

Part 6 — Real-Time Business Intelligence

Yesterday's Data is No Longer Sufficient

Use of an online data warehouse solves a strategic need of the enterprise because it manipulates massive amounts of data via data mining to derive new information and knowledge of an enterprise's operations. However, online data warehousing has little tactical value since the data in it is generally quite stale and can be days or weeks old. Data warehousing's primary value is in supporting strategic goals such as reducing costs, increasing sales, and improving profits. On the other hand, enterprise application integration (EAI) solves a tactical need of the enterprise since it allows systems to react immediately to events generated by other systems; it has little strategic value since it provides no repository of data suitable for data mining.

A solution is needed that satisfies both the strategic and the tactical needs of an enterprise. A store, for instance, needs to know that men who purchase diapers on Saturday also tend to buy beer at the same time. A strategic way to capitalize on this information would be to put the beer near the diapers, and have beer sales on Saturday. A credit card company needs to know that a credit card being used to purchase an item in New York City was used thirty minutes earlier in Amsterdam. A tactical way to capitalize on this information would be to deny the transaction, and place a hold on the card. A solution enabling this type of capability is known as real-time business intelligence (RTBI), or "fast data."

Another example of the use of RTBI is in customer support. For instance, a customer often needs to know if he has enough credit or cash on his credit card or in his debit card account before attempting to make a purchase. Some card companies allow a customer to get preauthorization for a specified amount via cell phone to ensure that the purchase is covered, meaning that the card company must have up-to-date balances for its customers.

Today's Business Intelligence Needs

The gap between analytical and operational processing is closing fast. Just as Moore's Law continues to characterize the rapid pace of technology development, the complex data-mining queries that used to take hours to run now execute in seconds. If only these data-mining engines had the latest values of the data, the tactical and strategic needs of business intelligence could be merged into a single solution. 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 RTBI to tactically help them with on-the-spot decisions (real-time decision support), the freshest data and the fastest response times are needed.

Data unavailability is a death knell for businesses. Business operations have progressed to the point that they are dependent on RTBI, the unavailability of this intelligence due to a failed system could bring operations to a halt. Continuous availability of the RTBI 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 (data latency). It also cannot immediately respond to events if the system that supplies the analyses of these events is down (data availability). If the problems of data latency and data availability are solved, businesses can react proactively to new information and knowledge rather than reactively. These problems are solved by sophisticated data replication engines such as HPE Shadowbase.

Real-Time Business Intelligence Systems

Operational Business Intelligence (OBI) systems represent a significant improvement in reducing data latency and enabling actions to be taken within hours of the events that triggered them; however they do not meet the criterion of immediacy that will allow a business to react in real-time to an event. For instance, an OBI system does not generate an offer to a customer while he is checking out at a cash register. Nor does it deny a potentially fraudulent transaction before it is executed.

An OBI system is needed that responds to events in seconds or less; but this response is not done by updating the OBI database with hourly mini-batches. Rather, the database must be updated with transaction activity in real time as it occurs or real-time change data capture (CDC); this type of update is called trickle-feeding the database. As transactions are received, they are stored and become a growing historical record of activity.

There must also be a very fast rules engine that analyzes incoming transactions against the historical database and makes decisions quickly enough so that immediate action of value to the enterprise is taken; this action is RTBI. There is a distinction between a data-mining engine and a rules engine. Both require an historical database; but a data-mining engine looks for relationships in the historical data to reactively support decision-making after the fact. A rules engine compares a real-time event to the historical data to proactively suggest an action to be taken.

The fraud-detection OBI system in can be extended into a real-time fraud detection system (Figure 1). In this figure, the reacts fast enough to cause a suspicious transaction to be denied before it is consummated. Real-time ATM/point-of-sale (POS) transactions are fed to an RTBI system which then posts each transaction to its database for a rules engine to analyze. In real-time, the rules engine checks this transaction against recent activity on the credit or debit card and makes an instant determination of suspicious activity. It then generates a message indicating whether the transaction should be accepted or denied.

