the Availability Digest

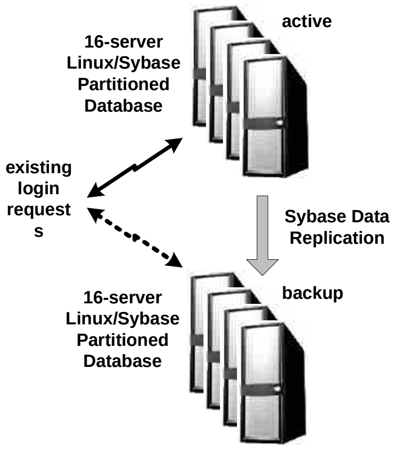
Major ISP Migrates from Sybase to NonStop with No Downtime

November 2008

Overview

A major international Internet Service Provider (ISP) offers email services, Internet access, and other functions to millions of global customers. Several million of these customers may be logged in at any one time. Maintaining continuity of service to its customers is a mandatory requirement for the ISP.

Its rapid growth has led to capacity strains in the ISP's IT infrastructure. One such issue recently occurred in its login subsystem. When a customer logged in, his login request was sent to the ISP's Login Request Complex. There his user profile was accessed and validated and his session established. Though comprising a large farm of redundant servers running many instances of Sybase on Linux, this login subsystem had reached the limits of its capacity. Further additions to capacity were going to be very expensive.


active
16-server Linux/Sybase Partitioned Database
existing login request s
Sybase Data Replication
backup
16-server Linux/Sybase Partitioned Database

The Login Request Complex

The ISP therefore decided to architect and build an entirely new login subsystem. It performed a series of functionality, performance, and load volume torture tests using many of the major commercially available database engines, including Sybase, Oracle, and NonStop's SQL/MP. The tests were run under the configuration and tuning guidance of each vendor. As the load was scaled up to the ISP's load levels, one of the competitors could not complete the tests. Another limped along under full load, not able to complete queries within the desired SLA. The clear winner was NonStop's SQL/MP, running on the HP Integrity NonStop hardware in an active/active configuration - it completed all tests satisfactorily.

Not only would the single-system image presented by the active/active system make the Login Request Complex significantly more manageable, but the rapid failover time offered by the active/active system (seconds) would also ensure that the loss of a node would not be noticed by the users that it was servicing at the time of failure. Furthermore, if capacity were added in the future, only the capacity needed would be purchased, rather than twice the capacity as required by the redundant Linux/Sybase arrays.

The problem then became how to migrate from the old Login Request Complex to the new NonStop system without impacting the ISP's customers. The goal was to perform an online migration of the application with zero (or minimal) application downtime, often referred to as Zero Downtime Migration, or ZDM. By the judicious use of the Shadowbase data-replication engine from Gravic, Inc. (www.gravic.com), the ISP was able to gracefully migrate all of its customers over a period of time to the new NonStop system with no interruption in service.

The ISP Login Request Complex

When a user wants to log in to the ISP, his login request is sent to the ISP's Login Request Complex. Until recently, this complex comprised a primary farm and a standby farm, each with sixteen login servers. The farms were geographically separated for the purpose of disaster tolerance and were interconnected by a WAN. The servers were large Linux systems running Sybase database managers. Each server was very powerful, comprising multiple processors. The login servers stored the profiles for all of the ISP's users. As a user login request was received by these servers, a user session was set up with his or her profile.

For each server in the primary farm, there was a similarly configured backup server in the standby farm. The databases of the standby servers were kept synchronized with the primary databases via Sybase data replication.

The users were partitioned across the Login Request Complex according to their user names (screen names). Intelligent routers directed a user login request to the appropriate server in the Login Request Complex. During normal operation, login requests were routed to the primary farm. Should a server in that farm fail, its login requests were then passed to its companion in the standby farm by the intelligent routing infrastructure.

The Challenge of Migration

The ISP's customer base was rapidly growing. To accommodate this growth, the Login Request Complex had to be expanded. This could have been done by adding more servers to the Login Request Complex and by repartitioning the users across the increased number of servers.

