

Payment Authorization – A Journey from DR to Active/Active

December, 2007

A major provider of merchant services to over four million small to medium-sized merchants throughout the world provides, among other services, payment authorization for Visa, MasterCard, American Express, Diner's Club, Discover, and other credit and debit cards. Card transactions made at merchant point-of-sale (POS) devices and ATMs are verified to ensure that they are proper.

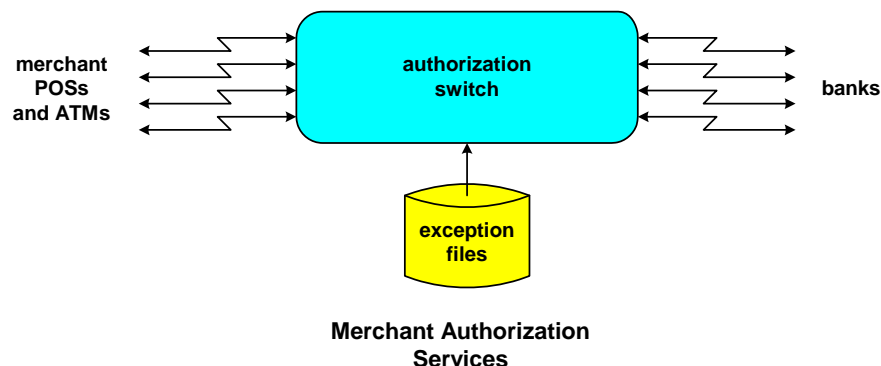
The company's payment authorization services are extremely critical. Should these services become unavailable, shoppers all over the world will not be able to use their credit or debit cards processed by the company. Therefore, the company has worked diligently to guarantee that its authorization services will always be available. It turned to data replication technology to satisfy this need.

Over a period of several years, the company has expanded its use of data replication technology from disaster recovery to active/active systems to application integration. This has been a slow and careful process as the company learned the benefits of data replication. Today, this effort has resulted in a system that indeed is continuously available.

Payment Authorization

Credit and debit cards are issued by banks. They are used by consumers to purchase items from brick-and-mortar and online stores and to withdraw cash from ATMs (automated teller machines). Each card transaction must be approved by the issuing bank before it can be accepted by a merchant or by an ATM.

The payment authorization process begins when a consumer's credit card is presented for payment. This might occur at a point-of-sale (POS) device, such as a credit-card reader attached to a cash register in a store. It might be initiated when a consumer gives his credit card number and other information to a sales person over the telephone or when this information is entered into a form on a web site. It might begin at an ATM (automated teller machine).



The POS device, the ATM, or the system hosting the sales support for telephone or online ordering is connected by a communication channel to an authorization switch established and managed by an organization providing merchant services. The authorization switch receives the transaction and analyzes it. It determines which bank is servicing the credit or debit card and sends the transaction to that bank. It will check if the card has been lost or stolen and may also check the transaction for suspicious charge activity.

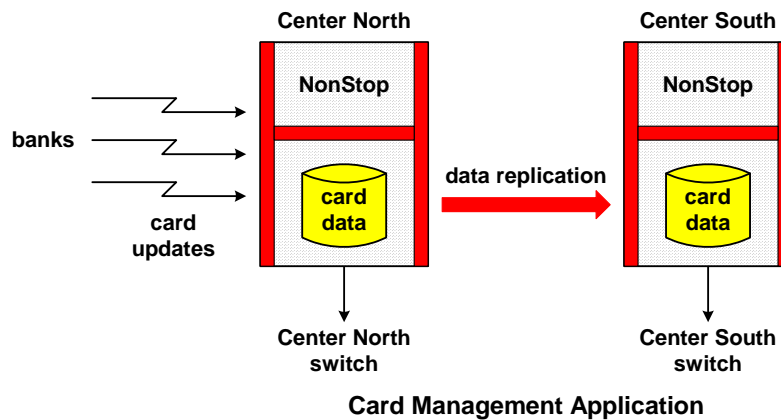
If the bank responds positively, and if the switch detects no other problems, an authorization message is returned to the originating device or system. If the bank rejects the transaction, if the card has been reported as lost, stolen, or frozen, or if the transaction is suspicious (i.e., a potentially fraudulent use of the card), the transaction is rejected.

A Start with Disaster Recovery

The merchant services provider operates a major authorization switch in North America. It has two data centers 1,000 miles apart on the East Coast of the United States. We will call these data centers Center North and Center South. The switch functions are duplicated in each data center to provide recovery from any disaster that might take down one of the provider's data centers. In normal operation, the processing load is distributed between the two sites.