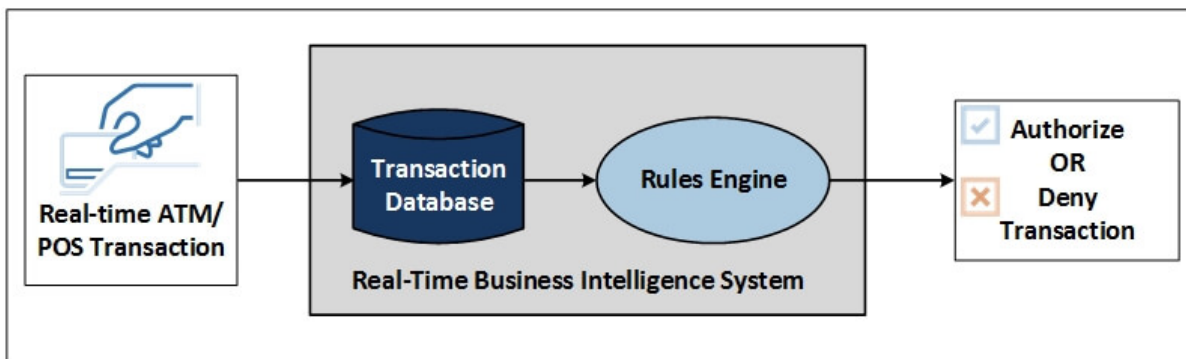


Figure 1 — A Fraud Detection RTBI System

Figure 2 shows an example of this particular fraud detection architecture. ATM and POS machines are generally serviced by a particular acquirer (bank), with the credit or debit card that was used typically being issued by a different bank. The card transaction must be sent to the issuing bank for authorization, and verification that the card balance is sufficient.

