MANAGEMENT BRIEF

VALUE PROPOSITION FOR TPF

Comparing TPF and NonStop for Travel Industry Applications

International Technology Group

4546 El Camino Real, Suite 230
Los Altos, California 94022-1069
Telephone: (650) 949-8410
Facsimile: (650) 949-8415
Email: info-itg@pacbell.net

TABLE OF CONTENTS

List of Figures

VALUE PROPOSITION

The travel industry is undergoing dramatic change. The continued growth of Web travel businesses, consolidation and the appearance of new players in online services, along with escalating cost pressures are driving far-reaching shifts in distribution structure.

Systems based on the Transaction Processing Facility (TPF) will play a critical role in these changes. In 2001, TPF-based systems, including all major geographic distribution systems (GDS) systems, processed more than 93 percent of worldwide airline bookings. This included more than 87 percent of Web bookings, along with more than 300 million hotel, car rental, railroad and other reservations.

The TPF infrastructure stands at a crossroads. Competitive agendas focus on Web marketing and service, customer relationship management (CRM), data exploitation and other new capabilities. Growth in mainframe capacity and legacy maintenance costs have become increasing concerns.

What happens next? There are two options:

1. ***Replacement***. Sabre's decision to migrate fare searching from TPF to Hewlett-Packard (HP) NonStop servers has publicized this option. But it is not clear that the cost and performance assumptions on which this project is based are correct. The company's plans involve major technical, financial and business risks.
2. ***Enhancement***. The benefits claimed for TPF replacement – including better functionality, increased performance, higher development productivity, greater interoperability and lower operating costs – can be realized through enhancements to the TPF environment. Moreover, they can be realized in a fraction of the time, at significantly lower cost.

   In an example detailed in this report, five-year costs of deploying a large-scale complex fare searching and pricing system in an efficient, properly optimized TPF environment total $83 million. A comparable NonStop-based system costs $154 million. Deployment times are less than two years, and four to five years respectively.

TPF replacement is not simply unnecessary – it will reduce competitiveness. Monolithic single-architecture solutions will reduce, not increase business agility. Multi-year "megaprojects" will tie up resources during a time of rapid industry change. Greater advantage will be realized by more flexible, multiplatform strategies that leverage TPF transaction-processing strengths while focusing new technologies on areas of immediate competitive impact.

Companies equipped for efficient, low-cost transaction processing will enjoy other competitive benefits. They will be more attractive to suppliers, travel agents, online specialists and other partners. They will also be better able to afford investments in marketing, CRM and other capabilities essential to competitiveness through all channels.

Clearly, in some companies there is a reluctance to invest in a TPF environment seen as old and unexciting. But this is a matter of mere technological image. It is of no concern to customers, partners or shareholders. No value is delivered to these by paying more to achieve inferior results.

Airlines will not pay higher fees, nor will travelers pay higher ticket prices simply because these are issued by an "open system." Winners will be companies that add new value, not extra cost.

POSITIONING TPF

Current Status

TPF has been the backbone of travel industry transaction processing since the 1970s. At least 23 travel companies worldwide, including all major GDS operators, as well as airlines, hotel chains, railroad companies and others use TPF-based systems.

Key characteristics of this user base may be summarized as follows:

- ***Volume***. TPF supports some of the world's largest transaction-processing systems. Users of TPF-based GDS have experienced peaks ranging from more than 2,500 to more than 17,000 messages per second.
- ***Availability***. TPF users report average availability levels of between 99.99 to 100 percent. In many cases, however, these figures reflect service-level commitments, or include network as well as other types of outage. Actual system-level availability is significantly higher.

  Among 14 TPF users in the travel industry who responded to a survey conducted for this report, for example, 7 experienced no significant outages of any kind during 2001, and 12 had no unplanned outages. Average availability was more than 99.998 percent overall, and more than 99.999 percent if planned outages are excluded.