The company's first use of data replication was to keep its Card Management Application (CMA) systems in each data center synchronized with each other. Hosted on NonStop servers, the CMA systems provide a database of all cards issued by participating banks. These cards include Visa, MasterCard, American Express, Diner's Club, Discover, and others.

Banks send batch ftp files to the CMA system for new cards they issue, cards they report as stolen or lost, and other card status changes. These batches are received by the Center North CMA system and stored in its card database. This database is then replicated to the Center South CMA system via the Shadowbase replication engine from Gravic (www.gravic.com). Thus, both CMA systems always have the complete and up-to-date data of all cards that have been issued.



The CMA database specifies which issuing bank services each card, what the card's characteristics are (its card number, its expiration date, its credit limit, its PIN, and so on), and its status (active, lost, stolen, frozen, and so on). Additions and changes to this database are sent by each CMA system to the switching nodes local to it. Therefore, each switching node has direct local access to all card data, a requirement for fast authorization. Fast authorization is needed since a consumer is waiting to complete his transaction. Also, while he is waiting, the POS, ATM, or telephone sales clerk is occupied and cannot service the next customer.

By maintaining a copy of the CMA database at each site, the company ensures that the CMA data is available at all times even in the event of the failure of one of the systems. Should one system fail, all batch updates from the issuing banks will be routed to the other CMA system; and this system will take on the responsibility for keeping all nodes at both sites updated.

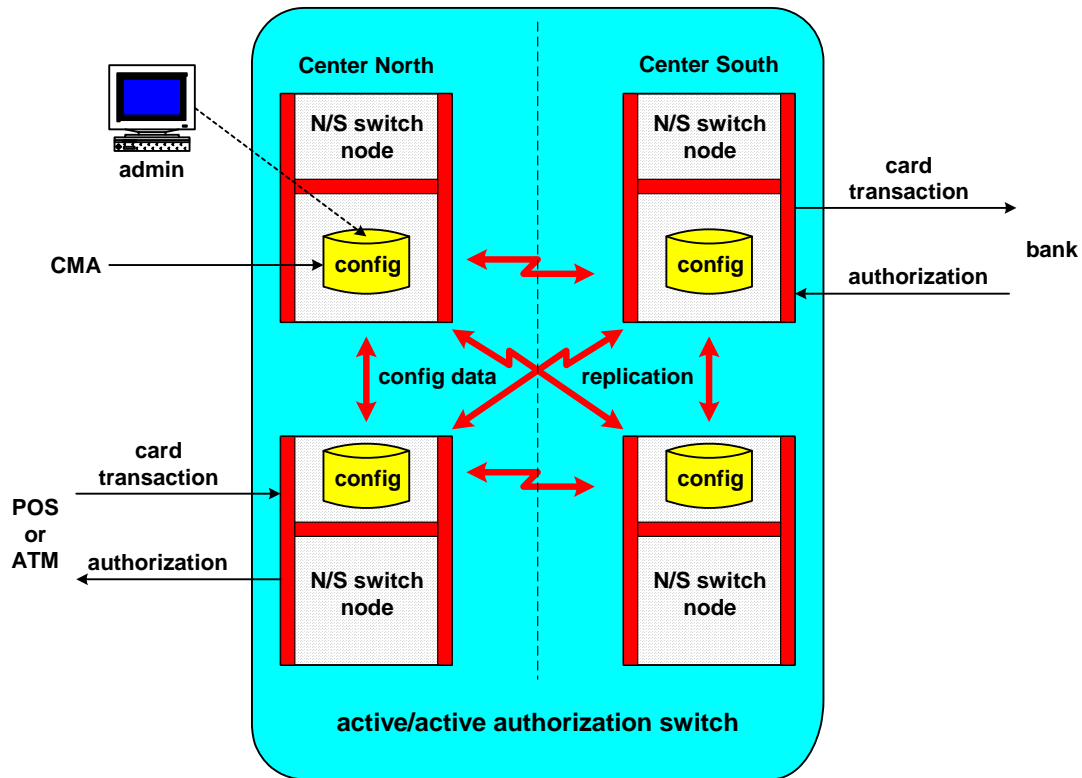
This architecture also allows rolling upgrades to be made to the CMA systems. One system can be taken down and its operating system, application, database, or hardware upgraded while the other system carries the CMA load. The first system can then be returned to service and the second system taken down for upgrade.

The Move to Active/Active

After several years of successful operation with the disaster recovery (DR) configuration for the CMA systems, the service provider moved to an active/active configuration for its authorization switch. The switch contains four nodes – two in Center North and two in Center South. Each node is connected to thousands of communication lines that connect to the company’s massive IP network. When a transaction is received from a merchant, the job of the switch is to validate the card/transaction, route it to the issuing bank, and then to return the response to the merchant.

