions or migrating to a new enterprise service bus create time-driven pressures to complete projects quickly. This rush often results in false economy. For this class of malady, the tools used, the process used, and the assumptions people bring to the project perpetuate the malady.

Inconsistent Service Oriented Architecture (SOA) Schema Usage

Service-oriented architecture (SOA) requires the designers of a service to balance the ease of creating the service's schema with the ease of using and maintaining the schema. Many enterprises generate XML schemas from logical data models, or use a pre-built schema library (such as OAGIS schemas or Oracle's EBOs). This allows them to rapidly create services that express necessary information. The richness of the message information lets designers avoid frequently creating new versions of services to satisfy the needs of a new user. However, users find that this architectural approach makes using the service extremely difficult.

These general schemas include many optional pieces of information, use the same structure in multiple places, and include multiple variants that are much like each other. Business analysts and data mappers often have to make guesses when transforming between application data and the service. To be safe, data mappers may populate data locations the service currently ignores, but which may be used by the service in the future. SOA implementations that do not publish and enforce service usage specifications become riddled with inconsistencies and inefficiencies, which result in delayed projects.

Problems with Authority

In this malady, "authority" refers to authoritative information. It is characterized by problems deriving, locating and sharing information that can be trusted. XML schemas, interface usage specifications, mapping specifications and other artifacts are only authoritative when they are:

- **Published:** Users can locate the correct version of the artifact from a secure, central location. For instance, teams should not keep specifications on laptops where they are difficult to locate, easily lost and quickly outdated.
- **Accurate:** Users can have confidence in the information in the artifact. The artifact has been thoroughly tested and proven. For instance, a mapping specification generated as a report from a working data transformation is better tested than a hand-created spreadsheet.
- **Up to Date:** Users can easily update or regenerate the specification when XML Schemas, Web Services Description Languages (WSDLs), application flat file format definitions, or other information changes.

When an organization's process is to manually keep artifacts synchronized with each other, the organization spends excess time and confronts errors. The lack of proper tooling to keep artifacts synchronized or to provide the ability to generate artifacts on demand leads to many problems obtaining accurate, dependable, and current specifications. It also hinders governance.