  Two relatively small users accounted for a disproportionate number of outages. Among the remainder, overall averages were more than 99.999 percent, and over 99.9995 percent if planned outages are excluded. These are industry-leading availability levels for any type of system. For systems handling volumes of such magnitude, they are unprecedented.
- ***Performance***. Among companies who reported response times for their TPF-based systems, the norm was under one second and the average was around 0.4 seconds, even during peak loading with extremely high volumes.

In addition to supporting most of the world's conventional travel agents, TPF-based GDS handle bookings for all major online travel agencies, including Expedia, Travelocity, Orbitz and, in Europe, eBookers and Opodo, along with many smaller Web discounters and specialists.

The growth of Web shopping has meant dramatic increases in GDS fare searching and pricing workloads. These are, today, processed predominantly by TPF. There is no serious dispute that, in terms of functionality and performance, the TPF environment is entirely capable of meeting requirements not only for conventional transaction processing but also for large-scale, Web-initiated fare searching, complex itinerary construction and pricing.

These and other conclusions presented in this report are based on data supplied by 14 travel companies worldwide employing TPF-based systems. User experiences contradict many of the arguments advanced by vendors advocating TPF replacement. Questions are also raised about the accuracy of recent press coverage of this subject.

Replacement versus Enhancement

All users surveyed recognized that TPF was an extremely efficient and reliable transaction-processing system. None believed that it could be easily replaced. Most doubted that any other platform could seriously rival its performance in handling extremely large, business-critical workloads.

Two main problems were being experienced – and resolved – by users:

1. ***Capacity growth***. There had been rapid increases in mainframe processor capacity to process Web-initiated fare search and pricing workloads. Legacy systems were designed to support travel agents, whose searches are typically shorter, faster and more focused, and require fewer itineraries to be generated and priced. They were poorly optimized to handle the more complex workloads generated by Web shoppers.

   When Web users began to access systems on a large scale in the late 1990s, average path lengths for fare searching and pricing escalated rapidly, requiring progressively larger amounts of capacity in millions of instructions per second (MIPS). Booking functions were less affected, but also experienced growth in MIPS consumption.

   According to users, relatively simple code optimization techniques could, however, be applied to improve TPF efficiency for such workloads. Reductions in mainframe capacity utilization of up to 80 percent were reported. Such techniques can be implemented rapidly and inexpensively, and significantly reduce TPF operating costs.

2. ***Programmer productivity***. Most older TPF-based systems were developed in assembler, and were modified and maintained in this language, using relatively inefficient manual coding techniques. Application development and maintenance productivity has thus been low relative to newer high-level languages and tools.

   TPF supports use of C/C++ and visual tools, which deliver levels of development and maintenance productivity comparable to those realized for UNIX or Windows servers. Among the 14 travel industry users surveyed for this report, 11 had begun – in some cases, very recently – to employ C/C++, visual tools or both. Productivity improvements from two to five times were reported.

Two major conclusions emerge. First, companies that experience high TPF operating costs have – if only by default, by declining to invest in their TPF systems and skill bases – chosen to experience them. Problems have been allowed to develop that could have been simply resolved, and considerable savings could have been realized if corrective action had been taken at an earlier stage.

Second, TPF replacement may appear economically attractive if new systems are compared with aging, inefficient TPF environments. But the cost picture changes significantly if the comparison is with efficient, properly optimized TPF systems.

The implications may be easily illustrated. For example, respondents for one TPF replacement project reported that benefits of migrating to HP NonStop servers included a "100 percent" improvement in application development productivity, a "75 percent" increase in fare load times, and a "40 percent" reduction in total cost of ownership (TCO).

These claims are, however, based on a comparison of a new system with a 30-year-old, poorly optimized legacy system on obsolete mainframe hardware. There had been no significant changes in application structure, or in development tools and practices for literally decades.

In this case, development productivity was increased by employing C++ (which did not require rehosting). Redesigning data input processes (which could have been done as easily in a TPF environment) reduced fare load times. Search and pricing routines were redesigned to require less processor capacity (the effect would have been the same if the system had been on TPF). Code optimization would have reduced TPF operating costs further.