Switch Configuration

Though all switching data is kept in memory, the switch configurations are kept on disk in a configuration database. Configuration data contains information about each card and each issuing bank as well as state data for each ongoing transaction. The configuration database also contains failover configurations so that surviving nodes can take over the functions of a failed node. Consequently, the configuration database is fundamental to proper switch operation. A copy of the configuration database is maintained on each node for local access. These copies are kept synchronized by the Shadowbase bidirectional data replication engine.



The credit and debit cards are partitioned across the nodes by card number. As a consequence, each card is “owned” by one of the nodes. This prevents data collisions on card updates due to transaction activity from occurring. All transactions for a card are routed to the node owning that card. That node makes transactional updates to its database, and its changes are replicated to the other database copies in the network by Shadowbase. Thus, no two nodes can be making simultaneous changes to a card due to transaction activity; and data collisions are avoided.

Transaction Processing

When a transaction for a card is initiated, the authorization switch’s IP network routes the transaction to the node owning that card based on the card number range reflected in the transaction. When a card transaction request is received by its owning node, the configuration database is accessed to determine which bank issued that card and therefore to which bank the transaction should be sent. The configuration database specifies the IP address of the issuing bank, and the owning node forwards the transaction request to that bank.

When the issuing bank returns a response to a transaction request, the company’s IP network returns that response to the node owning the card. An authorization or rejection message is sent by the owning node to the initiating device to complete the transaction. The transaction is authorized if the issuing bank authorizes it and if the authorization switch has detected no other problems, as described later (lost or stolen card, frozen card, suspicious activity, and so on).

Node Failover

Equally important is the failover configuration data maintained in the database. Should a node fail, the configuration database tells the surviving nodes for which cards they should take responsibility. The surviving nodes then assume ownership of their newly assigned cards so that switching services can continue. The configuration failover database also tells each node which applications it should take over to continue operations.

Thus, should a node fail, all cards that are owned by the failed node are switched to surviving nodes as are the failed applications. Authorization services continue without interruption. In fact, this procedure is repeated for multiple node failures so that operations can continue even if more than one node fails.

Replicating Changes

The redistribution of cards, card transactions, and applications to surviving nodes is one of the requirements for an active/active system. Equally important is that each node must have a current copy of the application database. In the case of the authorization switch, there is no real-time data to be replicated since it is all kept in memory. If a node fails during the processing of a transaction, the originating system will time out and will resubmit the transaction. The transaction will then be processed by the new node configuration.

However, there must be a current copy of the configuration database resident at each node. The configuration database is accessed and may be updated during each transaction. In addition, card changes are received from the CMA system. The configuration database may also be modified by administrative operators. All of these changes are replicated to each of the other nodes by the Shadowbase replication engine so that each node has a current copy of the configuration database.

Though transactional changes to a card are made only by its owning node and are replicated to the other nodes from that node, it is possible that administrative changes to a card may be made at other nodes. This, of course, opens up the possibility of data collisions should a change to the same credit or debit card be made at two different nodes almost simultaneously (within the replication latency interval). Though this is unlikely due to the nature of the card partitioning, if

there is a collision, the data collision resolution algorithm selected by this customer is that the later change is accepted. The other change is rejected and is logged for subsequent manual review.

The Shadowbase replication engine performs another important function. As the failover configuration is modified (which it must be for every new card), it is entered with respect to the node receiving the configuration updates. However, this configuration data is different for each node since each node will take a different action on failover. For instance, node 1 might handle card A. Should node 1 fail, node 2 might handle card A. Should node 2 then fail, node 3 might handle card A. Via built-in business rules, the Shadowbase replication engine automatically adjusts the configuration data for each node as it replicates the data.

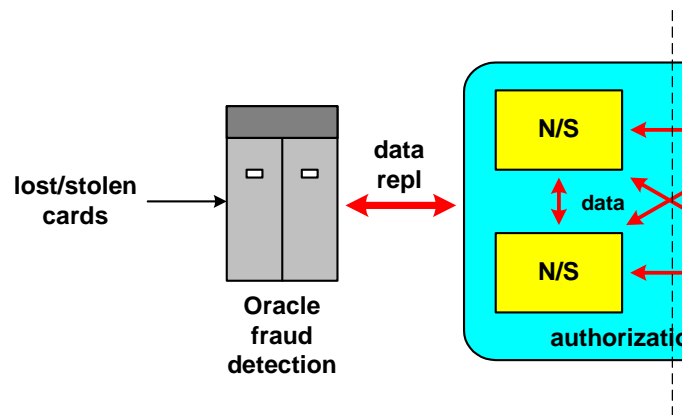
Thus, the company has converted its four-node switch to an active/active configuration that can recover almost instantly from any fault in the system, including a total node (or multinode) failure. Recovery is transparent to the users.