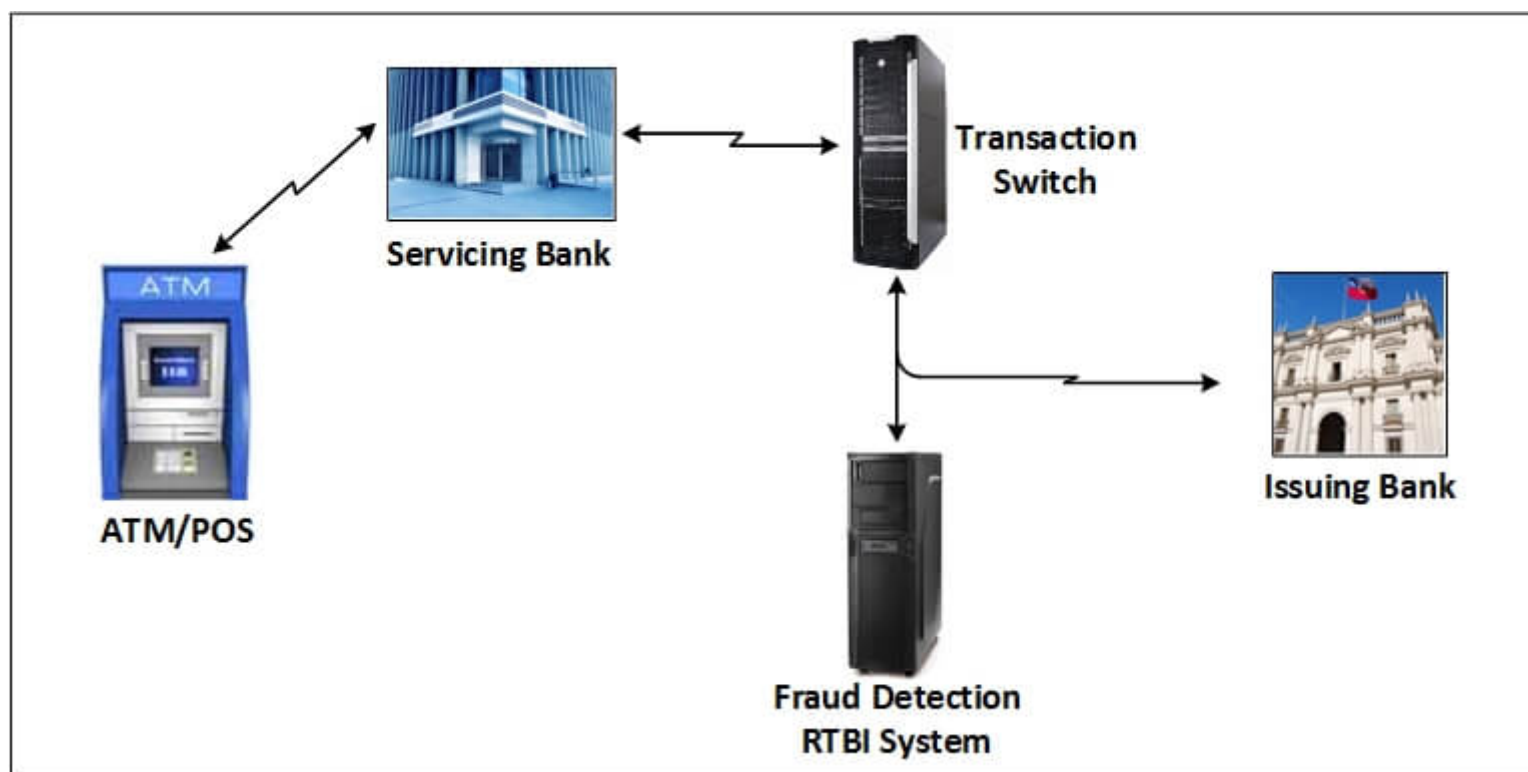


Figure 2 — Fraud Detection

This particular fraud detection example is actually used in real life, as shown in Figure 2. ATM and POS machines are generally serviced by some particular bank, but the credit or debit card used was probably issued by some other bank. The card transaction must be sent to the issuing bank for authorization, and verification that the card balance is sufficient, and if so, the bank authorizes the transaction.

In this case study, a bank service company operates a transaction switch that receives card transactions from the servicing banks operating the ATM or POS machines and forwards these transactions to the appropriate issuing banks. Upon receipt of an accept/deny message from the

issuing bank, the transaction switch returns this message to the servicing bank, which then takes appropriate action to accept or deny the transaction.

Additionally, the bank service company provides a fraud detection service. At the same time that it forwards the transaction to the issuing bank for authorization, it also sends the transaction to its RTBI fraud detection system. If this system determines that the transaction is suspicious, the bank service company takes several actions as requested by the issuing bank. It certainly denies the transaction and informs the issuing bank, but it may also alert the issuing bank to the circumstances of the suspicious activity so that it can decide whether to put a hold on the card. This system shows RTBI in action as the RTBI fraud detection system uses complex rules against an historical database to determine which actions to take that directly affect a transaction in progress.

RTBI Dashboards

RTBI dashboards are used to bridge the gap between operational business intelligence and real-time business intelligence. For instance, Figure 3 shows an IT server network-monitoring dashboard, which displays not only historical information but also shows the current status of the server network. The dashboard is interesting because it performs all three business intelligence functions: strategic, operational, and tactical.



Figure 3 — IT Server Network-Monitoring Dashboard

From a strategic viewpoint, it shows bandwidth usage and connection quality over the last several weeks, and by clicking on the History tab, further historical usage and quality data are displayed. This dashboard forms the basis for future planning of network upgrades. From an operational viewpoint, the dashboard shows the current status of the network, the current memory usage, and the current connections and their traffic. If a network problem occurs, operational staff could immediately take remedial action. From a tactical viewpoint, as problems are detected, the RTBI system driving the dashboard can issue audible, email, or cell phone alerts to the operations staff. The RTBI system even takes automatic remedial action such as rerouting network connections, shedding low-priority traffic, or invoking redundant connections. The transactions are fed to the new RTBI system from the enterprise's other systems with the online **extract, transform, and load (ETL)**.

