Information integration and the data warehouse
White Paper
January 2005

Information Integration: New capabilities in data warehousing for the on demand business

Dr. Barry Devlin
IBM Software Group

Dr. Bill O'Connell
IBM Software Group

ON DEMAND BUSINESS™

Contents

Introduction

Since the publication of the white paper "Information Integration—Extending the Data Warehouse" in March 2003 and the subsequent introduction of IBM® WebSphere® Information Integrator and the ongoing enhancements to IBM DB2 Universal Database™ (DB2® UDB), customers and partners have begun to integrate the new functions and tools into their existing environments. Their experiences, as well as the views of other players and analysts in the industry, confirm our original view that information integration allows the extension of the warehousing concept to include access to data that physically resides beyond the warehouse.

Further developments in both business and technology, however, indicate that an even bigger change is taking place in the marketplace. This change is causing leading companies to re-examine some of the fundamental concepts of data warehousing. The data warehouse is being reinvented—a reinvention that is being enabled and driven by the power and extensive functionality provided by DB2 UDB and WebSphere Information Integrator technology.

In this paper, we will briefly review the fundamental assumptions behind data warehousing and how they have led to today's implementations. We will use current, real-life experiences of data warehousing to show how business needs are changing and how implementation approaches are evolving in response to those changes. New tooling and the new possibilities that it provides are described briefly. Finally, we will present an updated view of the data warehouse as it is reinvented to meet tomorrow's on demand business needs.

A brief history of data warehousing

The first and most fundamental paradigm of data warehousing is so obvious that it is often overlooked. Data warehousing assumes that the world of information processing can be split into two parts: operational and informational. Although clearly related, these two areas—essentially areas of business activity—are assumed to operate largely independently of one another. This enables—and indeed drives—the existence of two separate spheres of IT implementation: operational systems responsible for running business transactions and a data warehouse environment for managing the business. This split was clearly advantageous for IT because it protected the operational systems from the performance and security impact of ad hoc queries. It also suited a business model in which different business functions operated in distinct and largely independent silos.

In the early 1990s, the data warehouse environment itself provided two conceptual sets of function: the business function supporting data analysis and decision-making, and an underlying and largely hidden function of "impedance-matching" between the detailed, disjointed and frequently "dirty" world of operational systems and the wide variety of informational needs of business users. These two functions were supported by derived and reconciled data respectively, and came to be known as data marts and the enterprise (or business) data warehouse. The need for a reconciled data layer is also based on the assumption, in this case, that the data in the operational environment is too complex, disjointed or dirty to be easily used by business users.

Throughout the 1990s, this conceptual division of the world of data into three areas became the basis for the physical implementation of the three-layered architecture consisting of operational systems, an enterprise data warehouse (EDW) and data marts, with batch-based extract, transform and load (ETL) processes linking the layers. Today, this architecture is the most popular approach to decision support[1] by a factor of two.

It has become increasingly obvious in recent years that this architecture cannot meet all business needs. It is particularly stressed by requirements for closer to real-time decision support that can easily overwhelm the warehouse with immediate, voluminous, but seldom-used details. Closed loop processes in

[1] Source: TDWI-Forrester Quarterly Technology Survey. San Diego Conference, August 2004.

which decisions must be fed back instantly into the operational environment also present significant data currency and consistency challenges to this architectural approach.

Furthermore, as the importance of unstructured data (or content) has increased, the limitations of the extract and copy approach of data warehousing have become clear. Volumes of unstructured data are often so large that making and maintaining copies can be prohibitively expensive. Differing technical requirements for storage or processing make it difficult to create combined structured and unstructured data stores. In addition, legal and privacy constraints are often onerous for such content.

IT factors also drive architectural change. The total cost of ownership (TCO), including disk and personnel costs, can be reduced by driving down the number of copies of data. Application development can also be speeded up because changes to applications can be accommodated more quickly by altering views in the database than by modifying ETL and physical data models and then reloading the data marts. This can be particularly important during the early stages of an application's life cycle, when users' needs are being dynamically discovered. In rapidly evolving businesses, this benefit can pervade the entire life cycle of the application.

Even more recently, the industry has focused on pervasive business processes and the need to monitor events and enable proactive corrections through business performance management. This raises further questions about the traditional data warehouse approach.

