A MAGAZINE FOR AIRLINE EXECUTIVES

2008 Issue No. 1

ascend

Taking your airline to new heights

THE PILOT

A Conversation with Tim Hoeksema, chairman, president and chief executive officer, Midwest Airlines. pg. 36

Special Section

INSIDE

26 Airlines are scrutinized for affects on the environment

44 Etihad doubles its revenue from 2006 to 2007

62 Carriers can become true customer-centric businesses

wearelistening@sabre.com

Photo by shutterstock.com

Best Of Both Worlds

By Bill Glover | *Ascend* Contributor

An airline's computing systems must provide high performance, reliability and the flexibility to change with today's business needs. The most efficient way for carriers to accomplish all three is through a hybrid solution that shares the strengths of both specialized processing and general computing.

Driving to work in an electric golf cart would be a risky choice in rush-hour traffic, but fuel costs for a huge, luxury sport utility vehicle might make the paycheck at the other end seem less worth the drive. Automobile consumers have embraced hybrid vehicles that take advantage of the power and reliability of the conventional internal combustion engine but also use a battery and electric motor to get the most out of the stop-and-go traffic of a typical commute. More than just a snazzy place to keep those organic foods bumper stickers, hybrids are cheaper to operate without sacrificing the performance and safety of a traditional automobile.

Executives in the travel industry face similar tradeoffs around computing resources where commodity hardware offers cheap computing, but powerful systems specially designed for transaction processing still dominate. Can a hybrid approach meet the needs of the travel industry for computing that is economical, scalable and reliable while still handling some of the highest transaction volumes in the world?

What Does It Take?

Getting people where they want to go takes coordination, and coordination takes communication. All of that communication creates a huge number of transactions flowing through travel IT systems — 15,000 to 20,000 transactions per second for large distribution systems. These systems need to be able to distribute those transactions, which may be long or short lived, over processors in such a way that no processor is too busy to take the load and no processor is sitting idle with nothing to do. This is more complicated than just asking the processors if they have room for another transaction. Each one of those transactions matters to someone, but they have very different priorities. A transaction related to boarding passengers needs to get through no matter how busy the system may be, while a shopping transaction might be able to a wait a short while to let more critical transactions through during especially busy periods. Making those sorts of decisions takes a smart load-balancing system that can tell with configurable priorities the difference between two transaction types and give preference to one over the other.

With so many people depending on them, travel IT systems need to be reliable. They can't be down due to a software or hardware problem. Reliable hardware is relatively easy, just buy components with a mean time between failures, or MTBF, long enough that they can be replaced according to a scheduled maintenance plan. The sort of hardware with a long enough MTBF for breathing room comes at a premium, but can pay for itself in total cost of ownership.

TPF And Commodity Systems

	TPF	Commodity
Architecture	Monolithic	Distributed
Inter-process Communication	Shared memory	Network
Scaling	Vertical	Horizontal
Acquisition Cost	High	Low
MTBF	Longer	Shorter

TPF and commodity systems differ significantly in key areas such as their architecture, scaling and costs. These differences must be considered when implementing systems designed to hold large amounts of data while operating in real time.

Avoiding software problems is much more complex. It requires careful change control and testing. Documentation has to be crisp and clear to avoid misunderstandings, and procedures must be constantly rehearsed to avoid surprises to keep a system up 24 hours a day, seven days a week, decade after decade. These things are easier to do with a specialized transaction processing system, but, these days, more and more systems are built on a commodity, general-purpose computing infrastructure.

What's The Difference?

There are significant primary differences between a specialized transaction processing facility, or TPF, system and general purpose, commodity systems:

- Architecture — A monolithic architecture puts all of the transaction processing together in one box, while a distributed architecture puts different processes in different boxes. For instance, one box may be a Web server and another a database server.
- Inter-process communication — In a monolithic architecture, the processes share the same machine and are able to communicate directly with each other using the machine's own memory. Distributed applications typically communicate over a network. Imagine a Web server connected to an application server and the application server to a database server over a local area network or the Internet.
- Scaling — Scaling refers to the way the system adds capacity to grow over time. A system that scales vertically adds more memory and more processors in the same machine while a system that scales horizontally adds more machines.
- Acquisition cost — Commodity hardware is cheaper to buy than hardened, specialized hardware.
- MTBF — Commodity hardware is designed and produced to be inexpensive and flexible, not for extended use under heavy load. Imagine running a desktop machine at 90 percent for months on end. It would quickly fail due to heat stress. Commodity servers are more robust but built using the same basic technology.

What Would It Take To Use Commodity Hardware?

