“Achieving Century Uptimes”
An Informational Series on Enterprise Computing

As Seen in *The Connection*, A Connect Publication
December 2006 – Present

About the Authors:

Dr. Bill Highleyman, Paul J. Holenstein, and Dr. Bruce Holenstein, have a combined experience of over 90 years in the implementation of fault-tolerant, highly available computing systems. This experience ranges from the early days of custom redundant systems to today’s fault-tolerant offerings from HP (NonStop) and Stratus.

***Gravic, Inc*.**
Shadowbase Products Group
301 Lindenwood Drive, Suite 100
Malvern, PA 19355
610-647-6250
http://www.gravic.com/shadowbase

Achieving Century Uptimes

Part 14: The Evolution of Real-Time Business Intelligence

January/February 2009

Dr. Bill Highleyman
Dr. Bruce Holenstein
Paul J. Holenstein

In today's competitive environment with high consumer expectation, decisions that are based on the most current data available are what will improve customer relationships, will increase revenue, and will maximize operational efficiencies. For instance, a store wants to know that a customer who is purchasing a set of drill bits bought a battery-operated power drill the month before. While the customer is at the register, the store offers him a 30% discount on an extra drill battery.

This is the technology of *real-time business intelligence,* sometimes referred to as event-driven business intelligence. Real-time business-intelligence systems provide the information necessary to strategically improve an enterprise's processes as well as to take tactical advantage of events as they occur.

In this article, we track business intelligence from its early days as a strategic tool to today's real-time capabilities for tactically responding to events as they happen.

What is Business Intelligence?

Business intelligence is the use of an organization's disparate data to provide meaningful information and analyses to employees, customers, suppliers, and partners for more efficient and effective decision-making. It does this by transforming information into actionable strategies and tactics to improve the efficiency of the enterprise, to reduce costs, to attract and retain customers, to improve sales, and to provide many other significant benefits.

Some typical examples of the use of business intelligence include retail sales patterns and campaign management, credit-card fraud detection, inventory analysis, telecom service-usage analysis, asset management, and corporate profit and loss analysis.

The Data Warehouse

Early business applications each had their own databases. These databases were "islands of information" in that no other systems had access to them. Businesses soon recognized the analytical value of the collective data that they had available in these databases. However, combining this data for analysis was a challenge because of the incompatibilities between systems. What was needed was an infrastructure for data exchange, collection and analysis that could provide a unified view of an enterprise's data. The data warehouse evolved to fill this need.

A data warehouse is a single system that is the repository of all of an organization's data in a form that can be effectively analyzed to support strategic decision making.

However, meeting this goal presents several very significant challenges:

- Data has to be acquired from a variety of incompatible systems.

- The same item of information might reside in the databases of different systems in different forms. A particular data item might not only be represented in different formats, but the values of this data item might be different in different databases. Which value is the correct one?

- Data is continually changing. How often should the data warehouse be updated to reflect a reasonably current view?

- The amount of data is massive. How can it be analyzed, reduced, and presented simply so that it is useful?

To meet these needs, a broad range of powerful tools have been developed over the years and have become productized. They include:

- Extract, Transform, and Load (ETL) utilities for the moving of data from the various data sources to the common data warehouse.

- Data-mining engines for complex predetermined analyses and ad hoc queries of the enterprise data stored in the data warehouse.

- Reporting tools to provide management and knowledge workers with the results of the analyses in easy-to-absorb formats (digital dashboards being a predominant example).

Early on, the one common interface that could be provided between the disparate systems in an organization was magnetic tape. Tape formats were standardized, and any system could write tapes that could be read by other systems. Therefore, the first data warehouses were fed by magnetic tapes prepared by the various systems within the organization.

However, that left the problem of data disparity. The data written by the different systems reflected their native data organizations. The data written to tape by one system often bore little resemblance to similar data written by another system.

This led to the development of powerful ETL utilities. An ETL utility extracts data from a source database (usually via magnetic tape) and loads it into an intermediate database. There it is transformed to the common format of the data warehouse (which could be quite different) and then loaded into the warehouse's database.