However, these servers were very large and expensive. Significant licensing expenses added to the cost. This already high expenditure was further compounded by the costs of managing such a large complex of servers.

Additionally, each of the databases on the individual servers were standalone and not integrated; and operational transactions had to be individually generated and assigned to the appropriate database instance. For example, to generate a query that aggregated the information across all of the databases, one query per database had to be generated and individually run. Then the results had to be manually aggregated. Worse, the amount of data per database instance had grown so large that queries that scanned the largest tables often timed out before completing.

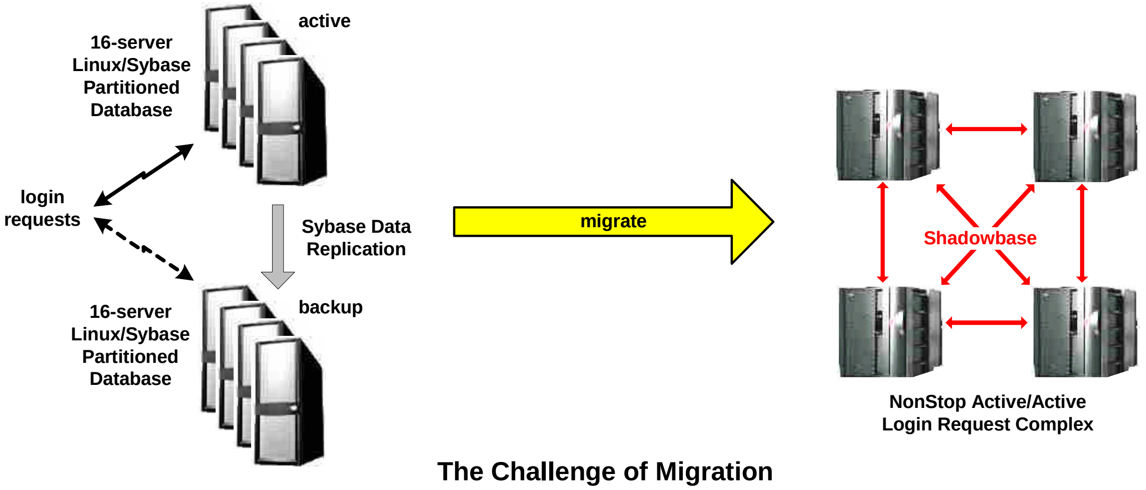
More problematic was that of transaction updates. If a transaction updated data across the databases (for example, an insert on one and a delete on another), the application had to programmatically submit two transactions, one per database instance. If one of the transactions failed, the application had to programmatically note this and "undo" the one that was successful. This led to complex application transactions and backout logic when updating multiple instances, and the databases often became corrupt when the application itself failed or a system fault occurred.

Therefore, the ISP decided to move its login processing to a new Login Request Complex organized as a four-node, geographically-distributed active/active system.[1] HP NS16200 NonStop servers were to be used as the nodes in the active/active network. Shadowbase was chosen to keep the nodes in the active/active system synchronized.

[1] What is Active/Active?, *Availability Digest*; October, 2006.

The query and update problems posed by the multiple, independent Sybase databases described above are solved by the active/active Login Request Complex. The NonStop active/active system presents a single-system view with the database partitioned over many disks. Thus, single queries and transactions that span the entire database can be issued.

The problem facing the ISP was how to migrate from the existing Linux/Sybase servers to the new NonStop active/active system without impacting service to its customers – a Zero Downtime Migration.



The Challenge of Migration

The Migration Solution

Working with the Shadowbase engineers, a Zero Downtime Migration (ZDM) solution was formulated.[2] Briefly, temporary servers would be set up to capture all changes to the Sybase databases in the Login Request Complex. This change capture function would then be activated while the Sybase database contents were loaded into the NonStop databases via an ETL (extract, transform, and load) utility. Once the load was completed, the changes that had accumulated during the load would be sent to the NonStop servers to synchronize them with the Sybase servers. At this time, the users could be switched to the NonStop servers for login service.

To minimize any potential negative impact to the users, this process would be carried out in phases. There would always be a failback path to the Sybase Login Request Complex should a migration step fail. The migration would proceed over a period of several months, after which the Sybase servers would be decommissioned.