Over the last 10 years, these needs and industry directions have led—with increasing urgency in the past 24 months—to the emergence of a variety of new constructs within this data architecture. Among the earliest attempts to address the timeliness of data was the operational data store (ODS), which has become a standard feature of most implementations. Other components that have emerged include operational data marts (or oper-marts), Web marts, warehouses specializing in enterprise resource planning (ERP) or customer relationship management (CRM) packages, federated data access, virtual data marts and, most recently, the virtual ODS.

Current experiences in the field

Currently, work at data warehouse projects is beginning to reveal some new trends. The following examples take a look at what is happening in real life. Our experiences fall neatly into four categories.

Faster feeds, increased throughput

Over the years, decision makers have become increasingly impatient for information they require to support their analyses. Monthly reporting has become weekly, then daily, and in some cases, even more frequent. Delays of weeks or even days in gathering, massaging and consolidating information across the organization have become unacceptable. Meanwhile, the volumes of data flowing into the warehouse have continued to grow as the business is tracked at increasing levels of granularity and around-the-clock business activity is shrinking or eliminating the batch windows when data warehouses were typically loaded.

The result is that organizations are demanding ever faster feeds and more efficient processing of incoming batches of data. While performance of ETL tools has improved, large gains have also been derived from optimization of database load and update processes. Improvements in data capture and replication tooling, for example, in IBM WebSphere Information Integrator Version 8.2, have also enabled more data to be fed more quickly into existing data warehouses.

DB2 UDB has also moved to a more real-time maintenance mode by introducing online utilities. Examples include online backup, load, table reorganization and statistics gathering. Administrative activities such as database configuration changes are also now online. There also is a tighter integration with WebSphere MQ for constant trickle feeds of data using the built-in WebSphere MQ Listener in DB2 UDB and built-in MQ user-defined functions (UDFs) for sending and receiving data flows. This capability allows pluggable stored procedures (SPs) in the data stream for transformations as the data is taken off the queue, allowing data to be loaded into production tables while simultaneously available for query.

Furthermore, DB2 UDB has also introduced tight collaborations with IBM storage and servers. For example, integrated split-mirror backups can be run during heavy query workloads, even while updates are occurring. Server virtualization can be used to dynamically switch CPU power from one or more database engine logical partitions (LPARs) to LPARs running ETL functions on demand.

However, the more interesting and challenging change relates to the timing of feeds into the warehouse. Many organizations are investigating and some have already implemented feeding the warehouse at regular intervals throughout the day with smaller chunks of data. This is driven by both business and IT needs. Business needs include decision support for 24×7 processes and multiple work shifts. Among the IT drivers are the increasing volume of data and shrinking batch windows. For the ETL tools themselves, this approach does not present any real problems—most tools can easily support smaller batches of data on a more regular basis. In many cases, we can see the emergence of true record- or transaction-level processing approaches.

Unfortunately, in the context of the overall warehouse maintenance process, significant issues arise from both the warehouse structure and the timing restrictions imposed by old operational systems. The physical design point for most data warehouses is based on the assumption that data will be loaded while the warehouse is effectively offline. Reconciliation and cleansing occur at the same time as loading, and during this period, consistency of results from user queries may not be guaranteed. Loading a warehouse while users are online may require substantial redesign of warehouse tables and load processes.

Most customers, however, face a more difficult problem. Many operational systems do not maintain continuous and complete data integrity with peer systems—sometimes even internally—during ongoing processing. They depend on close-of-day processes to complete these tasks. Prime examples occur in the financial industry, where transactions such as debits may affect one account instantly but the corresponding credits to other accounts are not recorded until the following night's batch run. Loading this type of data on a continuous basis during the day through ETL tools simply replicates the inconsistency of

the operational systems into the data warehouse and then requires additional processes at the end of the day to clean up the intra-day inconsistencies. Such a situation is likely to be unacceptable to business users and unpalatable to the IT department.

As enterprises are beginning to discover, the solution to these problems lies in recognizing that some decision support needs (such as intra-day consistent data) can be met only through fundamental changes in the operational environment. Fortunately, such changes are already being driven through the growing need to adopt the on demand business paradigm. Enterprise application integration (EAI) provides the basis for real-time interactions between separate applications in the operational environment by using a message-based approach. Increasingly, this infrastructure and the messages exchanged between different operational applications can also be seen as a foundation for passing information to the data warehouse, ODS or other downstream environments. While the transformation function available in today's EAI tools is considerably less sophisticated than what is available in ETL tooling, some companies are beginning to find significant advantages in using the same infrastructure and metadata to interconnect all applications, both operational and informational.