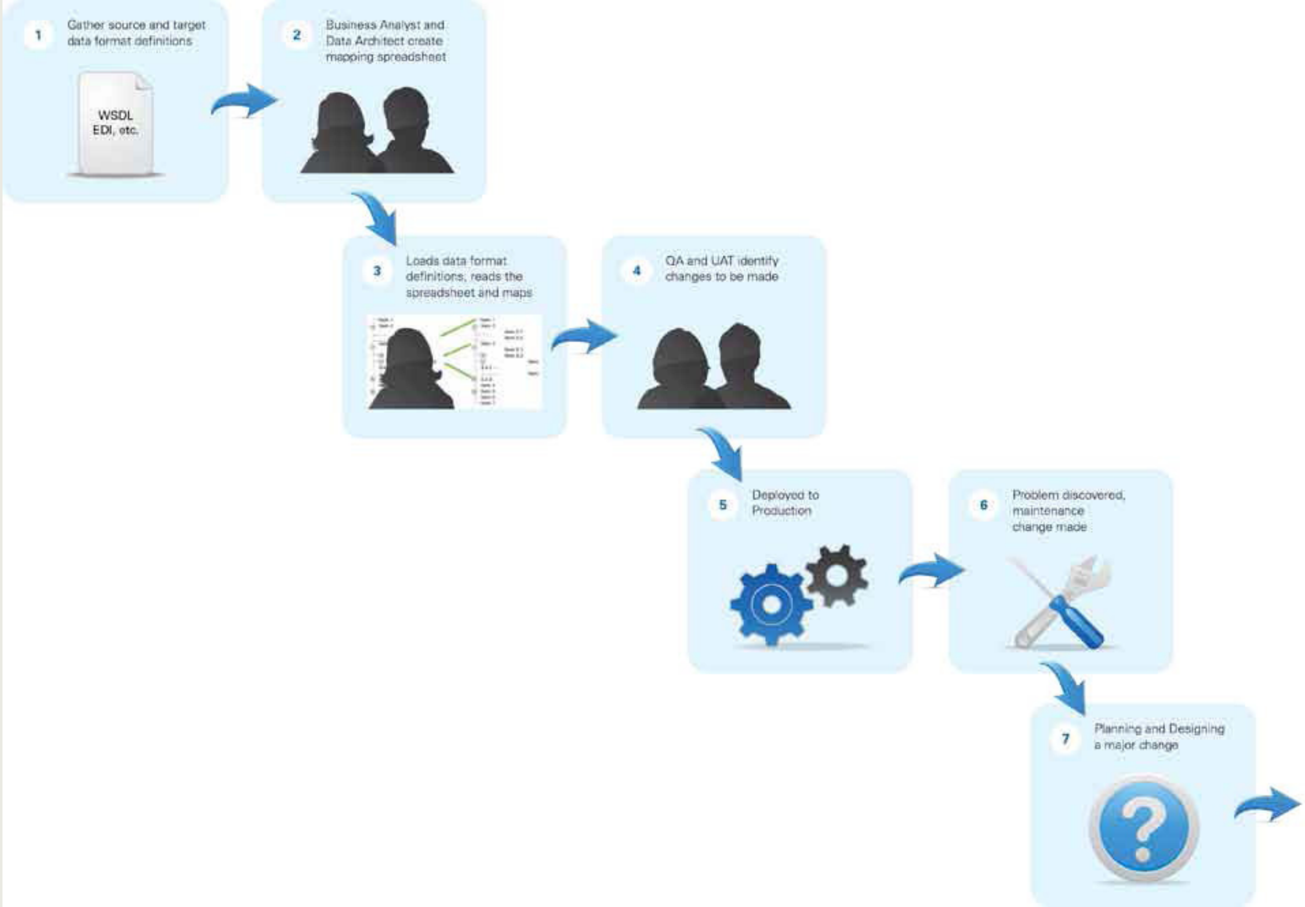
Lack of Governance

Providing authoritative information and standards is not the same as enforcing and maintaining standards. For instance, if specifications are not consistent, or if data flows and data transformations are not adequately validated against the specifications during testing and after deployment, bad data will be cemented into the architecture. Users may creatively warp the architecture if XML schemas are used to create usage or semantic specifications, which they are not designed to do for complex messages derived from rich information models. Business processes and tooling which do not adequately govern change at design time and run-time cause many problems.

Process Problems

For purposes of illustration, Figure 1 shows a data integration process commonly used today. This process is essentially an inflexible waterfall development model and is characterized by error-prone manual steps and conflicting sources of information authority. There is pervasive use of spreadsheets, which require manually typing in data and cumbersome processes for updating specifications when data transformation requirements change. There are multiple sources of authority emanating from spreadsheets, data maps or other sources. In an instance when SOA is deployed, usage information needs to be published for the message structure described by the WSDL.

FIGURE 1: A Common Approach to Data Integration is a Sequential Process That Lacks Flexibility


1
Gather source and target data format definitions
WSDL EDI, etc.
2
Business Analyst and Data Architect create mapping spreadsheet
3
Loads data format definitions, reads the spreadsheet and maps
4
QA and UAT identify changes to be made
5
Deployed to Production
6
Problem discovered, maintenance change made
7
Planning and Designing a major change

Using this approach makes it difficult to efficiently adapt to changing requirements like changes in message formats. In this diagram, multiple maladies result:

- **Communication Breakdown:** Information does not easily flow from one user to another. Information often does not flow from later stages in a project to earlier stages.
- **Data Integration Starts Before Key Facts are Known:** Because projects take a long time, there is project and management pressure to get started early.
- **Inconsistent SOA Schema Usage:** When users keep data exclusively inside spreadsheets, knowing about this data, locating relevant specifications, and then re-using it is difficult or impossible. It is often easier for people to avoid searching and wrestling with old data than it is to develop an entirely new specification. For any of several reasons, new specifications often are inconsistent with the old, creating inconsistent use of SOA services schemas.
- **Problems with Authority:** All too often, everyone has to get together in a room and make a decision about a particular item in a project. Then, they have to do the same thing again for the next project. Intellectual property is lost and reinvented – inconsistently.
- **Lack of Governance:** There is little visibility into the results and decisions of a project. Thus, governance of projects to improve data architecture is an extreme overhead expense, and is often not done.

