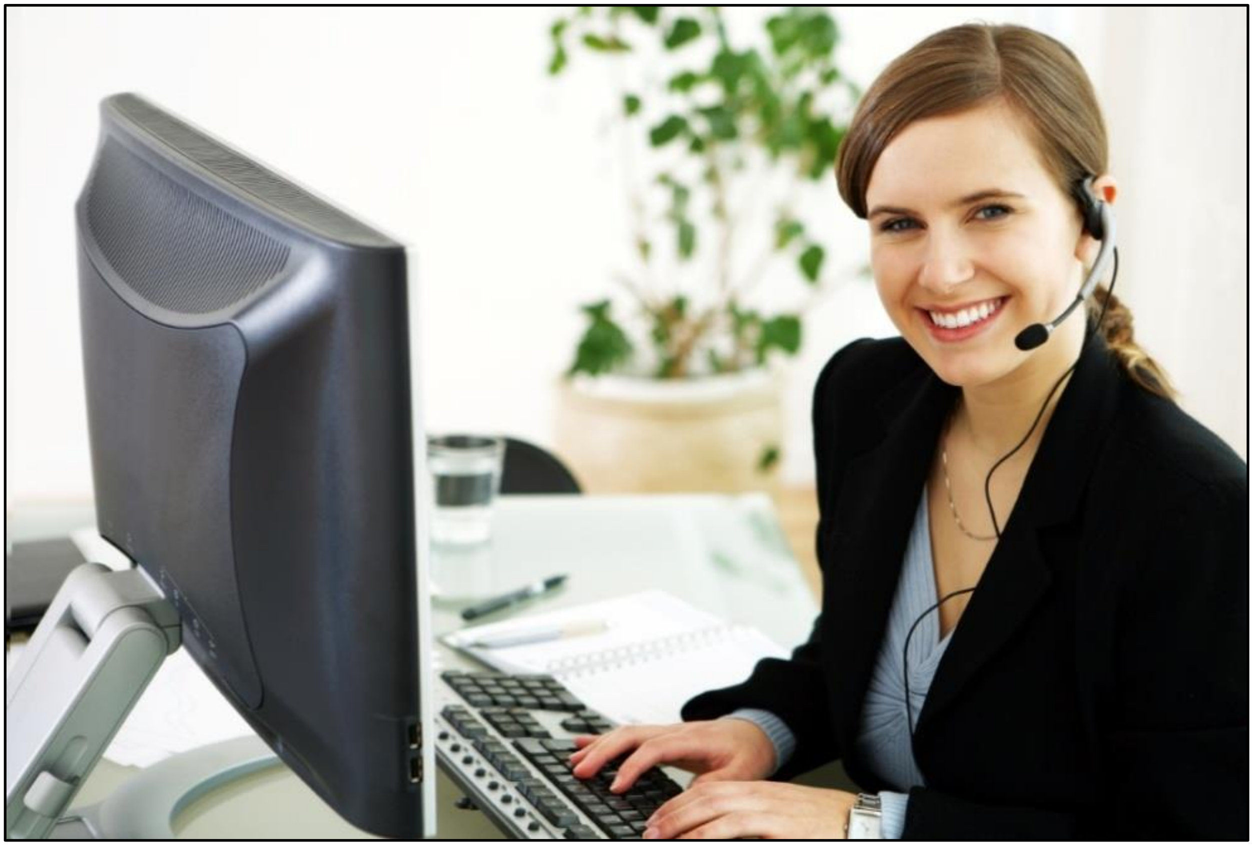
Large International Tour Operator Uses HPE NonStop to Optimize Look-to-Book Processing

A Gravic, Inc. Case Study



Executive Summary

A large, international packaged tour operator depends upon its HPE NonStop systems to provide around-the-clock services to its travel agent customers as well as to groups and individuals who book reservations on its tours.

The Company

The tour operator specializes in organizing tours that are marketed by its customer travel booking agencies. Its tours cover North America, Central America, the Caribbean, Europe, Hawaii, Asia, and many popular ski areas. More than 1,100 destinations worldwide are serviced by its tours. It also caters its tours to individuals and to groups. Founded in the 1970s, the company employs 1,500 people in 15 offices worldwide. It was the first tour operator to make leisure travel available through an airline reservation system. Many of its tours today are planned in conjunction with several major airlines.

It also was the first tour operator to dynamically package, market, and sell vacations via a variety of distribution channels. It currently markets over a dozen-and-a-half tour brands, each specializing in a particular tour venue. In addition to its own tours, the tour operator also provides reservation services for many independent travel booking agencies.

Its customer care and customer quality centers are open twenty-four hours a day throughout the year. The tour operator finds it imperative that its systems are always up. It therefore turned to Tandem Computers (now HPE NonStop) systems to provide round-the-clock availability for booking services for its customers.

The Look-To-Book Problem

A characteristic of travel reservation systems is that a customer makes a lot of queries searching for availability and comparing rates for airline seats, hotels, cars, and other services needed to complete a vacation. Only after a great deal of searching does the customer make a decision and book reservations.