Consolidation, reduced data duplication

As data warehousing has evolved, one feature that has become painfully obvious is the almost geometric growth in the number of copies or near-copies of data within the enterprise. These copies[2] of data have emerged for a variety of reasons. In the past, improved or even acceptable performance of end-user queries could only be achieved by creating specialized copies of data with specific physical data layouts, pre-aggregations and/or restricted volumes of data. Such copies are, of course, known as data marts.

Another driver for data duplication is the belief within organizations or departments that they need local copies of important data under their own control to assure access or quality. This often leads to a proliferation of data marts, some fed from the data warehouse and some directly from the

[2] In the context of data warehousing, "copies" of data do not imply exact replicas, but rather general derivations from a set of source data.

operational systems, but many more being fed in lengthening daisy chains from pre-existing marts. Further duplication arises from business decisions such as mergers and acquisitions, often leading to multiple enterprise data warehouses within the same organization. In addition, many packaged application vendors insist on providing specialized data warehouse implementations for their own environments, an approach that actually runs counter to the true spirit of data warehousing.

Such data duplication is not without its problems. All of these marts, warehouses and operational data stores must be loaded and updated on a regular basis, leading to bloated, long-running and complex ETL processes. Data storage needs increase, although the impact is usually offset by reduced disk costs. Of greater concern, however, is the cost of managing these multiple data stores and the increasing likelihood of inconsistency between them. Such inconsistencies defeat the entire purpose of data warehousing.

Some enterprises today have already begun the process of consolidating their informational environments, reducing the number of copies of data to a more manageable level. It is important to note that consolidation of a data warehouse environment does not necessarily mean collapsing it all to a single, highly normalized database in which each data item exists once and only once. Some level of redundancy is not only necessary, but desirable.

For example, experience of highly normalized data warehouses shows that some tables are joined in the majority of queries that use them. Similarly, summaries and aggregations that exist at the data mart level are constantly used and reused. Pre-computing and storing such joins and aggregates both improves query performance and provides the best balance between processing and storage resources. Such consolidation depends, of course, on the availability of powerful processors and specific database features to provide adequate query performance.

Recently, many large data warehouse customers have begun to test the possibility of combining multiple data marts into a single, centralized mart,