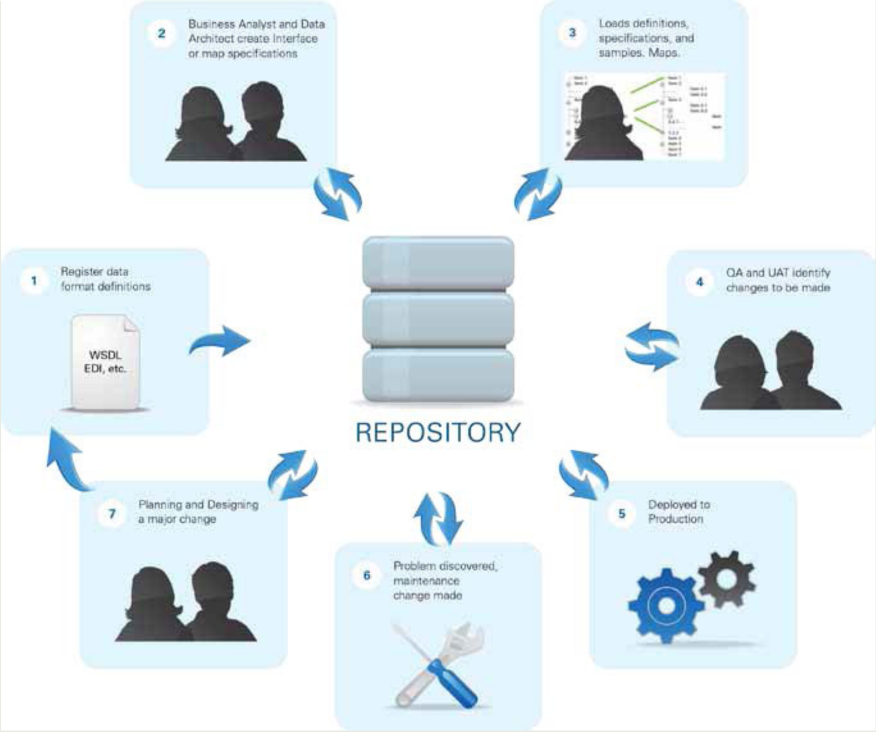
Good Data: New Tooling and Processes Create Business Opportunities

Recent advances in technology are creating opportunities for data architects, business analysts, programmers, and business users to collaborate to address business challenges with greater speed and more flexibility. Innovative ways to manage a company's information architecture that were once out of reach are now possible, tho some of these advances are relatively new and are not widely used.

Tooling and process advances have made a more agile development process possible. In Figure 2, there is a central repository and communications hub that enables a dynamic data integration process in which key players can easily collaborate and adapt. In this scenario, the tooling makes it possible to:

