the Availability Digest

www.availabilitydigest.com
@availabilitydig

Stock Exchange Speeds Clearing with Data Replication

June 2014
Paul J. Holenstein
Executive Vice President
Gravic, Inc.

A regional South American stock exchange realized that it could not reliably achieve same-day securities settlement-commitment times due to the manual reporting of trades to its clearinghouse and the often ensuing data-entry errors. It decided to re-architect its interaction with the clearinghouse to make the process fully automatic. This enhancement eliminated erroneous manual input and enabled the exchange to improve its settlement times.

The re-architected system required heterogeneous data replication between the exchange's HP NonStop trading system and its clearinghouse's AIX/Sybase system. Data replication now plays a major role in integrating the many heterogeneous systems in the exchange's new IT infrastructure as well as providing continuous availability for its mission-critical business services.

The Stock Exchange

The stock exchange is the result of the 2001 merger of three of the country's regional stock exchanges, one of which dates back to the late 1920s. The exchange operates a trading platform for equities, fixed income bonds, and standardized derivatives using an HP NonStop server pair in an active/standby configuration for the purpose of disaster recovery. In addition to its trading functions, the exchange provides equities and fixed income clearing and settlement services, market data, and electronic access to accounts. The exchange lists 89 corporations with a combined market capitalization of USD $200 billion.

The Clearinghouse

To provide clearing and settlement services, the exchange partners with a clearinghouse that acts as a security depository. A security depository maintains all of the stocks and bonds of its account holders electronically so that ownership can be transferred easily via a book entry rather than via a physical transfer of certificates. The clearinghouse provides the clearing and settlement functions per the settlement cycles mandated in the settlement agreements.

At the end of the trading day, the clearinghouse determines which members are due to deliver funds or securities and which members are due to receive funds or securities by the settlement date. Each trade is assigned to a buyer and a seller, and any funds and securities being transferred between them must match. Settlement is a two-way process that involves the transfer of funds and securities on the settlement date between the trading parties.

The clearinghouse does not transfer paper stock certificates or bonds, though paper records of account holders' assets are prepared. Rather, the record of securities being held by account holders is maintained in the clearinghouse's Security Depository database. Because there is no need to physically transfer physical securities, the clearing and settlement process can be very fast.

Straight-Through Processing (STP)

It is normal for settlement schedules to be measured in days (typically three business days, or trade-plus-three, "T+3") since the clearing and settlement process can be highly manual and very complex. It is the exchange's goal to achieve *Straight-Through Processing* (STP), or "T+0," in which settlement can be accomplished on the same day as the trade (or on the next day at worst). Achieving this goal requires that the entire clearing and settlement process be automated and be highly accurate and reliable.

The clearinghouse used by the exchange is able to meet this criterion. All securities are maintained electronically so that trades can be matched and assigned to buyers and sellers with no manual intervention. A trade is rejected for manual reconciliation only if there is a mismatch. For instance, Trader A says that he sold 100 shares of IBM to Trader B, but Trader B says that he bought 200 shares of IBM from Trader A.

A key challenge that the exchange faced in attempting to achieve T+0 was that the entry of trade information into the clearinghouse's system was an error-prone, manual data-input process that had to be done by the trader. The result, a high rate of trade-reporting errors, led to many trade rejects requiring manual reconciliation as well as to multiple STP settlement violations, in which trades could not be settled on the same or next day.

In order to correct this problem, the exchange and the clearinghouse cooperated to facilitate the electronic entry of trades from the exchange's trading system into the clearinghouse system. Ever since this solution was implemented, the exchange has offered STP to its traders.

The Original System

The original method for trade entry, reconciliation, and clearing is shown in Figure 1. The exchange's trading system is built upon a pair of HP NonStop server systems configured as an active/standby pair for disaster recovery. The trading system's databases use HP NonStop Enscribe and HP NonStop SQL/MP. The primary system is a four-CPU NS2100 HP NonStop server, and the standby system is a two-CPU NS2100 HP NonStop server.