The ratio of queries to the actual booking transaction is known as the *look-to-book* ratio. Before the age of the Internet, when travel agents were doing the booking, the booking activity involved 80% to 90% complex queries and 10% to 20% booking transactions. The Internet changed all that. Now users search many sites to find the best deals before booking. In response to a single user query, travel aggregation sites query many travel sites to find the best deal for the user. The look-to-book ratio has dramatically climbed to perhaps 500 queries or more for every booking.

The company installed its Tandem system to provide these services before the advent of the Internet, and for years the system performed its functions flawlessly. As the tour operator's business grew, so did the load on its booking system. Then came the Internet with its exploding look-to-book ratios. The problem was particularly aggravated around the holidays when query processing caused the query activity to increase hundreds of times the normal query processing load.

This load ultimately grew to the point at which the booking system was in peril of running out of capacity. The tour operator faced the problem of expanding the booking system's capacity beyond the capabilities of a single system.

The Move to Asymmetric Capacity Expansion

The company decided to move to a multinode, distributed system by placing the query processing and the actual booking transaction processing onto different systems. Its plan was to have one master booking system that handled all booking transactions. This system maintained the master booking database, which was to be replicated to multiple read-only query systems that would handle the heavy query load.

Architecture

The tour operator chose HPE NonStop for its master system, as shown in

Figure 1. The booking database is maintained on the master system by HPE's NonStop SQL/MP. A single NonStop system easily handles all of the booking transactions so long as it does not also have to field a massive number of queries.

The master system's booking database is replicated to the query systems via the HPE Shadowbase data replication engine. In this way, the heavy query load is distributed across many systems.

To maximize performance, the key structure of the master database is different from the key structure of the query nodes. The master database uses a key structure optimized for OLTP transaction processing ("skinny" keys). This structure allows it to post booking transactions with a minimum of processing load. The query databases use keys optimized for OLQP query processing ("fat" keys) which allows queries to be executed as quickly as possible.

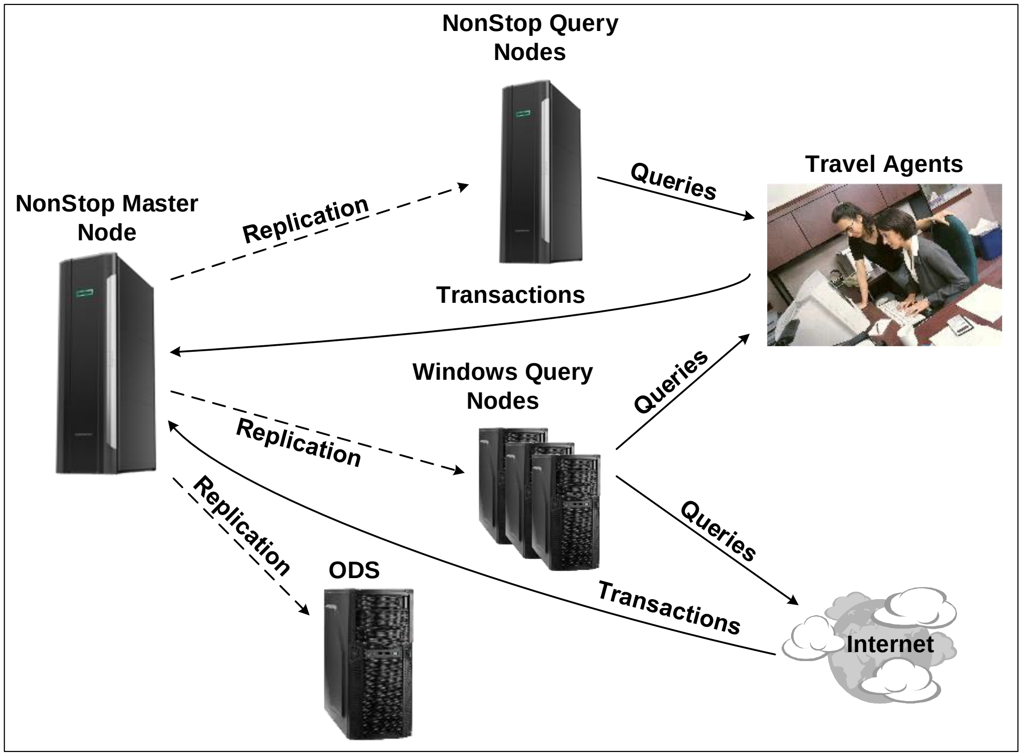


Figure 1 – Tour Operator Distributed NonStop System with Continuous Availability and Asymmetric Capacity Expansion

To provide the reliability that the tour operator needs, some of the query nodes are also NonStop systems. In this way, if the master NonStop system fails, one of the NonStop query nodes could be promoted to be the new master. After all, each query node has a full copy of the database available to it. However, for economy purposes, most of the query systems are Windows systems using Microsoft SQL Server (MS SQL). Shadowbase architecture provides the heterogeneous replication between NonStop SQL/MP and the Windows MS SQL databases.