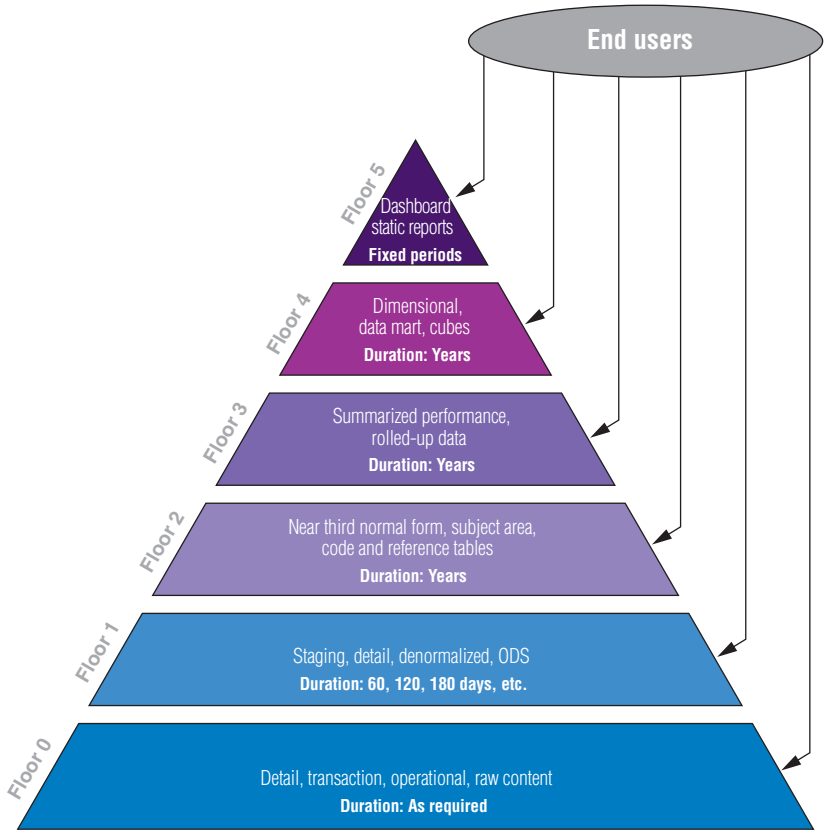

Figure 1: The information pyramid

using views and materialized query tables (MQTs) to create virtual data marts within the data warehouse or using common data to support some ODS and data warehouse needs. When using views in this environment, MQTs provide a powerful way to bridge the performance gap seen if accessing the normalized EDW directly via views by allowing queries to route to MQTs as appropriate.

This movement toward increased data consolidation gives rise to a more fluid view of the overall information environment, as shown in Figure 1. This diagram emphasizes several key points. First, it conveys the essential unity of the information used by an enterprise—from detailed transactional data to

consolidated and summarized knowledge. Traditionally, IT people have seen the levels of information as separate layers with mandatory data copying from one to the other (largely in an upward direction). The business, correctly, sees the levels simply as different views of the same information, although individual users usually focus on a particular floor to do their specific jobs. To emphasize this difference, we name these divisions as floors rather than layers. While some data copying may continue between the floors, this approach is no longer the only one possible.

Second, we can see that data in the different floors has different characteristics: volumes, structures, access methods and so on. We can choose how best to physically instantiate the floors; however, given appropriate technology, we could instantiate many of them together in a single physical environment. The diversity of today's operational environments does mean that Floor 0 will continue to remain physically separate in the short- to medium-term, but above this, any physical configuration may be possible.

Floors 1 to 5 can be broadly mapped to the layers in existing data warehouse architectures, because these layers are based on the same fundamental data characteristics that provided the basis for separating the different types of data when defining the architecture of the warehouse many years ago. However, such mapping disguises the essential unity of view of information and should only be used for migration purposes.

Finally, the diagram emphasizes that users today often require access to data at all levels in the hierarchy within a single activity or process. We will return to this final point later.

DB2 UDB is ideally suited to support a comprehensive and layered model of data usage that spans the full spectrum from transaction-consistent read/write activities to highly specialized analysis functions that require dedicated read-only data. DB2 UDB features include:

- *MQTs can provide auxiliary data structures to aid performance—specifically at floor 3 (supporting a virtual floor 4) and/or at floor 4 to support star schemas.*
- *Parallel SQL operations, such as MERGE (UPSERT), SELECT over INSERT/UPDATE/DELETE, Declared Global Temporary Tables (DGTTs), unlogged operations and the Crossloader enable building the upper floors in this approach. Such functions are critical when building new tables from existing ones for added application needs (for example, building parts of floor 4 or 5 from lower floors such as floor 3, which has the normalized subject area for the entire business).*
- *Workload management functions such as the DB2 Query Patroller and Governor address new challenges of mixed workloads in such an environment, including OLTP-like queries mixed with DSS-like queries.*
- *Tablespaces allow sand boxing in the larger environment that was previously done in data marts. Sand boxing allows application development within the production system. In addition to workload management controls, disk quotas prevent an application development team from taking over the database.*
- *Online utilities support an environment in which different parts of the system have very different maintenance windows. In the multi-floor architecture, the whole system may have a very small maintenance window, or ultimately none at all. In this case, online utilities are key.*

Event monitoring, near-real-time availability

Business activity monitoring (BAM) is a topic that is gaining considerable mindshare with business intelligence customers today. Its basic premise is that by monitoring events such as transactions, changes in external variables and intra-day performance indicators as seen in the IT environment, it is possible to deduce interesting conclusions and infer business actions that should be taken. For example, a sudden peak in transactions on a credit card account might lead a bank to believe that the customer was having financial difficulties and allow appropriate and early action to be taken.

It is clear that monitoring unusual events is possible only if the normal level is known. Hence, such monitoring has been largely restricted to the operational environment and the context limited to the data stored there. However, it is

becoming increasingly clear that this context should extend over longer time periods to enable more appropriate business decisions. It is also necessary to be cognizant of seasonal, cyclical or other longer term variations in the background activity level. Such historical data resides, of course, in the data warehouse.