Additionally, the customer dictated that no additional replication or application software other than the existing Sybase replication product be installed on the production servers. This meant that the Sybase replication server had to be used to replicate the database information off of the existing servers. Since Sybase does not directly support replication to NonStop targets, Shadowbase had to provide the intermediate "glue" to connect the systems.

Step 1 – Capture Changes

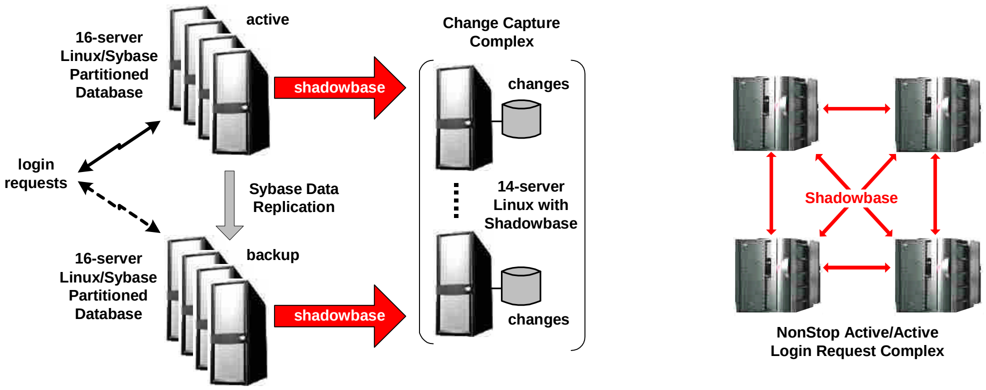
The first step in the implementation of this plan was to provide for the capture of Sybase database changes. This was achieved by installing a server farm of fourteen systems running the Linux operating system, thereby creating the Change Capture Complex. However, the only

[2] B. D. Holenstein, W. H. Highleyman, P. J. Holenstein, Chapter 8, Eliminating Planned Outages with Zero Downtime Migrations, *Breaking the Availability Barrier II: Achieving Century Uptimes with Active/Active Systems*, AuthorHouse; 2007.

application running on these servers was the Shadowbase replication engine. There were no ISP applications, and Sybase was not required.

As a result, the processing load on the servers was not large; and smaller, less expensive servers could be used. The servers comprised a minimal configuration of only two CPUs each.

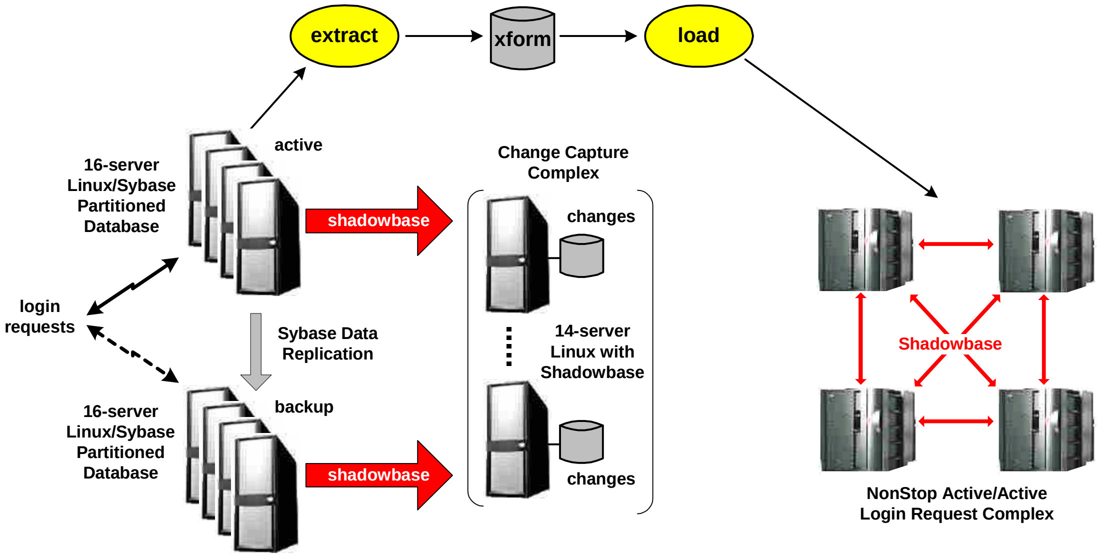
Shadowbase was used to capture changes on the Sybase servers and to replicate them to the Change Capture Complex. There the changes were queued until needed to synchronize the NonStop servers. Replication was provided for both the primary and standby Sybase farms to accommodate times when one or more primary Sybase servers were down. However, all changes were captured by the common Change Capture Complex.