Figure 1 shows five-year costs for a representative large-scale complex fare searching and pricing system deployed in an efficient TPF environment, and on HP NonStop servers. The basis of these calculations is detailed in the case study section of this report.

Figure 1

Comparative Costs for Complex Fare Searching and Pricing System: NonStop and Efficient TPF Scenarios

	Year 0	Year 1	Year 2	Year 3	Year 4	Year 5	Total
NONSTOP SCENARIO							
Hardware acquisition	–	37,620	7,524	9,405	11,286	13,167	**79,002**
Hardware maintenance	–	1,806	2,167	2,618	3,160	3,792	**13,543**
Software acquisition	–	12,390	2,478	3,098	3,717	4,337	**26,020**
Software support	–	620	743	898	1,084	1,301	**4,646**
Subtotal ($ thousands)	–	**52,436**	**12,912**	**16,019**	**19,247**	**22,597**	**123,211**
Application development	8,580	–	–	–	–	–	**8,580**
Application maintenance	–	1,030	1,030	1,030	1,030	1,030	**5,150**
Admin/operations/support	–	3,536	3,536	3,536	3,536	3,536	**17,680**
TOTAL ($ thousands)	**8,580**	**57,002**	**17,478**	**20,585**	**23,813**	**27,163**	**154,621**
EFFICIENT TPF SCENARIO							
Hardware acquisition	–	7,258	1,469	1,469	1,895	2,032	**14,123**
Hardware maintenance	–	301	349	408	484	565	**2,107**
Software	–	5,407	6,751	8,095	9,811	11,647	**41,711**
Subtotal ($ thousands)	–	**12,966**	**8,569**	**9,972**	**12,190**	**14,244**	**57,941**
Application development	4,719	–	–	–	–	–	**4,719**
Application maintenance	–	1,888	1,888	1,888	1,888	1,888	**9,440**
Admin/operations/support	–	2,184	2,184	2,184	2,184	2,184	**10,920**
TOTAL ($ thousands)	**4,719**	**17,038**	**12,641**	**14,044**	**16,262**	**18,316**	**83,020**

These figures assume that systems are fully implemented and operational at the beginning of the five-year period, and thus understate the costs of the NonStop scenario. In practice, while workloads are phased over, it is necessary to allow for continued operation of the legacy TPF-based system – with rapidly escalating MIPS consumption – over a multi-year period. It could also be expected that NonStop deployment would entail extensive outside services assistance during this transition.

From this perspective, the business case for replacing TPF is somewhat less than clear-cut. It is not correct, as some critics have claimed, that new technology capabilities cannot be implemented in a TPF environment. Claims that replacement will reduce costs neglect the significant – and more easily realized – potential for gains through TPF enhancement.

Industry-wide, replacement of TPF-based systems would – even using conservative assumptions – require at least $3 to $5 billion in new IT investment over 5 to 10 years, equivalent to between $0.25 and $0.60 for each ticket processed over this period. There would, under any scenario, be major risks of delays, cost overruns, performance bottlenecks and outages affecting all users.

Replacement assumes, moreover, that other platforms could handle TPF-equivalent workloads while maintaining adequate performance and 24x7 availability. It is doubtful whether today's UNIX servers could do so, even if these were clustered and configured with high levels of redundancy. It is virtually certain – as the experience of Expedia and others have shown – that Windows servers could not.

The IBM zSeries mainframes running z/OS and/or Linux, and the HP NonStop platform are more viable for high-volume 24x7 operations. But replacing TPF with NonStop servers would mean moving into uncertain territory. The characteristics of large-scale TPF-based travel systems are significantly different to those for which NonStop systems are employed in other industries. There is little experience with high-volume transaction processing systems employing NonStop SQL.