Recognizing this, a number of vendors have proposed solutions that involve continuously loading real-time event data into the warehouse, allowing active monitoring and analysis of the events and their implications there. While this is an obvious warehouse-centric approach, it does have several drawbacks. Continuous loading of transactions into the warehouse may have performance implications for the ETL tooling or process, as mentioned earlier. However, more importantly, the approach violates one of the fundamental precepts of the data warehouse—namely that the warehouse provides a consistent and stable data environment for decision support for its users. The original design of both the data structures and the load processes reflects this requirement and is unlikely to be compatible with these new needs. Furthermore, continuously loading such unreconciled data into the warehouse immediately raises questions about how it can be eventually cleansed and reconciled.

Recognizing the essential unity of the enterprise's data, as shown in Figure 1, points the way to a more elegant solution—one that is attracting more customer attention today. This solution uses the analytical tools of the warehouse against a combination of historical data from the warehouse and event data from event monitoring tools in the operational environment. This approach removes the necessity to load potentially dirty data continuously into the warehouse.

Integration, linking in more data

The concept of information integration is gaining increasing credence among customers. Unfortunately, some vendors and analysts equate the phrase "information integration" with federated or distributed access alone and they are beginning to use phrases like virtual data warehouse or virtual ODS. We believe such phrases to be misleading and dangerous.

By definition and common usage, an EDS or ODS is a physically instantiated copy of operational data that has been cleansed, reconciled and structured in line with reporting or usage requirements. Some or all of such cleansing and reconciliation simply cannot be carried out in-flight (while the federated user query runs) as the word "virtual" suggests. At IBM, information integration implies the result: integrated information, not the method of integration. Making a copy of data and accessing the data at its original source or at another existing copy of that data all contribute to the goal of integrated information, and are thus included in information integration.

While strategic, more broadly implemented information integration platforms are still in the early stages of customer acceptance, experience to date leads us to conclude that (1) the federated query approach is viable with acceptable performance under reasonable conditions and (2) some store of reconciled data continues to be required as the common reference point in the environment. Let's explore these two points in more detail.

When we speak of "reasonable conditions," we assume network connections that are both fast and not overloaded, as well as processors that are operating within their optimum performance ranges. For every query, whether local or distributed, the database structure and the query need to be optimized for one another; therefore, some degree of enterprise data architecture is presumed. Although most federated queries deliver acceptable performance under such circumstances, we have found that when performance is unacceptable, manual optimization of database structures can often provide dramatic improvements. Such optimization may require the creation—and ongoing maintenance—of a copy of a common, usually small table from one machine to the other, perhaps implemented as an MQT in DB2 UDB that is effectively transparent to the calling application.

The continued need for a store of reconciled data is a vital consideration in most cases. Broadly speaking, it arises from the fact that few enterprises today have a single, internally consistent and non-overlapping set of identifiers for core

business entities such as customers or products. Reconciled data stores such as EDWs or ODSs can be viewed in simple terms as cross-reference tables between the keys of one application and those of another. An ODS, in particular, is a cross-reference table—often between multiple sets of keys—that is maintained in as close to real time as possible. An EDW also provides such cross-reference capability, although it focuses on consistency over time as well as across systems.

The existence of such a store of reconciled data is vital for information integration. Today, the best source for such a store is the EDW or ODS, and these sources are often central points of consistency for ETL and federated query implementations. However, the same need for a point of consistency, but in real time, is also emerging in the operational environment as enterprises try to link operational applications together through an EAI infrastructure. We therefore suggest that a common set of reconciled, consistent data will soon be required for both operational and informational environments. This will almost certainly be a distributed set of data, requiring information integration techniques to access and maintain it. The implication is for EAI and information integration functionality to converge.

When using WebSphere Information Integrator as an information integration platform, the sources that can be accessed include operational systems databases, message queues, content stores, application systems, Web pages, flat files, spreadsheets and many more. However, one source that is attracting much attention from customers at present is the data warehouse itself.

In such implementations, WebSphere Information Integrator is used to create queries that span two or more data warehouse instances. As a result of mergers and acquisitions or simply divergent prior IT strategies, many companies have multiple data warehouses, often on different platforms. Today, users demand consolidated access to data spanning these multiple warehouses. Such access can be achieved by physically consolidating the underlying warehouses. This, in fact, may be the preferred IT solution, since it reduces complexity and cost. However, there may be reasons why it is not currently feasible. The migration

project may be too costly or complex. The different business departments may be unwilling to cede ownership of their warehouses. In such cases, WebSphere Information Integrator provides the answer by allowing direct access across the existing data warehouses to data where it already resides.