The exchange's trading application is active in the primary node of its trading system, and the primary node's databases are replicated in real time to the standby node's databases in an active/passive, disaster-recovery architecture. The standby node is thus ready to take over the trading function should the primary node fail. To do so, it must bring up the trading application, mount the trading databases, and switch over the traders to the new primary system.

Traders (the broker/dealers) enter buy/sell orders via their trading terminals. The orders are stored in the exchange's Order Database as they await execution. As the exchange's trading application executes orders, the trade details are stored in the exchange's Trade Database. The trade details are returned to the originating traders as a trade confirmation.

The clearinghouse maintains the security depository, which is the record of ownership for all securities managed by the clearinghouse. At the end of the day, trading activity is reconciled by the clearing application, and the security depository is updated to reflect the new security ownerships.

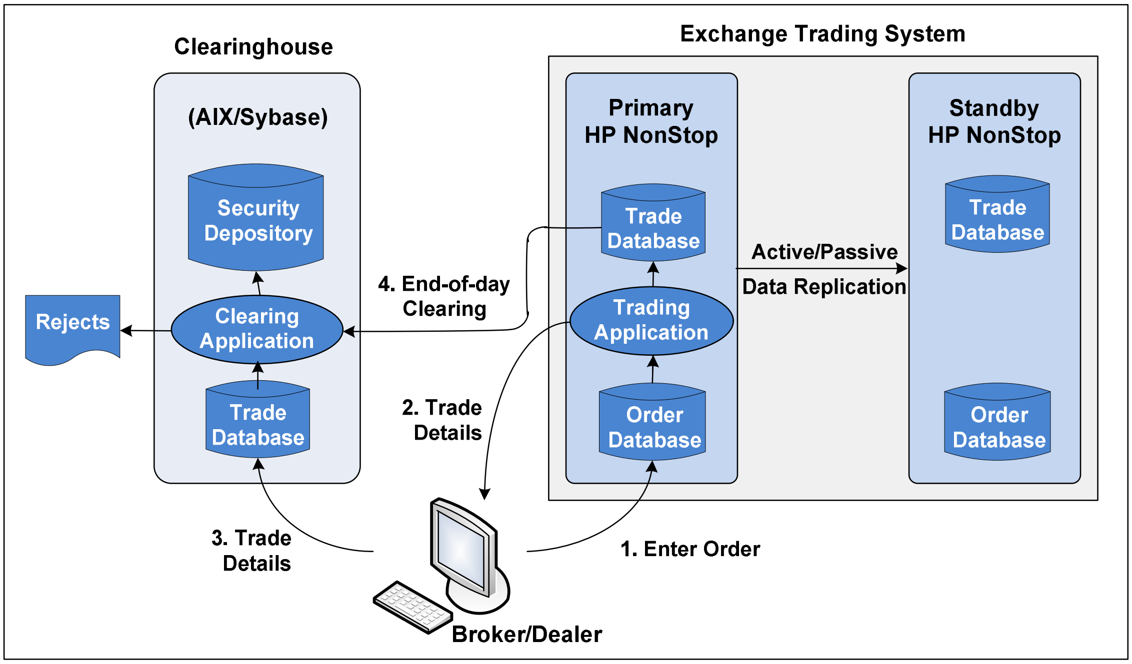

Figure 1 – The Original Trading and Clearing System

The procedure used to trade and clear with this system was originally implemented as follows:

- A broker/dealer entered an order at his trading terminal and submitted it to the exchange (1).
- All pending orders were stored in the exchange's Order Database.
- When the exchange's trading application executed an order, the result of the trade (successful or not, the price, the broker/dealer on the other side of the trade, etc.) was stored in the exchange's Trade Database.
- The trade details were returned to the broker/dealer and displayed on his trading terminal (2).
- The broker/dealer then had to manually enter the trade results into a Trade Database resident in the clearinghouse's system (3).
- At the end of the day, the exchange's trading system transferred its record of the day's trading activity to the clearinghouse's clearing application (4).

Unfortunately, step (3) proved to be time-consuming and highly subject to data-entry errors. Typically, a trader was very busy intensively managing multiple trades, and his primary focus was not on this manual intrusion into his primary trading activity.