The risks of rehosting to HP NonStop servers are, moreover, magnified by HP plans to start transitioning this platform from its current RISC base to Intel Itanium CPUs starting in late 2004 – i.e. when any TPF replacement projects started in the near future would already be well advanced. It will be necessary to recompile all NonStop code developed up to that point. The transition also involves broader technical risks that are detailed later in this document.

Open Systems

One final argument in favor of TPF replacement – that it is necessary in order to create an "open system" – should be addressed.

"Open systems" – in normal industry usage of this term – refers to support for industry standards in order to ensure application portability, and application and data interoperability. TPF supports all major industry standards required to achieve these goals in a typical travel environment – including C++, POSIX, TCP/IP, Web mail, SQL, Internet Inter-Orb Protocol (IIOP) and Extensible Markup Language (XML). TPF is thus, by any realistic definition, neither more nor less "open" than HP NonStop or most UNIX implementations.

It is curious to see HP NonStop systems characterized as more "open" than TPF. The NonStop environment is built around a proprietary operating system, NonStop Kernel (NSK) and a unique system architecture built around dual fault-tolerant microprocessors and parallel processing. It includes a non-standard database – NonStop SQL – with few users and little third-party support. None of the major components of the NonStop software environment runs on any other platform.

In practice, however, it is not necessary to standardize underlying systems software and architecture in order to realize portability and interoperability. These are implemented through standardized application-level interfaces, enabling users to create heterogeneous environments employing a range of technologies and platforms for different applications.

This approach – which is being adopted by most TPF users – enables companies to exploit TPF transaction-processing strengths while deploying UNIX and Windows servers for a wide range of new technology marketing- and service-oriented applications. In practice, it will be more flexible. It will also allow users to employ relational databases such as DB2 and Oracle which are de facto industry standards, and for which third-party applications and skills are widely available.