- **Create Infrastructure Flexibility:** Heterogeneity, as we have discussed, complicates things. In other ways, it simplifies things, by allowing an organization to use the best tool for a given integration project or portion of an enterprise. Organizations can bring homogeneity into a coherent data architecture based on:
  - Tooling that provides a global perspective of data formats
  - Data format specifications that reflect run-time data transformations
  - Logical data format analysis tools
  - Logical data models
  - Enterprise vocabularies
  - Semantic technologies
  - Governance process step-stools
  - The ability to deploy runnable processes and operations to the different run-time environments
  - Monitoring run-time environments
- **Centrally Store Information:** By storing artifacts in one centralized location, a business can manage the homogeneity in tooling, capture data from integration projects, and provide governance over the shifts in the data architecture inevitably caused by integration projects.
- **Easily Access Information:** Tooling needs to provide search capabilities so users can easily locate the artifacts and information stored in the central repository.
- **Publish Authoritative Information:** Capabilities are included so that authoritative information can be published to many users. For example, a business analyst might need summary spreadsheets about the usage of a data format. If this information is automatically refreshed or created on demand, users can have confidence they are looking at current, reliable data.
- **Increase Project Agility:** Business agility is a relatively new concept and for many businesses is an elusive goal. It has become necessary because the speed at which business requirements change has increased dramatically in recent years. The tooling and business processes associated with agile software development play an important role in enabling project agility. It is now possible for developers to adapt tools to their needs instead of being forced to adapt their business processes to compensate for the limitations of the tools. Teams can now automate complex tasks like refactoring and running unit tests to become more nimble in their response to changing requirements in data integration projects and data architecture.
- **Better Communicate and Collaborate:** In an agile process, users can share changes coordinate changes, and see changes made by others, regardless of where they are located. For example, if a tester discovers a problem, he or she can easily and precisely communicate the problem to others who can address the problem by fixing a data transformation step, providing clearer instructions, or providing additional information to the tester.
- **Automate Processes:** New tooling enables automation of key tasks. For example, changing an XML schema can impact other artifacts and information. Automated processes can track the work that needs to be done and perform the required tasks.
- **Analyze the Impact of Changes:** The greater insight afforded by the new tooling and new data integration paradigm makes it possible to anticipate and perform changes. For instance, if a user investigates modifying an XML schema, impact analysis tooling can identify data flows that would also need to be modified.
- **Improve Governance:** Like agile software development, agile data architecture depends on advanced tooling and processes. Agile tooling is a prerequisite for impact analysis, the ability to verify the reliability of design-time and run-time artifacts, publishing authoritative information, and controlling change. Architectural visibility and control are essential to governance, which is required to reliably and flexibly address business needs. With greater visibility and control, it becomes easier to ensure that changes are made only by authorized users and that information is validated before it is promoted to a later step in the process.

Figure 2: Leveraging New Tooling Can Make Possible a More Dynamic Information Architecture


2
Business Analyst and Data Architect create Interface or map specifications
3
Loads definitions, specifications, and samples. Maps.
1
Register data format definitions
WSDL EDI, etc.
4
QA and UAT identify changes to be made
REPOSITORY
7
Planning and Designing a major change
6
Problem discovered, maintenance change made
5
Deployed to Production

Using new tooling to create an agile information architecture that better links data architecture and data integration can have far-reaching positive effects on business processes. In particular, development time can be shortened. Users of a service or common data format can work independently of the designers of the service and the back-end applications enabling the service. Data flows and data transformations can be built while XML schemas are still in flux, making it possible to encapsulate specifications in tooling so they can change as document formats change. Work can continue around the clock when low-cost overseas resources are used because communication is clearer, more comprehensive and more accurate.

Good Data: How Your Business Can Benefit

Widespread business benefits are possible when companies make their data architecture and data integration projects complementary components of their overall information architecture. These benefits accrue because projects run more efficiently as everyone in the organization works in unison toward a common goal. This harmony results because people and teams have better tools – and because organizations are able to create a culture of success.

Some of the key business benefits include:

- Common best practices and shared assumptions can be communicated
- IT and business units can communicate better with each other
- IT staff focuses on activities that contribute to the health of the business
- Business keeps pace with a high rate of change
- Ability to experiment with new business models becomes practical
- Better management of the variety of local business needs
- able to more successfully outsource data mapping and other data processes
- Your costs are reduced through shorter time to market and greater staff productivity
- Your strategic and tactical business requirements are balanced

To take this discussion a few layers deeper, here are four scenarios where tangible business value is derived by taking a holistic view of your data architecture and data integration.