At the end of the day, the clearing application attempted to match each trade that it received from the exchange to the trades accumulated during the day from the broker/dealers. Inevitably, there was a high error rate, as many trades did not match due to the data-entry errors. These trades had to be rejected and then manually reconciled. The time that it took to perform these manual trade submissions and eventual reconciliations for mismatched trades added overhead and extra cost and prevented the exchange from meeting many of its STP commitments.

Enhancing Clearing with Data Replication

To correct this problem so that STP could be achieved, the exchange and the clearinghouse cooperated to eliminate the manual re-entry of trade results by the broker/dealers, thereby allowing electronic entry of trades from the exchange's systems directly into the clearinghouse system. The modifications to the original system to achieve this enhancement are reflected in Figure 2.

The first step was to add the clearinghouse's Validation Database to the exchange's trade-entry verification sequence. This database contains all of the consistent and correct values for trade parameters, including security CUSIP numbers, symbols, and valid broker IDs and names. Updates to the database are entered by the clearinghouse, and the Validation Database is replicated to the exchange's trading system in real time in order to keep the two synchronized. The exchange's trading system uses the Validation Database to ensure that orders entered by the broker/dealers are accurately formatted and that they contain standardized parameters. Thus, all orders and their subsequent trades are guaranteed to be consistent and acceptable to the system applications.

One issue that needed to be resolved was that the clearinghouse's Validation Database was a Sybase database stored on an IBM AIX system, whereas the exchange's copy of that database was an HP NonStop SQL/MP database stored on an HP NonStop server. Replication between the two therefore required a data replication engine that supported heterogeneous replication. The exchange chose the Shadowbase replication engine from Gravic, Inc. (www.gravic.com/shadowbase), which supports a wide variety of source and target systems and databases.