BUSINESS VIEW

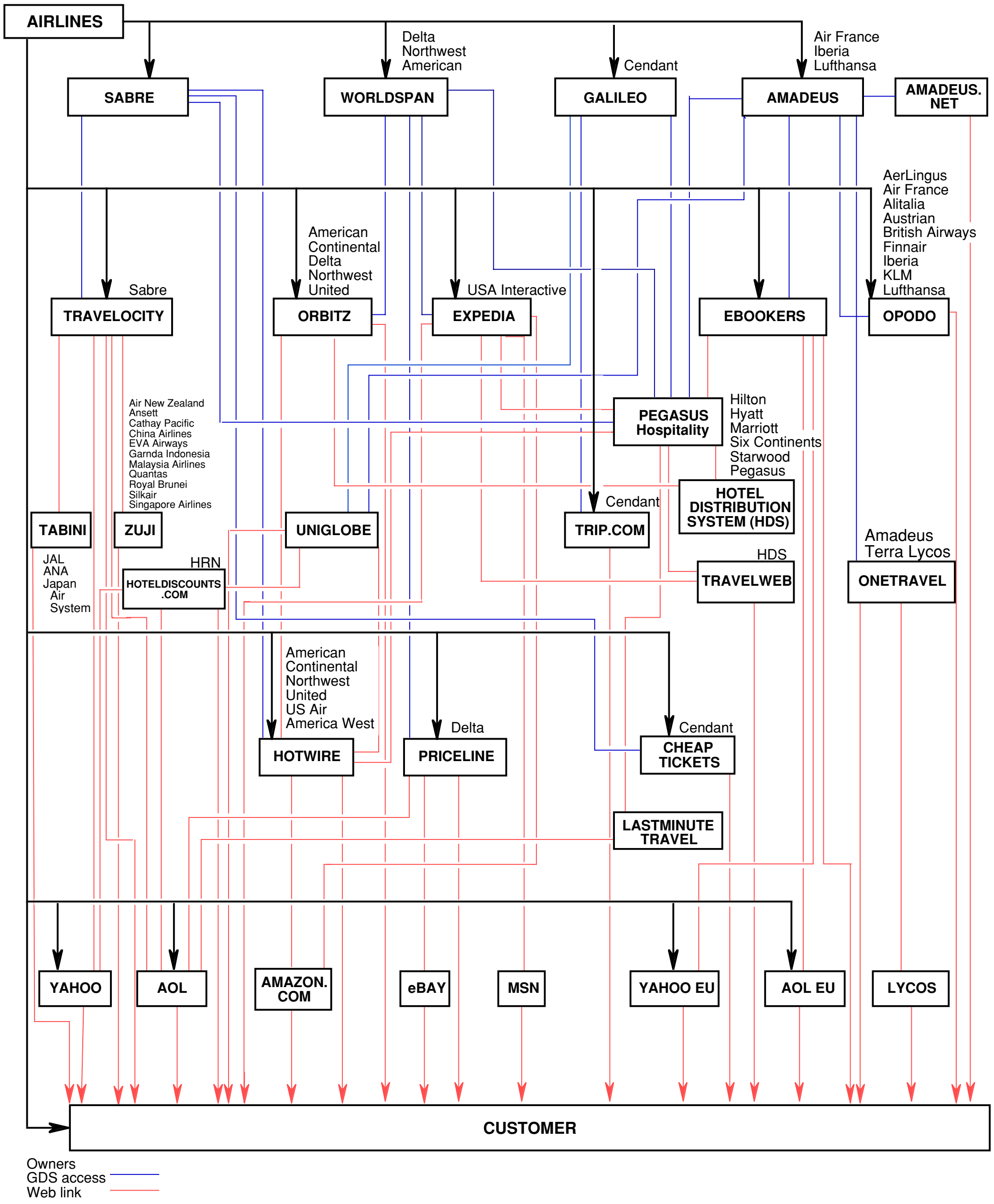
Change and Flexibility

TPF runs systems whose workload impacts and service level requirements – large transaction volumes, real-time processing, fast response times and 24x7x365 availability – are highly demanding. Even if design goals and technology assumptions were realistic, replacement of these systems would lock up resources, dominate IT strategy and determine organizational focus for periods of five years or more.

This would occur during a period that is going to see rapid and dramatic change. The industry's distribution structure, which is partially illustrated in figure 2, has become progressively more complex and unstable as successive waves of new players have entered the market.

Figure 2

Representative Industry Distribution Affiliations



One of the most striking characteristics of this structure is that, despite predictions that the Web would lead to large-scale “disintermediation” (i.e. suppliers would bypass travel agents and sell directly via the Web), it has instead created entire new sectors of intermediaries. These have included major online travel agencies such as Expedia, Travelocity and (in Europe) eBookers, as well as a plethora of discounters and specialists.

Airline-backed Web sites such as Hotwire, Orbitz, Opodo, and more recently Zuji and Tabini have also appeared on the competitive landscape, along with the Hotel Distribution System, backed by five of the world’s largest hotel chains. (Orbitz, in particular, emerged virtually overnight as one of the industry’s principal online players.) More such ventures are expected.

Airlines, hotel chains, car rental firms and other suppliers have continued to target aggressive growth through their own Web sites. Conventional travel agencies have, increasingly, developed Web presence. The field has also expanded to include an entire new tier of Web players such as broad-based portals (e.g. AOL, Yahoo, MSN), online retailers (e.g. Amazon.com) and auction sites (e.g. eBay). Clearly, this process of diversification still has a long way to go.

Diversification by existing companies and the appearance of new players such as Cendant and USA Interactive, which are seeking to assemble broad portfolios of online businesses, are contributing to consolidation. Figure 3 shows representative examples of this process.