Step 1 - Capture Changes

Step 2 – Load Active/Active System Databases

With the Change Capture Complex installed and tested and the new login applications ready on the NonStop servers, now was the time to begin the actual migration. An Extract, Transform, and Load (ETL) utility was used to copy the data from the actively running Sybase servers to the NonStop servers (Step 2a).

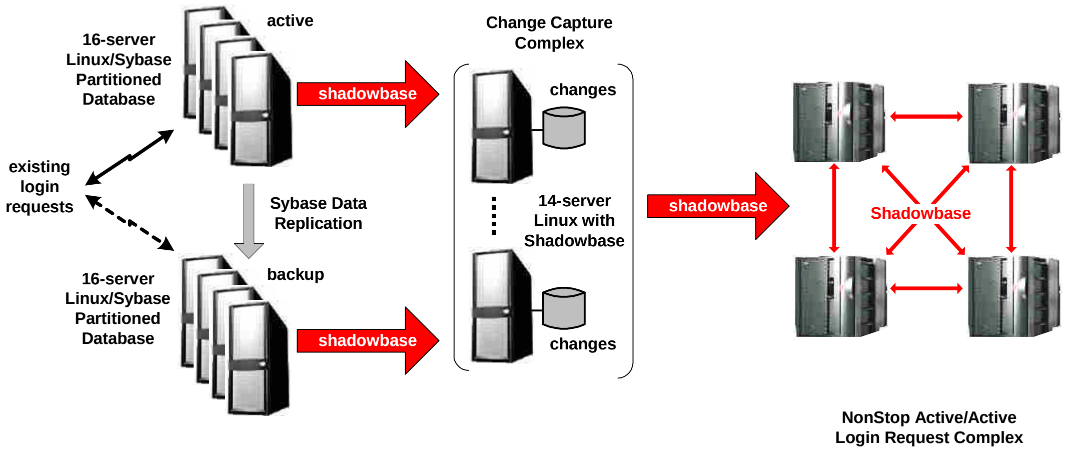


Step 2a - Load Active/Active System Database

At the same time, any changes that were made to the Sybase databases following the initiation of the ETL process were replicated by Shadowbase to the Change Capture Complex, where they were queued for later use.

At the end of the load process, the database for the ISP’s users was now on the NonStop servers. However, the NonStop database was inconsistent and out-of-date since copying was taking place while the source database was active and being changed by the ISP’s applications. Row changes that arrived at the Sybase Login Request Complex after those rows had been loaded were not reflected in the NonStop database. Therefore, the NonStop database still had to be brought into synchronization with the Sybase database.

This was the role of the Change Capture Complex. When the load had been completed, the changes that had been made during the load and that were queued in the Change Capture Complex were now replicated to the NonStop servers via Shadowbase (Step 2b). Of course, during this process, changes were still arriving at the Sybase servers; and these were also queued by the Change Capture Complex and ultimately replicated to the NonStop servers. This continual updating of the NonStop database from the original Login Request Complex continued throughout the migration to ensure that the two databases were properly synchronized.