We call this process *offline ETL* because the target database is not being continuously updated. It is updated on a periodic batch basis. Though offline ETL serves its purpose well, it has some serious drawbacks:

- The data in the data warehouse is *stale*. It could be weeks old. Therefore, it is useful for strategic functions but is not particularly adaptable to tactical uses.

- The source database typically has to be quiesced during the extract process. Otherwise, the target database will be in an inconsistent state following the load. This means that the applications have to be shut down, often for hours.

Therefore, though it serves well the strategic needs of an organization, it does little to support day-to-day tactical needs, much less real-time needs.

Enterprise Application Integration (EAI)

To more appropriately meet tactical needs, another solution for integrating enterprise-wide data appeared. This was *enterprise application integration (EAI).*

In many cases, the functioning of one enterprise system can be significantly enhanced if it can request data or can immediately receive newly generated data from other systems. For instance, an inventory-control system can be much more responsive if it has immediate access to point-of-sale data so that it can monitor product movement in real time. This capability requires that various legacy enterprise systems with different hardware, operating systems, and databases be enhanced so that they can talk to each other and can exchange information in real time or near-real time using a common data format.

There are three primary techniques for implementing EAI – adapters, message-oriented middleware, and data replication. *Adapters* are specialized interfaces between diverse applications that allow these applications to directly communicate. *Message-oriented middleware (MOM)* provides a mechanism for an application to asynchronously send messages to other applications via message queues. *Data-replication engines* exchange data between systems at the database level.

However, adapters and MOM require changes to the application programs to determine the information to be exchanged and to convert that information into a common format. Often, this means modifying third-party applications, which can at times only be done by the application vendor. Additionally, these approaches mean that all systems, applications, and the network must be up and available at all times, dramatically increasing the availability requirements of each; otherwise, interactive requests between the systems will go unanswered when needed. These issues create a serious roadblock to EAI.

Data replication, on the other hand, works not at the application level but rather at the database level. Based on rules incorporated into the replication engine, certain database changes are replicated to other systems in the application network. These changes are placed in tables in the target systems and can be used by new applications implementing new functionality for the enterprise. Because there is no need to modify source applications and the data to be shared is locally available when and where needed regardless of the state of the network or other systems, data replication has become the method of choice for EAI.

EAI allows applications to be integrated on a real-time basis as events occur and as databases are updated without having to wait hours, days or weeks for data from a data warehouse. Consequently, EAI supports tactical decision making very well. However, it does not immediately provide the historical data needed for strategic decision making.

Operational Business Intelligence

Operational Business Intelligence (*OBI*) systems provide an intermediate step toward satisfying the strategic needs that data warehouses address as well as the tactical decision making that EAI addresses. An OBI system provides an event database that is frequently updated via minibatches, perhaps on an hourly basis. As an historical database, events can be summarized and can satisfy strategic requirements. With frequent updates, strategic decision making can be extended to daily or intra-day information that can be used to take operational action to address an immediate problem.

OBI systems bring together two needs for business intelligence – a historical database for strategic analysis and the capability to make rapid suggestions for operational actions. What is required next is the capability to make real-time suggestions for actions that can be taken immediately upon the occurrence of some specified event. This is *real-time business intelligence*.

Real-Time Business Intelligence

There are two primary impediments to effective and efficient real-time business intelligence: data latency and data unavailability.

Data Latency refers to the *staleness* of data. The value of data degrades rapidly with its age. When people are relying on real-time business intelligence to tactically help them with on-the-spot decisions, the freshest data and the fastest response times are needed.

Data Unavailability is a death knell for real-time business intelligence. If a company's operations have progressed to the point that they are dependent on real-time business intelligence, the unavailability of this intelligence due to a failed system can bring operations to a halt. Extreme availability of the real-time business intelligence services is paramount.

A business cannot respond to events as they happen if it cannot find out about these events for hours, days, or weeks. It also cannot immediately respond to events if the system that supplies the analyses of these events is down.

What is needed is an OBI system that can respond to events in seconds or less. However, this cannot be done by updating the OBI database with hourly minibatches. Rather, the database must be updated with transaction activity in real time as it occurs. We call this *trickle-feeding* the database. As transactions are received, they are applied and become a growing historical record of activity.

Furthermore, there must be a very fast *rules engine* that can analyze incoming transactions against the historical database and can make decisions quickly enough so that immediate action of value to the enterprise can be taken. This is real-time business intelligence (RTBI).