In some cases, there may be no intention to physically consolidate the multiple data warehouses, marts and so on. For example, many businesses that have an enterprise-wide data warehouse also need a packaged warehouse such as SAP Business Information Warehouse (BW) for a certain part of the business. Theoretically, it is best to have one of these feed the other so only one master exists for a single, global point-of-truth. However, this is not always practical—federated query allows these two warehouses to co-exist, allowing users to benefit from the information in both.

Global corporations often have divisional and regional structures, each with its own data warehouse. For these largely autonomous organizations, this type of structure makes sense. However, there is always an overarching corporate headquarters function that needs to consolidate data at the company level. WebSphere Information Integrator allows such consolidation to be performed as part of the reports or queries that need to be run, without the need to copy all of the regional or divisional data to a single global data warehouse.

Another way to extend the reach of the increasingly popular warehouse is to reach out to unstructured or semi-structured data stores. For example, insurance companies have significant information about policies, claims and inquiries that often resides in a variety of content stores such as e-mail systems, XML file stores, content management systems and so on. For storage and performance reasons, it seldom makes sense to copy such data into the warehouse.

WebSphere Information Integrator allows queries that span both the relational content of the warehouse and the unstructured /semi-structured information that resides elsewhere. And through MQTs, it enables transparent caching of results of distributed queries so that frequently accessed or specially

favored information is physically stored locally for improved performance. For example, the XML-formatted insurance claim of a "platinum" customer, while only residing in the operational system, may be transparently cached locally to provide enhanced service to that customer throughout the claims process. Recent extensions of the content integration functions offered by the introduction of the WebSphere Information Integrator Content Edition offer powerful and flexible capabilities in this area.

Customers are also exploring ways to bridge the enterprise boundary with information integration in two distinct ways: one with public Web content and the other in cooperation with business partners. In the first case, diverse Web content can be made available in conjunction with internal data. Executives and other high-level decision makers who need to understand business performance in the wider context often find this combination of external and internal views of special interest.

The second case is perhaps even more interesting. Increased cooperation between companies and their suppliers, dealers or other partners drives the need to share a larger volume of more timely business information between organizations. However, considerations of privacy, business ethics and independence mean that it is seldom reasonable to store copies of a partner's data in one's own warehouse. WebSphere Information Integrator provides the ability to perform queries that access data from both sides of the organizational boundary, even leveraging service-oriented architectures to access partner data.

On demand decision making: Part of the business process

As shown in the previous sections, much of the business interest in decision support today centers around the speed with which decisions can be made. Such interest is, of course, reflected in the rush by vendors and analysts to support or discuss topics variously termed as real-time warehousing, right-time business intelligence or active data warehouses.

In many cases, the implicit assumption is that traditional data warehouse problems of temporal, historical and cross-system consistency have now been solved. Of course, real data warehouse practitioners know from personal experience that these issues often remain. However, we also recognize that much of the technology required to solve them exists; most often it is political will, organizational skills or simply money that is lacking. The challenge over the coming years will be to fix the consistency problems in the background while solving the business demand for more current information.

Applying the adjectives "real-time" or "active" to data warehousing can be misleading because they imply that the data in the warehouse should be fully up-to-the-minute or exactly current, which is seldom necessary. In many cases, it is impossible. By definition, the data warehouse is the logically single, consistent, historical record of the business. Consistency and historical truth actually emerge over time from the often chaotic or inconsistent events, positions and views that define human activity on a minute-by-minute basis. This does not deny a business need for more current information, but this physical reality of emerging truth does define limits around what can be achieved.

It is important to understand business requirements in terms of speedier decision support and how that relates to the traditional data warehousing paradigm. Over the past few decades, decision support had come to be seen as an independent, stand-alone business activity, human-centric and rigidly divorced from the day-to-day running of the business. The model that emerged was one of business analysts who pored over data and provided input to decision makers, typically managers or executives. For less complex analysis, the decision makers could even manipulate the data themselves using spreadsheets.