Based on these differences between the two types of systems, what would it take to build a system from commodity hardware that can do the job of a TPF system? A distributed system typically is separated into tiers. The resource tier (databases) would run on a separate machine from the business logic while the presentation tier (Web pages) would operate on still another machine. This makes end-to-end transaction monitoring more difficult. The monitoring records from each machine will have to be compared and the transaction identified in some way as it hops from machine to machine. The individual systems must also share the responsibility for security. Controls on one tier will not necessarily protect another tier. There may be tight login restrictions at the presentation tier, for instance, but that won't by itself protect the database tier from being accessed by the wrong person. Throttling and smart load balancing are also more difficult with commodity systems. Typically, these systems use a hardware load balancer that understands how to distribute load according to relatively simple algorithms, but to conduct smart load balancing, the system must use software

load balancing and throttling that understands transaction priority. No single off-the-shelf system currently offers all of these capabilities, so offloading TPF requires custom development and extensive testing. The business logic must also be designed in such a way that it works well with smart load balancing and can report when a particular machine is too busy to handle more transactions. This puts some constraints on how the business logic can be distributed in the system.

To handle the high transaction volumes, a commodity system must scale horizontally. This has implications for the overall complexity and reliability of the system. Examine the low end of the range that was previously discussed for a conservative estimate, 15,000 transactions per second, and make a generous guess that each machine can handle around 100 transactions per second, sustained. Then assume 150 machines are needed for the transaction processing business logic. Adding machines for databases and Web servers, load balancers, logging monitors and a few other miscellaneous systems, one can reasonably assume 200 machines. If the MTBF for a given machine is around two years, which is not unusual in commodity hardware, a machine could expect to be replaced every three days on average as part of scheduled maintenance.

Then Why Bother?

If building a system with commodity hardware is so difficult, why bother? There are several advantages to commodity hardware. The first and most obvious is the lower acquisition cost. Another advantage is that because general purpose computers can be used in many different kinds of applications, they are more widely adopted in the IT industry. This helps make the skill pool of knowledgeable people much larger than for specialized computing systems such as TPF. Commodity hardware also tends to be more flexible and improves more rapidly. But how can the benefits of commodity computing be realized while limiting some of the liabilities?

Son Of Big Iron To The Rescue

To take on some of the jobs previously handled by TPF systems, general-purpose commodity systems have had to step up to the next level of maturity. Previous articles in *Ascend* have discussed service oriented architecture, which provides some pieces of the puzzle by externalizing security and end-to-end monitoring and providing critical buffering and self correcting throttling through pull-based messaging. Cutting-edge technologies, such as organic server management and grid computing, offer to help solve the problem of managing huge numbers of short-lived machines in such a way that they appear to be one big system. These technologies are still relatively new and maturing, and they have primarily been used for compute-intensive applications such as protein folding and rendering graphics. Systems with extremely high reliability needs, huge transaction volumes and massive input/output requirements will challenge these technologies and drive additional innovation.

Sabre Airline Solutions® has approached technology of all sorts from a strategic position that, while aggressive, values dependability and cost-effective solutions for its customers. A hybrid solution currently provides the best balance of flexibility and stability, but careful attention to the end-to-end solution positions the technology company to adopt new approaches efficiently.

The Duck Test

Those familiar with mainframes and the discussions around offloading applications from those systems may be surprised that this article hasn't used the term "open systems." The term is somewhat dated and arguably obsolete. In the 21st century, TPF can run on a commodity laptop system, and mainframes can run Linux and Java. A laptop cannot, however, handle the transaction loads of more than a modest-sized airline … yet. And it certainly does not have the MTBF required for reliability. So is a system that runs TPF at the center for transactions on top of Linux and commodity hardware and that is accessed exclusively through XML services an open system? Is an organically managed grid computer farm running on commodity hardware but presenting the behavior of a single machine a mainframe? To quote the poet James Whitcomb Riley, "If it walks like a duck and quacks like a duck, I would call it a duck." What matters is that the system does what it needs to do with high performance, reliability and the flexibility to change with the business needs of the industry. The most cost-effective way to do that today is to share the strengths of both specialized processing and general computing, a hybrid — minus the organic foods bumper sticker.

Bill Glover is chief architect for Sabre Airline Solutions. *He can be contacted at william.glover@sabre.com.*

+count it up

2.1 billion
The amount of savings in U.S. dollars that was achieved across the globe last year based on IATA's campaign to shorten routes, improve operational procedures and share best practices in fuel management.

12 million
The amount of savings in U.S. dollars by shortening routes in Bahrain, Iran and Algeria, improving departure, approach and landing procedures in Doha and Yenbou, according to the International Air Transport Association.

16.5
The percentage of airline fuel efficiency improvements since 2001, according to the International Air Transport Association.