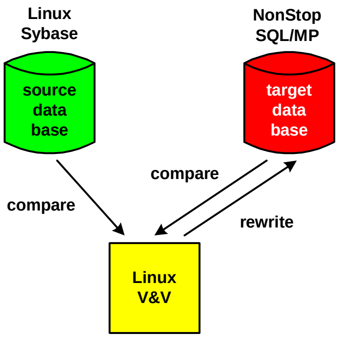
Step 2b - Update Database

Because there may have been cases in which old changes were applied to new data, Shadowbase’s “fuzzy replication” was used. If an update were made to a row that didn’t exist, the update was turned into an insert. If an insert were to be made, but the row already existed, the insert was changed to an update. If a row were to be deleted, but it did not exist, the delete was ignored. Ultimately, however, only the latest changes were reflected in the NonStop database; and the database was properly synchronized with the Sybase databases.

Step 3 – Verification and Validation of the NonStop Database

Once the copy operation had been completed, it was imperative to ensure that the NonStop database was, in fact, a consistent copy of the Sybase databases. The databases had to be compared, and any differences had to be repaired by bringing the NonStop database into conformance with the Sybase databases.

The ISP wrote its own verification and validation (V & V) utility to perform this task (Step 3). The utility ran on a separate Linux system that compared rows within key ranges to obtain higher



**Step 3 -
Verification and Validation**

speed through parallel operation. Comparisons were bidirectional so that extra rows and missing rows on the target could be detected.

The complexity of this step was compounded by the fact that the structure of the Sybase databases and the NonStop database were different. Shadowbase made this conversion automatically as changes were replicated from the Change Capture Complex to the NonStop Login Request Complex according to rules the ISP embedded into Shadowbase.

However, the initial load of the NonStop database was not done with the Shadowbase loaders. Therefore, the ISP had to incorporate the data-conversion rules into the ETL utility that was to do the loading.

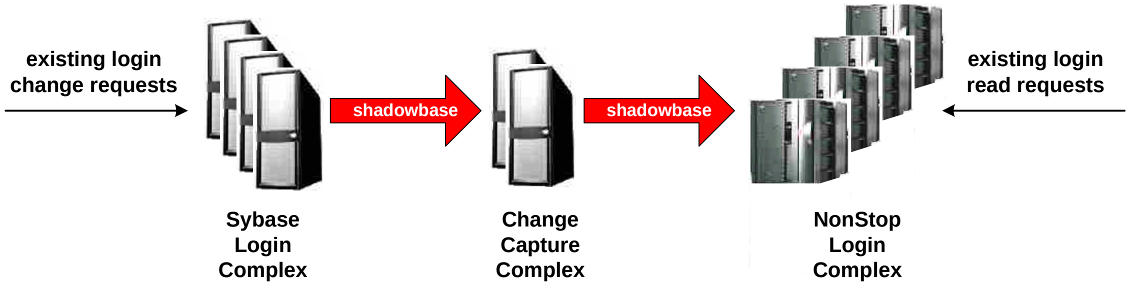
Furthermore, the same rules had to be incorporated into the customer's V & V utility. As a result, many issues in the NonStop database and in the V & V verification logic were found. They included incorrectly mapping fields and improperly converting the Sybase data types into NonStop SQL/MP data types. Consequently, corrections had to be made to the ETL loader and the V & V utility; and the load had to be rerun multiple times.[3] .

However, when all this had been done, the copy accuracy of the process and the data was verified.

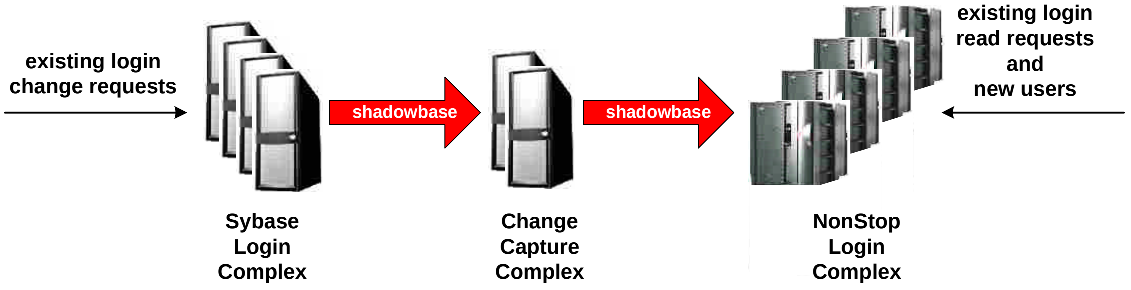
Step 4 – Migrate Users

At the conclusion of the process, all changes had been sent to the NonStop servers. Introducing the new NonStop active/active system into service was done in a controlled fashion (Step 4).