Online ETL

4/8/2019 Part 6: Real Time Business Intelligence - Shadowbase
ETL is the facility that allows data to be extracted from a source database, transformed into a common format, and loaded into a target database (the data warehouse's database). Since contemporary ETL facilities are batch-oriented and run periodically, they are characterized as being offline ETL facilities. EAI exchanges current information between systems in an application network but provides no historical record of enterprise activity for strategic-analysis purposes.

RTBI needs an online ETL facility that not only preserves historical strategic data but can also provide 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 actively updated and used by multiple applications. In effect, as transactions occur in the enterprise, they are trickle-fed to the RTBI system in such a way that this activity is transparent to other ongoing operations. As with EAI, three methods can be used to create an online ETL facility – connecting via adapters, using message-oriented middleware, and synchronizing via low-latency replication engines.

Adapters

Early adaptations of RTBI used an extension of existing EAI technology. Adapters were used to interconnect enterprise systems with the RTBI system. As transactions were executed by an external system, the results of those transactions were communicated to the RTBI system via adapter connections. The adapters also serviced requests from external systems and returned the RTBI system replies to those systems.

However, adapter technology suffered from the same problems that it faced in EAI applications. It was invasive and often required application modification in order for the applications to interface with the adapters. In addition, adapters were specialized to the applications. Each adapter knew the proprietary formats of the application data structures and how to interface to its application and was thus custom designed for that application. Every time the application changed, the adapter was modified, so consequently, not all applications could participate in the online ETL function.

Message-Oriented Middleware (MOM)

Message-oriented middleware (MOM) also exhibited many of the same problems that plagued adapters. MOM was invasive and required changes to the application to send and receive appropriate messages in a common interconnect data format, or to use adapters. In order to make application changes, access to the application's source code was required, but this source code was often lost or was the proprietary property of a third-party vendor, and so was not available. Additionally, every time the application changed, the MOM code potentially had to be modified. As with the EAI implementations, adapters and MOM RTBI implementations also suffered from the problems related to network interconnect issues. Due to the immediate nature of RTBI, if the interconnect went down, the RTBI response was either delayed, or worse, failed altogether.

Data Replication

Data replication is synchronous (data is replicated to the target system concurrent with the application data changes) or asynchronous (data is replicated to the target system some very short time after the application has made its changes). In RTBI applications, asynchronous replication is generally used, allowing the replication activity to be totally transparent to the application. The application proceeds with no knowledge of or impact from the replication activity and the application's database activity is extracted by the replication engine via a transaction log, triggers, or intercepts. Selected updates are sent to the RTBI target system by the replication engine, where they are applied. The replication latency interval, which is the time that it takes to propagate a source database change to the target database, is measured generally in sub-seconds.

Data replication solves the adapter and MOM problems of application invasiveness and specialization, as well as data availability, in an RTBI environment. Because a data replication engine is non-invasive and is application-unaware, it only deals with the database, and is isolated from the application by the database. Today's replication engines support most relational databases and many non-relational databases as well. An additional benefit is that the same data replication engine, such as HPE Shadowbase, can also serve other purposes at the same time as fulfilling the RTBI mission, for example, providing business continuity and disaster recovery.

Figure 4 depicts data consolidation. An application makes changes to a source database, which are captured in a transaction log and then triggers the replication engine to transform and send the change data capture (CDC) event information into an RTBI database.