Get Better ROI from an Acquisition or Merger

The costs of IT integration can make or break the value derived from a corporate acquisition or merger. In most situations, identifying the synergies in the products and customer sets that prompted the merger or acquisition is the easy part. The more difficult task is cost-effectively integrating the organizations' disparate operations – including their IT infrastructures and business processes – in order to make those synergies pay off. If the cost of the acquisition exceeds the incremental new revenue or better market position, the merger will be a net loss. Taking a holistic—and strategic—view of the IT integration requirements and using advanced tooling to support the integration can help realize the potential business value.

To begin with, there are usually significantly different business cultures, business data and processes, security policies, and master data management strategies (to name a few) at the two companies. While no technology can overcome cultural clashes, tooling can help people from the different cultures understand each other and facilitate their alignment. Even if data and business processes are similar, there often are subtle differences that can take significant time to discover and resolve.

A first step toward integration – simply moving data – may require connecting multiple enterprise service buses and other data conduits. The heterogeneity of the two infrastructures requires multiple run-times - and the staff with the skills to work with the multiple run-times - just to move data. The technical landscape as a result of the acquisition becomes more heterogeneous.

In the interest of time, key data integrations are often undertaken before the underlying differences in the two information architectures are clear. This creates a set of inflexible point-to-point mappings, which complicate the IT team's task to create a new enterprise-wide data architecture.

Complex challenges due to differences in data and process, and differences in technology, have to be solved quickly by people steeped in different corporate cultures. Many things are learned rapidly and at a cost. In addition, as a result of the rush, many things are forgotten.

That is a problem which tooling can help solve. If the teams use tooling that captures the results of decisions and that can extract information about of the data integrations themselves, little will be forgotten.

After the initial integration rush, the organization is then often left with two separate infrastructures that have been taped together. Tools cannot solve this messy problem, but tools can give data architects the ability to analyze and plan, give business people insight into the viability of different initiatives and give business analysts guidance as to how to do projects in ways that contribute to a better architecture.

Profiting from a merger or acquisition requires creating and operating a new, unified business. Tooling to facilitate good data architecture can have an especially high ROI at such times.

Manage Change at Lower Cost

Given the rapid rate of change in business today, the ability to cost-effectively adapt IT infrastructure to new requirements is a prerequisite for success. IT infrastructures are typically inflexible and this is especially true of the information architecture at most companies.

While "bad data" is not easy to direct, good data architectures adapt to change more easily. With good data architecture, teams doing individual projects will have fewer surprises and will spend less time re-inventing intellectual capital that had been lost. With good data architecture, changing one integration will have a smaller circle of consequences. With good data architecture, IT will better know what it needs to do starting with the first conversation with the business through the final rollout into production. This has several positive business consequences.

First, the business can adapt to new market opportunities faster. It can also organize special promotions and programs faster and can create partnerships faster. In short, IT will not keep the business from being agile.

Good data architecture can also reduce the drag regulatory compliance can create. From assessing the impact through implementation and into completion, adapting to regulatory changes needs not be onerous.

With good data architecture, the business can be more proactive with the same budget. It can address risks, such as security concerns for sensitive data, before a leak occurs. It can also eliminate the dead-weight of pent-up projects.

Just as a building designed for a business can make the business more efficient and let it provide better service, a data architecture supported by the right kind of tools can improve the profitability and services of a business.

Integrate IT into the Core of Your Business

People working in IT like to think they are special. They solve technical mysteries, use words without vowels, and often have interesting personalities. In some cases, IT people can be treated as special enough that normal things like accountability are ignored.

Not every organization treats IT this way, but for IT to be a good partner to business units, and for the business to respect IT as a strong partner, companies need to apply the same cost/benefit analysis to IT projects as they do to other core business activities.

Furthermore, IT staff members need to know how their performance affects the health of an organization. If a salesperson is rude and loses a sale, the organization clearly didn't benefit. With IT, protected behind walls of acronyms and mystery, the relationship isn't as clear. Often, IT team members don't realize what is at stake for the business in decisions they make, such as decisions which make a current project easier, but complicate future projects.

A data architecture is a key part of IT's business plan. It tells how it expects to be able to scale, how it will adapt to the needs of its internal business customers and ultimately how the company will respond to external customers.