At first, only read requests were routed to the NonStop servers (Step 4a). All update requests were still routed to the original Linux/Sybase Login Request Complex. After experience was gained with this activity, new users were assigned to the NonStop system, which handled both read requests and update requests for the new users (Step 4b).



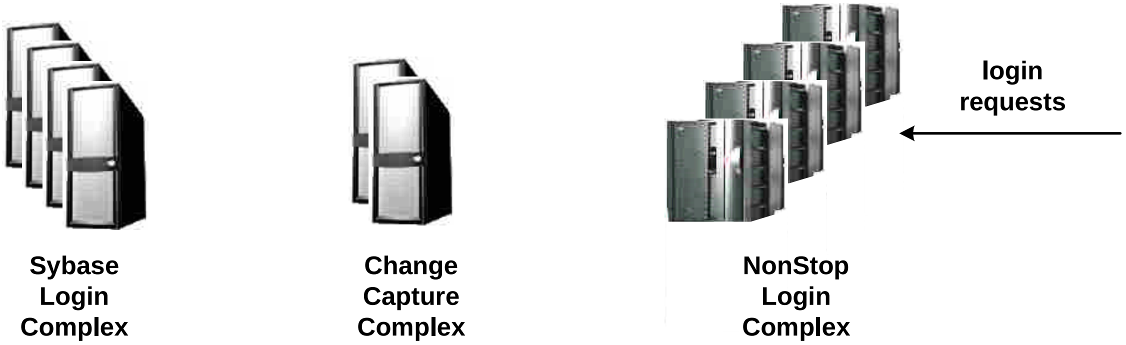
Step 4a - Move Read Requests to NonStop Login Complex



Step 4b - Move New Users to NonStop Login Complex

[3] The customer later acknowledged that it would have been a lot easier and faster to have used a Shadowbase loader to avoid these problems. If a Shadowbase loader had been used instead, the conversion rules would have been consistent.

Finally, the original users were moved to the new complex; and all processing by the Linux/Server farms was discontinued (Step 4c).



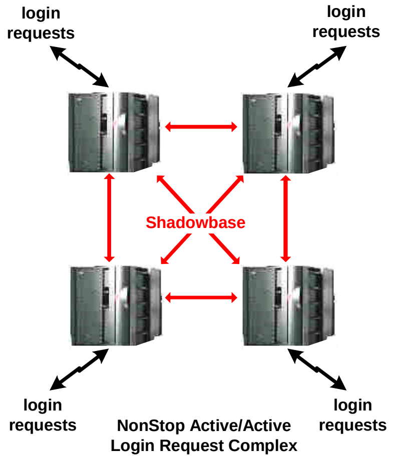
Step 4c - Move All Logon Requests to NonStop Login Complex

The Result

As described above, the migration proceeded cautiously. After several months, all users were successfully migrated; and the Sybase Login Request Complex and the Change Capture Complex were decommissioned.

Each node in the NonStop active/active system acts as a master node for 25% of the ISP's users. When a user request is received, it is routed to its master node by intelligent routers for processing. Since only the master node can update its portion of the database, there is no possibility for data collisions. Should that node fail, the intelligent routers reassign that subset of users to one of the surviving nodes, which becomes the new master for that part of the database.

Should additional capacity be needed, it can be supplied by adding additional nodes to the active/active network. Likewise, load balancing is straightforward. If one node becomes heavily loaded, some of the users who are being serviced by this node can be moved to other nodes.



The Result

Summary

When companies have user communities measured in the tens of millions of active users, any consideration of migrating a core system to another platform can be daunting, especially if it is to be achieved with no application service outage. By using data-replication technology, this ISP was able to successfully migrate several hundred million user accounts to a new NonStop system with no impact on user service. Shadowbase from Gravic provided the necessary services to perform this migration and to implement the new active/active system.