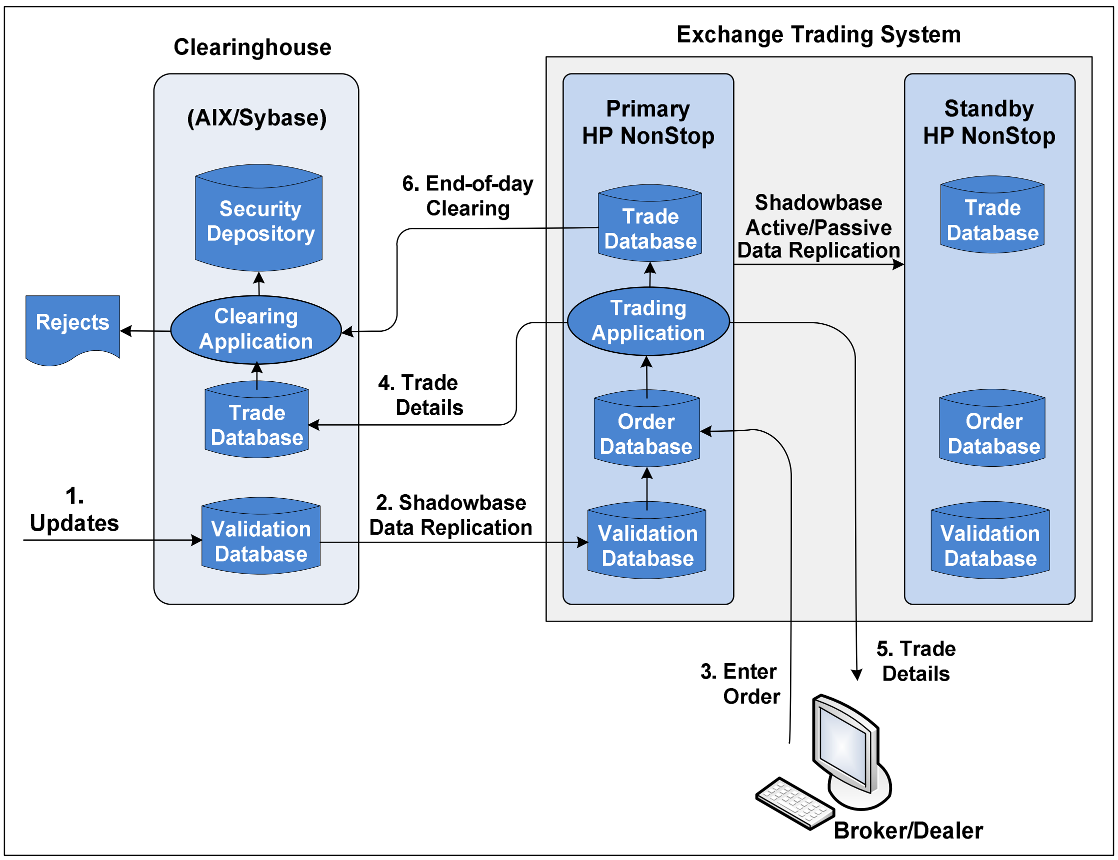

Figure 2 – Enhanced Trade Reporting System

With reference to Figure 2, the trading procedure is now as follows:

- Updates to the clearinghouse's Sybase Validation Database are entered by the clearinghouse (1).
- These updates are replicated in real-time to the exchange's HP NonStop SQL/MP Validation Database (2).
- A broker/dealer submits an order to the exchange's trading system (3).
- The order parameters are verified against the exchange's Validation Database.

- If there is a parameter error, the order is rejected.
- If the order is formatted properly, it is placed into the exchange's Order Database.
- The exchange executes orders in the Order Database as soon as it is able, based on the order parameters.
- On execution, the trade details are sent by the exchange to the clearinghouse, which stores the trade details in its Trade Database (4). (This step replaces the earlier manual re-entry of trade data by the broker/dealers.)
- The results of the trade (the executed order) are returned to the broker/dealer (5).
- At the end of the trading day, the exchange sends its entire trading history for the day to the clearinghouse (6).
- The clearinghouse compares these trades to the trades already stored in its Trade Database that it received during the day from the exchange.
- If there is a mismatch, the trade is rejected for manual resolution.
- For matching trades, the trades are cleared. The security depository is updated to reflect the new ownerships.

Now that manual trade reentries are no longer required of the broker/dealers, the number of trades rejected for manual resolution is minimal, and the exchange is able to meet its STP obligations.

Stock Exchange's Business-Continuity Architecture Imposes Asymmetric Failover Processing

Note that the exchange uses a four-CPU primary system and a two-CPU standby system for the exchange applications. This mode of operation lessens the cost of software licensing and hardware. However, it can lead to load shedding (or reduced response times due to less processing power) in the event of a failover to the standby system. The exchange identified various non-critical applications (e.g., certain reporting subsystems) that will be taken offline in the event of a failover to reduce the overall system load.

The exchange was using another replication product to provide data replication between the HP NonStop servers in its active/passive disaster recovery pair. However, the exchange realized that it did not make sense to have multiple replication products in production. It therefore moved to Shadowbase data replication for its active/passive HP NonStop server replication. In future project phases, the exchange is also considering moving to a higher level of business-continuity solution such as a fully active/active architecture – a configuration supported by the Shadowbase data replication engine.

Exchange Moves to Data Replication for Data Warehouse Feeds

In addition to its trading system, the exchange provides a series of query nodes that act as data warehouses for trading history, as shown in Figure 3. These data warehouses are useful to broker/dealers who want to follow a security over a period of time to help them make purchase and sales decisions. The data warehouses use MySQL databases running on Linux servers.