Heterogeneous Application Integration

A major function of authorization is to detect the potential fraudulent use of cards. If a transaction looks suspicious – for instance, it is for a purchase in a New York store whereas the last transaction one hour ago was for a purchase in a Los Angeles store - it may be rejected.

The company implemented a fraud detection application using an Oracle system. Each transaction received by the authorization switch is sent to the Oracle fraud detection system. This system maintains a log of recent transactions and checks each transaction against other recent transactions for the same card. If a transaction appears to be suspicious, the result is passed to the authorization switch, which may then reject the card's transaction. The switch may also notify the issuing bank of its action; and the bank, at its discretion, may freeze the card. The card holder is notified and must call in to verify his credentials and recent transactions.

The fraud detection system is also periodically sent a list of cards that have been reported lost or stolen or that have been frozen by the issuing bank. If a transaction is made against such a card, this situation is reported back to the switch, which will reject the transaction.



Fraud Detection

At each data center there is one fraud detection system serving the nodes at that site. The company implemented the communication between the fraud detection system and the authorization switch nodes again via the Shadowbase replication engine. When a transaction is received, a notation is made in the card's database on the authorizing switch. This notification is replicated to the fraud detection system, where Shadowbase detects the request as it is updating

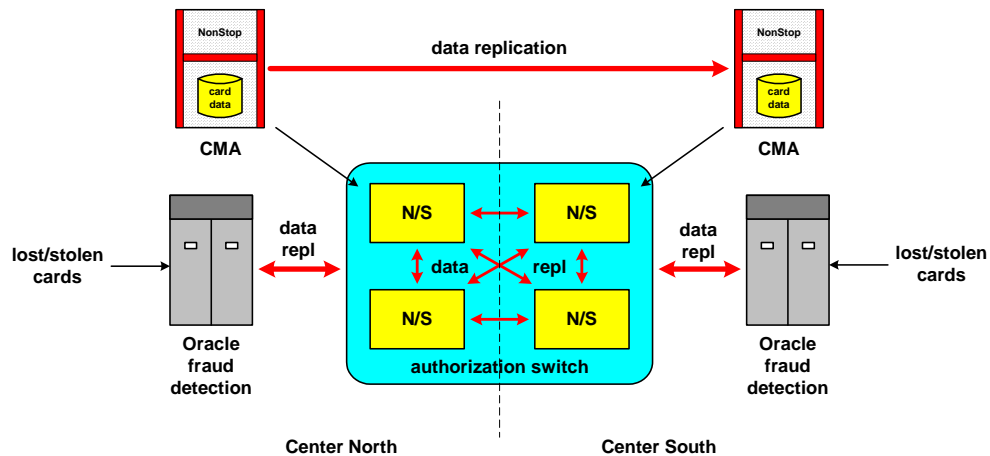
the Oracle database. The request is passed to the fraud detection application, which logs a response by updating its database. When Shadowbase replicates this change back to the authorization switch, the Shadowbase software on the receiving end detects it and passes the response to the authorization switch's switching logic. This is a case of heterogeneous application integration since the authorization switch nodes are NonStop systems and the fraud detection systems are Unix/Oracle-based.

The authorization switch maintains its own database – an exception file – of all cards against which transactions have been initiated and that have been reported as lost, stolen, or frozen. Included in this exception file is also a list of cards that have been rejected by the switch due to suspicious activity. A card is cleared from this list when the issuing bank so notifies the switch during one of its batch updates of card changes.

The authorization switch is heavily involved in the authorization decision. When it receives a transaction, it first checks its exception file. If the card is not listed in the exception file, the switch sends the transaction to the fraud detection system and to the issuing bank. Only if all tests are positive – it is not in the exception file, it is not suspicious, and the issuing bank authorizes it – is the transaction accepted. The originating device is then notified to accept the transaction. If any of these tests fail, the originating device is instructed to reject the transaction.

The Integrated Authorization Switch

This merchant services company has carefully expanded its use of Shadowbase's data replication over the years. The company initially used data replication to provide a uni-directional active/passive data-recovery capability for its Card Management Application. After feeling comfortable with this use of data replication, the company expanded its use of data replication technology to provide continuous availability for its multinode switch. Finally, it integrated heterogeneous applications (the fraud detection system) via data replication.



This case study is important in that it demonstrates the many uses of data replication engines in mission-critical applications.