This separation of managing the business from running it is somewhat artificial. Decisions made while "managing the business" clearly should have an impact—and often do—on business activities, which feed into decision

making. This is a feedback loop. In fact, with a little more thought, it can be seen as a series of feedback loops operating at different speeds and at different levels in the organization. For example, a strategic decision to acquire another company affects many levels of the business organization, but its effects in the feedback loop to and from business operations may take many months to become apparent. At the other end of the spectrum, decisions about what products to offer to particular customers online need to be taken instantaneously if one is to react to actual customer behavior, and the resulting feedback loop must operate in real time. On another axis, these two decisions are also at opposite extremes: the first is human, the second fully automated.

Although most customers are only at an early stage in this thinking, what can be seen emerging is a new way to position decision making as part of the broader business context. For example, many customers use IBM DB2 Alphablox precisely in this way to embed decision-support activities directly into their day-to-day operational intranet applications. In addition, we increasingly see decision support being included in business process development. In this view, a decision-making activity is simply another step in the overall business process flow.

Having made the leap to recognize business intelligence as increasingly no more than a part of a business process, we can now clarify what is meant when people say real-time or active data warehousing. Simply put, they mean decisions based on current (real-time or active), consistent (data warehousing) information. But, in today's IT environment, this poses a dilemma, because current data resides in the operational systems whereas consistent data is the realm of data warehousing. Prior attempts to combine the needs for current and consistent data led to the concept of the ODS. The question therefore arises: to what extent does the design purpose of an "active data warehouse" differ from that of an ODS?

To pose the question differently: Why would one try to address business requirements that span the traditional operational and informational environments with a solution whose original design point was purely informational? We assert that the answer lies in the combination of strengths of the two environments rather than in an attempt to make one into the other.

Fortunately, current trends and directions in thinking about the operational environment lead in precisely the same direction. The emergence of the service-oriented architecture (SOA) and the enterprise service bus (ESB) points to a radical restructuring of the operational environment. Although driven primarily by the need for flexibility and adaptability in the operational environment, the approach offers much to the informational world. Both SOA and ESB demand a comprehensive and internally consistent set of semantics describing data and function, as well as consistency of data content, if the services are to interoperate as envisioned. Significant effort is being expended to ensure this occurs, as can be seen in tools such as IBM WebSphere Business Integration Modeler and the IBM Rational® suite of products. Consistent process and data semantics are, of course, the foundation for data consistency in the informational world.

Furthermore, the communication and messaging core of the ESB is also emerging in products such as IBM WebSphere MQ and IBM WebSphere Business Integration Server. Customers today are beginning to look to this type of functionality to support their data warehouses. Clearly, widespread and reliable messaging in the operational environment can also be extended to the informational space. The feedback loop is then closed through the use of DB2 UDB triggers to detect and capture events in the warehouse environment and return relevant information to the operational systems via the same queues. Given the ability of WebSphere Information Integrator to access messages directly on the queues as well as its increasing use of WebSphere Business Integration adapters, we can envision a possible basis for a single communication infrastructure within and between both worlds.

A number of large retail and telecommunications organizations are already actively engaged in creating such environments. Initial implementations have focused on unidirectional data flows, but bi-directional flows are also beginning to emerge. A common characteristic of both industries is enormous volumes of transactions that are continuously generated—point-of-sale records for retailers and call data records for telecommunication operators. Continually collecting these records, shipping them via messaging infrastructures and applying them to the warehouse enables rapid response to changes in the business, while triggers in the warehouse allow specific conditions to be detected rapidly and actions to be automatically fed back to the operational environment.

Reinventing the data warehouse

As seen from the previous sections, a number of interesting trends are beginning to emerge. In the classical data warehouse environment, we see customers actively consolidating data warehouses, marts and ODSs into a single multi-floored database image as technology advances deliver better performance in more mixed processing environments.

As users demand access to more extensive and varied data types for decision making, information integration technology is allowing the development of a first generation of distributed, federated queries and applications that reliably access and use data from diverse sources. Meanwhile, mixed mode applications such as customer relationship management and business activity monitoring are blurring the boundaries between the operational and informational environments. In addition, the emerging integration and process-oriented movement in the operational world is beginning to address issues of data timing and consistency described earlier that have long been the bane of data warehousing.