Capacity Expansion

All that is necessary using this architecture to support asymmetric capacity expansion as the tour operator's business increases is to add Windows or NonStop query systems. No system needs to be replaced with a larger system, although this is always an option. No changes to any applications need to be made. Capacity expansion is simple and economic.

Total Cost of Ownership

The tour operator's new architecture significantly reduces its total cost of ownership (TCO). It now expands to meet increased query demands by simply adding inexpensive commodity Windows servers as additional query nodes.

Availability

The architecture has many advantages from an availability viewpoint. In fact, its availability characteristics are very similar to those of active/active systems it provides, in effect, continuous availability.

First of all, the distributed architecture allows the nodes to be geographically separated for disaster tolerance. If a query system fails, all that is necessary is to reroute the queries normally handled by that query system to other surviving query nodes, whether they be Windows nodes or NonStop server nodes. Thus, recovery time following a query node failure is fast enough that it may not even be noticeable to a travel agent making a query. In the worst case, he or she will simply have to repeat the query. When a failed query node is returned to service, all that needs to be done is to synchronize its database with the master node and then redirect the appropriate queries to it.

A similar procedure is used for a master node failure. The query nodes each contain an up-to-date copy of the master database. Should the NonStop master node fail, a NonStop query node could be rapidly promoted to be the new master node. All that is necessary is to reroute transactions to the new master node and to redirect the replication configuration to point to the new master node.

Since the new master node (a query node in its former life) has the entire booking database, it immediately begins processing booking transactions. True, the new master node is applying transactions using the less efficient “fat” keys; but it is nonetheless processing transactions properly. The longer transaction processing time is a fair tradeoff for continuous availability in the unlikely event of a master system failure. The promoted node continues to handle its original query load as well as the transaction processing load. Alternatively, queries normally directed to it are now directed to other query nodes.

When the original master system is ready to go back into service, its booking database is synchronized with the current master system. The original master system is then returned to service by routing new transactions to it and reconfiguring the replication engines. The query system that stood in for the failed master system returns to query-only service.

Eliminating Planned Downtime

This architecture also is used to eliminate planned downtime. To upgrade a query node, queries directed to it are directed to other query nodes. The query node is then taken down, upgraded, and returned to service. Its queries are now redirected to it. In this way, an update is rolled through all query nodes one node at a time.
The master node is upgraded by first promoting a NonStop query node to be the new master. The original master node is then taken down, upgraded, and returned to service. At this point, all transactions are redirected to it, freeing up the promoted node to return to dedicated query processing.

Data Locality

Another advantage that this distributed architecture provides is that query systems are located in each of the tour operator’s fifteen worldwide offices. Providing this level of data locality significantly decreases the response time of queries and reduces communication costs.

Application Integration

The tour operator took advantage of the new system’s data replication facilities to integrate an Operational Data Store (ODS). Shadowbase software replicates data from the master booking node to Windows servers running MS SQL that support a large ODS application used to improve the quality and appeal of the operator’s tours.

Summary

By expanding its original HPE NonStop system to a multinode, heterogeneous distributed system, the tour operator gained many advantages. The new system provides continuous availability, and in the event of a failure of a node, will have a negligible impact on the operator’s travel agent customers. This availability advantage extends to disaster tolerance since the destruction of any one processing site is immediately recovered by rerouting traffic to another site.

Availability is further improved by the elimination of planned downtime. Upgrades are rolled through the system one node at a time without denying services to any user. Equally important is that the system is now easily expandable by adding Windows query nodes. Expandability is simple and economic.

Performance is improved by structuring query nodes to efficiently process queries (complex read-only transactions) and by structuring the master node to efficiently process booking transactions (update transactions). One system does not need to do both (except during a master node failure). Furthermore, since query systems are deployed in the vicinity of clusters of clients, query response time is improved. Finally, by incorporating data replication, other applications are integrated with the major booking application by replicating booking data to them or to a database local to the applications being integrated.

This case study is an excellent example of HPE's move to a converged infrastructure. It demonstrates how one customer took advantage of facilities available to the HPE community to integrate diverse systems into a common application while leveraging the inherent strengths of each system.

International Partner Information

Global

Hewlett Packard Enterprise
6280 America Center Drive
San Jose, CA 95002
USA
Tel: +1.800.607.3567
www.hpe.com

Japan

High Availability Systems Co. Ltd
MS Shibaura Bldg.
4-13-23 Shibaura
Minato-ku, Tokyo 108-0023
Japan
Tel: +81 3 5730 8870
Fax: +81 3 5730 8629
www.ha-sys.co.jp

Gravic, Inc. Contact Information

17 General Warren Blvd.
Malvern, PA 19355-1245
USA
Tel: +1.610.647.6250
Fax: +1.610.647.7958
www.shadowbasesoftware.com
Email Sales: shadowbase@gravic.com
Email Support: sbsupport@gravic.com

Hewlett Packard Enterprise Business Partner Information

Hewlett Packard Enterprise directly sells and supports Shadowbase Solutions under the name ***HPE Shadowbase***. For more information, please contact your local HPE account team or visit our website.