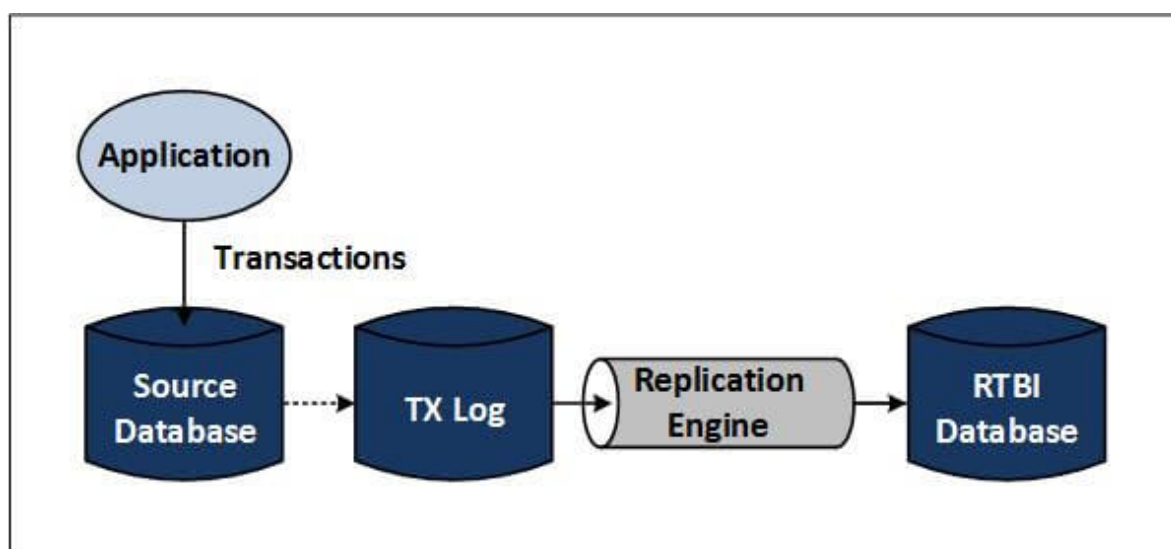


Figure 4 — Data Consolidation

Replication engines support rules for data transformation. Since some rules are built in, and others are specified by user-supplied routines, data transformations of any kind are possible without modifying the applications, the databases, or the core replication engine.

Since replication engines are required to preserve transactional consistency, source updates are not randomly applied to the target database. Transactions must be applied in the same order to the target database that they were made at the source database.

4/6/2019 Part 6: Real Time Business Intelligence: Shadowbase
An additional problem, called a data collision, occurs when multiple sources simultaneously update their copy of the same data item. For instance, what happens if two enterprise systems update the same customer address with different data at the same time? Which one is correct? Data collisions may happen with any asynchronous mechanism used to feed the RTBI system, whether it is replication, adapters, or some other form of messaging. Data collisions do not occur if synchronous replication is used. Most replication engines are particularly adept at handling collisions, detecting and resolving a collision via specified rules. Some of these rules are built in, and more application-dependent rules are added via user-supplied routines.

Another problem can occur if a target system fails, and during its downtime, other external applications try to send it data. If an application does not get a response to its sent data, the application may fail or its data may be lost. However, transactional replication engines queue this data and will send the data updates to the system when it is returned to service. As discussed below, in the unlikely event of an RTBI system failure, the replication engines in each enterprise system queue their data changes and send them to the RTBI system when it is returned to service, and no valuable business data is lost.

A similar problem can occur when a communication connection is lost. With data replication, a system that loses its connection with an RTBI system continues to operate in so-called “split-brain mode.” It is unaffected except that it does not send its data updates to the RTBI system, and it does not receive updates or recommended actions from the RTBI system. When the connection is restored, all of the updates that accumulated in either direction are sent to the opposite system via the queues maintained by the replication engines. In split-brain mode, each system applies updates to its local copy of the data invisibly to the other system while the connection is down; so there are bound to be data collisions. These collisions are resolved on-the-fly by the replication engines after recovery of the network and during the resynchronization of the databases.

Online Copying

Another requirement for online ETL is an online copy utility, which is needed to bring an RTBI system into operation. An up-to-date snapshot of the data in the various enterprise systems that will feed the RTBI system must first be loaded into the RTBI system before it becomes effective. This load must occur without affecting the source systems since they are busy running the enterprise and must include all the various changes that occur during the copy, which could take hours or even days to complete.

Additionally, the copy must include all of the transformations that were otherwise made by the online ETL facility in order that the initial RTBI database properly reflects the state of the enterprise. The HPE Shadowbase SOLV product (built by Gravic, sold by HPE) meets all these requirements.