Figure 3

Industry Consolidation Examples

USA INTERACTIVE

- **Expedia (OLT)**
  - **TravelScape (LDG)**
  - **VacationSpot (VTC)**
- **Hotel.com (LDG)**
- **Hotel Reservations Network (LDG)**
- **Ticketmaster/CitySearch**
- **Home Shopping Network**
- **TV Travel Group (UK)**

CENDANT

- **Avis**
- **Hotel franchises***
- **Galileo (GDS)**
- **Trip.com (OLT)**
- **Cheap Tickets (LMD)**
- **Highwire (CBT)**
- **WizCom**

SABRE

- **Sabre (GDS)**
- **Travelocity (OLT)**
- **GetThere (CBT)**
- **Site59 (LMD)**
- **VirtuallyThere**

WORLDSPAN

- **Worldspan (GDS)**
- **Worldspan GO! (VTC)**
- **Worldspan TM (CBT)**

AMADEUS

- **Amadeus (GDS)**
- **Amadeus.net (OLT)**
- **e-Travel (CBT)**
- **Vacation.com (VTC)**
- **Amadeus Cruise (VTC)**

GDS	Global distribution system
OLT	Online travel agency
LMD	Last minute/discount travel
VTC	Vacations/tours/cruises
CBT	Corporate/business travel
LDG	Lodging

*Includes Amerihost, Days Inn, Holiday Cottage, Howard Johnson, Knights Inn, Ramada Inn, Super8, Travelodge, Villager & Wingate Inn

Other factors add to instability. Consolidation among Web players has accelerated as the dot-com boom has wound down. Boundaries between business and leisure travelers, and between airline, hotel, and other travel businesses continue to erode. U.S. players are expanding into Europe and the Pacific region. Antitrust concerns may affect new ventures. eTicketing raises a range of new issues.

Under any scenario, the next three to five years will be a time of far-reaching change in the travel industry’s distribution structure. Strategies based on systems that will not be implemented until the end of this period represent obvious dangers. The business and technology assumptions on which they are based may all too easily change in the meantime. More diverse, multiplatform strategies IT will offer a great deal more flexibility.

Economic Models

Continued economic weaknesses and the slowdown in air travel since the events of September 11th 2001 have increased pressures on an already stressed industry.

When Web travel began to develop in the mid-1990s, it was widely expected that the industry's distribution structure would move toward a simpler economic model. Suppliers would, increasingly, deal directly with customers via the Web, mid-tier players would be progressively undermined, and the fees and commissions paid to these would decline and eventually disappear.

A number of factors have qualified this picture. Conventional travel agents have continued to play a major role. Online players have represented a useful channel for discounted inventory. Antitrust concerns have limited – and may limit even further – the competitive potential of supplier-backed Web sites. The Web travel business has also proved sensitive to marketing and customer service, in which online players have often been able to build strong capabilities.

Another factor has been the economics of back-end infrastructures. Although suppliers remain dissatisfied with GDS fees, the costs of building equivalent systems, and the probability that they would be obliged to offer equal access if they did so act as a deterrent to more aggressive bypass strategies. For online independents, building GDS-equivalent systems is an equally unattractive proposition. The result has been an uneasy and contentious coexistence.

In this environment, competitive strategies have tended to converge. Suppliers, as well as conventional and online travel agencies, and specialists are moving toward a common business model. All are focusing on building and leveraging brands; and developing direct customer relationships through targeted marketing, loyalty programs, creatively packaged specials and deals, customization, quality customer service and similar techniques.

In IT terms, these translate into capabilities customers utilize directly via the Web such as searching, custom itinerary creation and pricing and personalization; customized email newsletters, alerts and notifications (including wireless messaging); CRM; and data exploitation systems for analytical and real-time marketing applications.

But such business models involve high levels of marketing and service expenditure, and the solutions that support them are far from inexpensive. Most players will be faced with significant increases in IT expenditure. Experience in other sectors of Web commerce has, moreover, shown that high levels of technology investment must be maintained almost indefinitely.

Premium capabilities rapidly become the norm. Others may copy successful techniques almost immediately. Content and functionality must be continuously upgraded. Quality of service – including 24x7 availability, response time, service time (end-to-end job throughput time) and security – must be maintained while deploying increasingly complex and sophisticated technologies requiring progressively greater amounts of processor power and bandwidth.