Together, these trends signal a time of significant change, not only in data warehousing but in the IT environment as a whole. As a result, the coming years will likely see the reinvention of the data warehousing concept along with the possible disappearance of a number of components typically seen in

today's warehouses. Some of these changes will be dependent on substantial reengineering or replacement of the more archaic operational systems of today, and may thus be delayed. However, a redefinition of data warehousing looks increasingly likely in the short to medium term.

The initial phase will see a gradual reduction in the number and variety of data marts and data warehouses. Driven by the increasing power of database engines, advances in information integration and the need to reduce ongoing maintenance costs, companies will be able to reduce redundancy in the informational environment. While redundancy will not be totally eliminated for organizational, geographical, backup and other reasons, the trend points clearly to larger and more centralized data warehouses and fewer, often logical, data marts.

In parallel, one can envisage the core data warehouse itself slimming down. Over the years, the data warehouse has become bloated with information that is peripheral to its true purpose. That purpose—the maintenance of a long-term memory of the business—is, of course, as valid as ever. In fact, the new focus on compliance and the need for a single, agreed and true history of the business makes the core data warehouse even more necessary than in the past. However, businesses have added much more data to their warehouses over the years to support processes that are more operational than informational in nature. The increased flexibility of information integration platforms using a variety of technologies to access data across multiple systems will allow the removal of this extraneous data from the warehouse.

But what about the other, often hidden, purpose of the warehouse and one that it shares with the ODS—cleansing and reconciliation of data from the operational environment? These requirements spring solely from the disintegrated nature of today's operational environment. While integrating operational applications continues to be a significant challenge, constructing the on demand business is wholly dependent on solving this issue.

We see an increasing focus on application integration and the underlying need for clean, semantically and temporally consistent data in the operational world. While this may be a rather slow evolution, the implications for the data warehouse and ODS are clear. Cleansing data as it enters the informational environment will eventually become unnecessary. Reconciliation of data becomes simple and straightforward enough to allow the vast majority of it to occur during query execution. At that stage, the ODS becomes an endangered species and the equivalent function in the warehouse should disappear.

These trends also have some substantial implications for ETL. Post-operational cleansing and reconciliation has been a major requirement for ETL tools and, as we have seen, this need is being reduced as operational systems need to do this for themselves to support the on demand business. Furthermore, as discussed earlier, EAI in combination with information integration techniques creates an infrastructure in which data can be consistently and correctly passed in real time between applications. Given that this data is increasingly clean, well-described and reconciled, there is a compelling argument that this same infrastructure should be the basis for feeding data into the historical warehouse, whether on a record-by-record basis or in small, defined batches. ETL thus becomes little more than an extension of EAI and information integration. Such an approach reduces duplicate function and eliminates another source of possible inconsistency.

Finally, as the business process view comes more pervasive, including in its scope activities considered both operational and informational, it is inevitable that the boundaries between operational applications and the data warehouse will become increasingly blurred for end users. Again, the implications are clear. Business users will demand a single, consistent interface to the information and function they require; the data warehouse will disappear as a distinct element in the minds of the users.

From the IT point of view, there will be increasing pressure for the warehouse to be more closely integrated with the operational environment as the business process view expands and as operational and informational data become

more closely aligned. While it is unlikely in the medium term that the data warehouse and operational databases will collapse to the nirvana of a single copy of the data for all purposes, the distinct and independent existence of the data warehouse in contrast to the operational environment seems likely to disappear.

Conclusion

The data warehousing environment is currently undergoing more change than it has seen in the past five years. Improvements in database technology and information integration techniques have allowed customers to begin to make fundamental changes in the logical and physical architecture of their data environments.

Redundancy of data is being attacked on two fronts. On one side, data warehouses and data marts are being collapsed into more efficient and more easily managed repositories. On the other, information integration is providing federated access across diverse data stores and reducing the need in certain cases to create additional explicit copies of data in the first place.

In parallel, the adoption of on demand business is driving new levels of integration and consolidation in the operational environment. Over time, these changes will further simplify the preparation of information for use in decision support and business intelligence. Meanwhile, the increasing emphasis on a process view of the entire business is beginning to blur the boundaries between operational and informational activities.

The data warehouse will become what it always should have been: nothing more than the repository of historical information of the business, an adjunct to the operational databases for no longer current data and, for the users, simply one more place to go—unbeknownst to them—for the data they need to run and manage their business.

IBM Software Group
Route 100
Somers, NY 10589
U.S.A.

Produced in the United States
January 2005

GC34-2486-00