Online ETL

What is needed for real-time business intelligence is an *online* ETL facility that not only can perform the offline ETL function of preserving historical strategic data but that can also provide the EAI function of providing current tactical data. The online ETL's job is to create and maintain a synchronized copy of a source database on a target database (the RTBI system) *while* the source database and the target database are being actively updated and are being used by multiple applications. In effect, as transactions occur in the enterprise, they are trickle-fed to the RTBI system in real time in such a way that this activity is transparent to other ongoing operations.

As with EAI, data replication can provide this function without being invasive to the enterprise's applications. Data-replication engines can transform and feed data in real time to a central database for both strategic analysis and immediate tactical decision making. This database is known as an *operational data store (ODS).*

The Operational Data Store

One special characteristic of an ODS is the requirement for it to handle mixed workloads. On the one hand, it must be able to respond to complex queries from knowledge users, data-mining facilities, and rules engines. This is the realm of Online Analytical Processing (OLAP). The database structures suitable for OLAP are characterized by *fat keys* that allow rapid searching of the database to respond to complex queries.

On the other hand, the ODS must be capable of processing an extremely high transaction rate, as it is being fed transactions in real time from many enterprise systems. This is the realm of Online Transaction Processing (OLTP). The database structures suitable for OLTP are characterized by *skinny keys* that require a minimum of updating as data is added to the database.

Another special characteristic of an ODS is that it is can be implemented to be bidirectional. Unlike a data warehouse, which only accepts information from enterprise systems, an ODS can both accept information from and deliver information to the other enterprise systems. One example of this is the act of keeping databases in synchronization. A particular data item, like a customer's address, may be stored in several application databases around the enterprise. If one system changes this data item, it is the ODS acting as a central data repository that informs the other systems of the new data value so that they can update their databases.

Another example of outgoing information is the results of the rules engine. If the rules engine decides to recommend a particular immediate action, that action is communicated to the appropriate enterprise system for execution. For instance, if the rules engine for a credit-card processor detects suspicious activity, it can immediately alert the authorization system to take appropriate action.

In concept, the ODS, which contains all corporate data, could become the *database of record* – the single version of truth – for the enterprise. However, this generally has not happened because of regulatory requirements and other considerations. The database of record remains where it has been resident for decades – on the legacy systems that provide the source information for the ODS.

Extreme Availability

Once an enterprise gets wedded to real-time business information, it will suffer gravely should it lose this capability. Instant reactions that have made it competitive and efficient are suddenly lost. Therefore, extreme availability of the real-time business information system is of paramount importance.

The first step is to choose an architecture that is especially resilient to failure. NonStop systems from HP are an ideal solution. However, though these systems give reasonable protection against single-component failures, they do nothing for disasters that can take out an entire data center. Therefore, the RTBI system must be backed up by a geographically remote site that can quickly take over in the event of a primary site failure.

The best backup for an RTBI system is an active/active system.[1] An active/active system comprises two or more geographically-dispersed nodes that are already up and running. Each node can be actively processing and sharing the application load with the other nodes. Should a node fail, all that needs to be done is to switch transactions (or users) from the failed node to the surviving nodes, a switch that can be done in seconds. With an active/active system, in the event of a node failure, there is no failover. There is only resubmission of transactions.

Summary

It was noted earlier that there are two impediments to real-time business intelligence – data latency and data unavailability. Online ETL using bidirectional data replication solves the problem of data latency. An active/active ODS solves the problem of data unavailability.

Real-time business intelligence is available today, and many pioneering enterprises are taking competitive advantage of this exciting technology.

Gravic, Inc. Shadowbase Products Group
301 Lindenwood Drive, Suite 100
Malvern, PA 19355 610-647-6250
http://www.gravic.com/shadowbase

---

[1] What is Active/Active?, *Availability Digest*; October, 2006.
Breaking the Availability Barrier: Survivable Systems for Enterprise Computing, AuthorHouse; 2004.
Breaking the Availability Barrier Volume 2: Achieving Century Uptimes with Active/Active Systems, Authorhouse; 2007.
Breaking the Availability Barrier Volume 3: Active/Active Systems in Practice; Authorhouse, 2007.