Extreme Availability

Once an enterprise activates RTBI, it would suffer greatly if it lost this capability. Instant reactions that made it competitive and efficient would be suddenly lost. Continuous availability of the RTBI system is of paramount importance.

The first step is to choose an architecture that is especially resilient to failure. NonStop systems from HPE are an ideal solution. These totally redundant systems provide proven availabilities in excess of four 9s (99.99% availability, equating to approximately 52 minutes and 32 seconds of downtime per year). Clusters of highly reliable industry-standard servers using redundant array of independent disks (RAID) or mirroring storage are another choice and are configured to provide availabilities in the same range as HPE NonStop systems, but are much more complex to manage and generally have a higher total cost of ownership (TCO).

Though these solutions give reasonable protection against single component failures, they do nothing for disasters that take out an entire data center. The RTBI system must be backed up by a geographically remote site that takes over in the event of a primary site failure; this is called a *fault tolerant* solution. Otherwise, it might take days, weeks, or even longer to replace the system, during which time normal business operations are severely impacted.

Backup systems using magnetic tape to rebuild the database take days to recover and are unsuitable for RTBI system backup. Virtual tape eliminates some of the problems associated with magnetic tape, but virtual tape systems still take a prohibitively long time to recover. Data replication to a backup site provides a reasonably complete copy of the database, but following a primary site failure, the database must still be brought to a state of consistency, the applications started, the database opened, and the system tested before it is returned to operation. This strategy takes some time (minutes to hours) and is hindered by the same problem that the other backup strategies face, which poses the question: will the backup system properly come up when it is brought into operation?

The best “backup” for an RTBI system is a continuously available, active/active system, which provides for continuous availability. An active/active system comprises two or more geographically-dispersed nodes that are already up and running, with each node actively processing and sharing the application load with the other nodes. If a node fails, transactions (or users) must be switched from the failed node to the surviving nodes, a switch that can be done in seconds.

The primary advantages of having an RTBI active backup are twofold. First, failover is accomplished in seconds. Users of the RTBI facilities may not even know that a failure has occurred. Second, the failover process itself does not fail. Since the backup system is already up and running, it is known that it is fully operational, and its operation is being verified with every transaction it processes.

For some applications or other operational reasons an active/active RTBI configuration may not be possible. An alternative configuration that may be used in such cases is known as sizzling-hot-takeover (SZT), also known as sizzling-hot-standby. The only difference between this alternative and

an active/active configuration is that the standby system is not actually processing transactions, in all other respects it is identical, and shares all the advantages of an active/active system.

As previously stated, there are two impediments to real-time business intelligence, data latency and data unavailability. Online ETL (and especially data replication) solves the problem of data latency. Active/active or Sizzling-Hot-Takeover (SZT) RTBI systems solve the problem of data unavailability.

Related Business Intelligence Information:

- **Real-Time Business Intelligence Overview**
 - **Part 1 – The Evolution of Real-Time Business Intelligence**
 - **Part 2 – What is Business Intelligence?**
 - **Part 3 – The History of Business Intelligence**
 - **Part 4 – Enterprise Application Integration (EAI)**
 - **Part 5 – Operational Business Intelligence**
 - **Part 7 – The Operational Store (ODS): The Next Evolutionary Step**
 - **Part 8 – HPE Shadowbase Real-Time Business Information Solutions**
 - **Real-Time Business Intelligence Summary**

Related White Paper:

- **The Evolution of Real-Time Business Intelligence**

Related Case Study:

- **Real-Time Credit and Debit Card Fraud, an HPE Shadowbase Real-Time Business Intelligence Solution**

Related Use Case:

- **Prevent Fraudulent Activity for Online Banking**

Mailing Address

17 General Warren Blvd. shadowbase@gravic.com
Malvern, PA
19355-1245 USA

Contact Us

Shadowbase Sales

+1.610.647.6250

Shadowbase Support

+1.610.647.8180