Good data architecture and the tools to maintain it are key tools for the health of the business overall. Good data architecture and tools are essential for individuals in IT to better contribute to the success of the business.

Realize the Business Potential of Cloud Computing

At an abstract level, it is easy to see how moving business processes to the cloud can reduce IT costs and increase business agility. On a practical level, however, integrating cloud computing into your business is fraught with process and technical obstacles and business risk.

Cloud computing can increase both the distance that the information must move and the available expertise to move it. Adopting cloud computing means relying on people less steeped in the corporate culture and history. Thus, more information needs to be articulated and deliberately shared. Team members won't just remember key information about systems and data, they will need to find out or be told. Thus capturing intellectual property and making it available increases in importance. At the same time, cloud computing provides an organization access to people more focused on particular practices or technologies than any one organization can provide.

One of the major technical obstacles organizations encounter is the need to reconcile disparities between the company's approach and the cloud provider's approach to data flow and data transformation. If a data flow involving a transformation step is to be migrated to the cloud where a different approach is used, it can be costly and time consuming to reconcile the two technical approaches. For cloud computing to be successful, companies must:

1. Capture intellectual property
2. Publish specification information
3. Use tooling to analyze transformation steps
4. Generate or capture test data samples
5. Test data flows
6. Make information about data flows easily accessible

Using cloud computing can increase security concerns, since sensitive business information and customer data moves back and forth between a company and the cloud provider and within the cloud. As run-time data is moved to the cloud, it can more easily be seen by unauthorized people. Techniques such as tokenizing credit card information and other personally identifiable information can be used to protect sensitive, high value data from threats. Having good data architecture allows an organization to identify what needs protecting and allows the organization to govern integration projects to ensure that data is handled securely. It also enables the firm to articulate all of this information to their customer base or governing authorities, if required.

Cloud computing can better interoperate with non-cloud infrastructure via tools that provide a global behind-the-firewall and cloud picture. At design time, tools that provide visibility into on-premise and cloud data flows allow global data architecture analysis. At run-time, tools that trace on-premise and cloud data can help IT track down problems in a timely manner and ensure service levels are met.

Good data architecture simplifies each of these steps. Developing and maintaining good data architecture depends on tools, and without IT tools there would be little need for data architecture. With IT tools comes the need for more tools – and the opportunity for stronger and more competitive businesses that better satisfy customers.

Conclusion

In order for a business to succeed, the pursuit of new process efficiencies needs to be pervasive throughout the enterprise. Potential new business value can be found in some unlikely places and one of those sources of new business value is the abstract world of data architecture. In this paper, we have outlined how data architectures can work for or against the goal of greater business efficiency. We have shown how advances in tooling can help companies create an information architecture that contributes to improved competitiveness and profitability.

Your organization has a data architecture. It is either helping or hurting you. If it is hurting you, breakthroughs in tooling can make addressing a problematic architecture practical. Tooling can help you improve the efficiency of data integrations based on a good architecture, and can help you govern new integration projects so they maintain or improve the architecture. The foundation for business success enabled by data architecture is now within reach.

Atlanta – US HQ
3157 Royal Drive
Suite 200
Alpharetta, GA, 30022
Tel +1.866.336.7378
+1.770.442.4900
Fax +1.770.642.5050

Finland – European HQ
Patamäenkatu 7
33900 Tampere
Finland
Tel +358 (0) 10 3060 900
Fax +358 (0) 10 3060 901

The Netherlands
Liaison Technologies BV
Barbara Strozzilaan 201
1083 HN Amsterdam
The Netherlands
Tel +31 (0) 20 700 9350
Fax +31 (0) 20 330 6478

Sweden
Ålkärrsvägen 32 681 38
Kristinehamn
Sweden
Tel +46 708102213

United Kingdom
Liaison Technologies Ltd
Brightwater House
Market Place
Ringwood, BH24 1AP
Tel +44 (0) 1425 200620
Fax +44 (0) 1425 478656