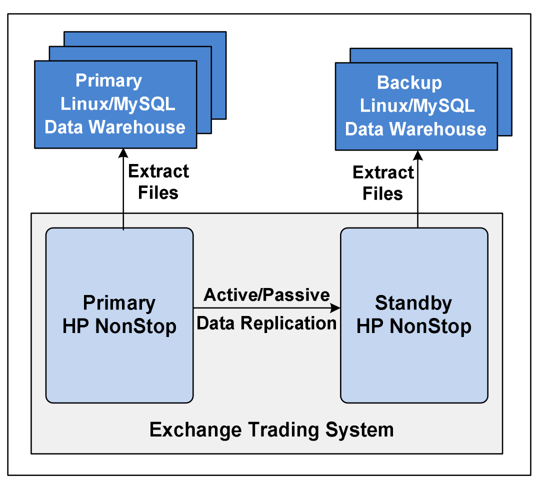

Figure 3 – The Exchange's Original Data Warehouses

At the current time, the exchange has deployed ten data warehouse systems. Seven are collocated with and are connected to the primary HP NonStop server, and three are collocated with and are connected to the standby HP NonStop server.

The previous approach used mini-batches of trading activity to update the data warehouse systems. These were created periodically and shipped as flat files from the exchange servers to the data warehouses. These files were then periodically loaded into the target database. This approach meant that the data warehouses were often lacking the latest data to support broker queries.

The exchange moved to Shadowbase technology to provide heterogeneous, real-time transactional replication of updates from the source HP NonStop databases into the target Linux/MySQL databases (the data warehouses). The warehouse data is no longer stale. Trading activity is now replicated to the data warehouses in real time to keep the information in the data warehouses current with the exchange's trading database.

The Exchange's Many Uses of Shadowbase Data Replication

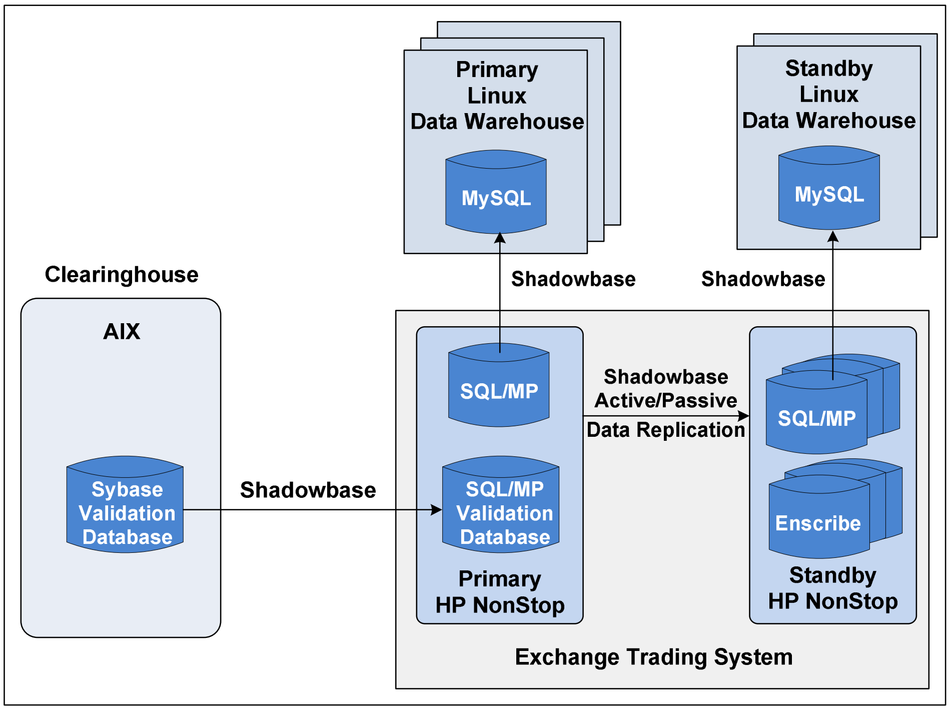

Figure 4 – The Exchange's Many Uses of Shadowbase Data Replication

Once the exchange chose Shadowbase software to meet its original heterogeneous replication requirements, it subsequently discovered other key applications for the product, as discussed below. All of these uses are summarized in Figure 4.

Shadowbase functionality:

- provides active/passive replication between the exchange trading system's HP NonStop servers for the purpose of disaster recovery. Both HP NonStop SQL/MP and Enscribe databases are replicated.
- replicates the validation data from the clearinghouse's Sybase database running on an IBM AIX to the HP NonStop SQL/MP Validation Database running on HP NonStop servers (the original requirement to solve the exchange's manual data-entry problem).
- replicates real-time trading data from the exchange's HP NonStop SQL/MP databases to the data warehouses' MySQL databases running on Linux systems, thus keeping the data warehouse information synchronized with the current state of the exchange's primary trading database.

The exchange's use of the Shadowbase replication engine demonstrates its suitability for both business continuity and data and application integration requirements.

Moving to Continuous Availability

With the current active/passive architecture used by the stock exchange for its trading system, a failure of the primary system could take the exchange down for a significant period of time – several minutes to perhaps several hours. The downtime occurs because at the point of failure, management must first be apprised of the situation and must approve the failover (often leading to extended decision time). The failover must then be initiated (network switching, interface failovers, etc.), and the trading applications must be brought up on the standby system with the databases opened for read/write access. The system must then undergo testing before it is put into service. All of these steps take time, and there are many subtle problems that could lead to *failover faults* that prevent the standby system from being brought into service, resulting in extended downtime.

The stock exchange is considering a Shadowbase *Sizzling-Hot-Takeover* (SZT) configuration to improve availability. In an SZT configuration, the backup system has all applications loaded with all databases mounted for read/write access. Each update made to the primary database is replicated across the application network, thereby keeping all databases synchronized. The backup system is an active system, except that it is not processing any update transactions. If the primary system fails, traffic is simply rerouted to the standby node, and failover is accomplished in just a few seconds. In effect, an SZT system is an active/active system in which all transactions are routed to just one node.

With this approach, the standby node is continuously tested by periodically sending it test transactions to ensure that it is end-to-end operational. This eliminates the need to take down the primary node for testing, as is typically the case for an active/passive system. Therefore, there will be no failover faults, and recovery will be fast and reliable. An SZT architecture also avoids data collisions which can otherwise occur with a fully active/active configuration.

Summary

A stock exchange using a trading system built with HP NonStop servers faced a serious problem in that high data-entry error rates and the resulting manual reconciliation process required to correct these errors prevented the exchange from complying with its same-day clearing goals.

Working with its clearinghouse, the exchange re-architected its system to eliminate manual data re-entry. Orders were pre-validated to ensure that their format was consistent and correct, and the resulting trades were sent directly and electronically from the exchange to the clearinghouse. This eliminated the necessity to manually reenter the information. A necessary part of the new architecture was the requirement to replicate a Sybase Validation Database from the clearinghouse's AIX system to a NonStop SQL/MP database located on the exchange's HP NonStop server trading system. The exchange chose the Shadowbase replication engine, which provides data replication between heterogeneous systems and databases, for this task.

The exchange also manages several Linux/MySQL data warehouses that provide trading history to traders. The data warehouses were originally updated periodically using a batch-updating process. Again making use of Shadowbase heterogeneous replication capabilities, the batch updates were replaced with real-time transactional data replication from the HP NonStop SQL/MP database to the MySQL databases. The result is information in the data warehouses that is now current with the primary trading system.

The exchange also deployed Shadowbase replication for disaster recovery of its HP NonStop-based trading system (providing active/passive data replication between the primary and standby HP NonStop servers). Because Shadowbase replication supports bi-directional replication, the exchange now has an opportunity in the future to move to a continuously available SZT or to a fully active/active architecture, thereby improving its overall application availability.

Paul J. Holenstein is Executive Vice President of Gravic, Inc. He is responsible for the Shadowbase suite of products. The Shadowbase replication engine is a high-speed, unidirectional and bidirectional, homogeneous and heterogeneous data replication engine that moves data updates between enterprise systems in fractions of a second. It also provides capabilities to integrate disparate operational application information into real-time business intelligence systems. Shadowbase Total Replication Solutions© provides products to leverage this technology with proven implementations. For further information regarding Shadowbase data integration and application integration capabilities that can assist in solving big data integration problems, please refer to the companion documents Shadowbase Streams for Data Integration and Shadowbase Streams for Application Integration, or visit www.Gravic.com/Shadowbase for more information. To contact the author, please email: SBProductManagement@gravic.com.