The question is: who pays for all this? Does the intermediary absorb the additional cost? Or is it passed on to suppliers, or to customers, or to both?

Suppliers will clearly resist paying higher fees and commissions, and will respond with more aggressive bypass efforts. Equally, it is unclear what kind of premium customers might be prepared to pay for, say, personalization or loyalty perks once the cost of these becomes explicit. Much of the appeal of Web travel has been based on lower costs.

Individual companies may be successful with marketing- and services-centric strategies. But the distribution structure clearly cannot sustain major, across-the-board increases in IT expenditure within each tier. And it is unlikely that, under any scenario, customers will accept pervasive ticket price inflation.

The implications are important. While customer-facing capabilities will clearly remain critical, long-term economic pressures suggest that companies should not simply relegate their transaction-processing infrastructures to the “back burner” in terms of IT priorities. Nor should they assume that escalating expenditures on solutions for Web shoppers will deliver high levels of payback indefinitely. Cost containment in IT, as in other areas of the business, will be mandated.

For GDS operators, the quality and cost-effectiveness of transaction-processing infrastructures will be particularly important in that, while direct Web shopping may enjoy higher growth rates, conventional travel agent business will remain significant for the foreseeable future. For this group, forms of access to GDS will become more sophisticated over time. But this is a significantly different proposition from the Web shopper-centric approach adopted by a number of players.

One final point should be noted. In an industry undergoing major structural change, the possession of an efficient, low-cost transaction-processing infrastructure may in itself be an increasingly important edge. For most players, competitive success or failure will be strongly affected by the ability to attract, retain and build relationships with partners – who will be concerned with quality and cost of service. It is unlikely they will care much about the underlying technologies that deliver these.

In a recent presentation, the CIO of a major GDS company posed the question: “What value do I add to my customers by spending $100 to $200 million simply to replace TPF?”

This is the heart of the issue. Rebuilding existing TPF-based systems around new platforms would inevitably be an exercise in “reinventing the wheel.” Once the process was completed, the company might be more competitive than it was. But this is not the bottom-line standard of comparison. The question is whether better results might have been achieved through alternative approaches. In the case of the TPF enhancement scenario outlined in this report, the answer is clearly yes.

TECHNOLOGY VIEW

Why TPF?

Overview

Most of the world's large reservation systems are mainframe-based. The largest are TPF-based. Although these have in some cases been in place since the 1970s, they have proved remarkably resilient. While several have been consolidated or replaced because of mergers and acquisitions, none have been phased out in favor of systems running on other platforms.

TPF has retained a particularly strong position in the airline industry. Its only real competition has been from Unisys mainframe-based reservation systems operated by a few European carriers. A number of low-end systems based on UNIX servers and (in one case) the HP 3000 platform have been deployed by small airlines, but these have not posed a serious alternative for larger players.

The longevity of TPF has, to some extent, been due to companies preferring to avoid the costs and difficulties involved in replacing large legacy systems. But primarily it has been because TPF is highly optimized for high-volume, business-critical applications.

The key point is that the TPF environment possesses strengths in system architecture, and in the implementation of underlying hardware and software components which – to a much greater degree than competitive platforms – map to the unique requirements of large-scale business-critical transaction-processing systems in general, and of travel industry systems in particular. The viability and cost of potential replacement solutions cannot be properly assessed without understanding these.

General Principles

Any high-end business-critical transaction-processing system must meet a diverse set of requirements. It must be able to handle very large transaction volumes, particularly at times of peak loading. It must, at the same time, maintain: fast job throughput and response times; 24x7 availability, including avoidance of planned as well as unplanned outages; data integrity; continuous logging (for record-keeping purposes as well as to ensure recoverability of data); and security.

The system must also be capable of immediate, reliable (i.e. records are not lost or corrupted) failover in case an outage disables an individual processor, system or data center. The process must occur in a manner invisible to users and applications, so no disruptions to normal service occur.

Realization of any one of these capabilities is a demanding process, placing a great deal of stress on application structures, as well as on underlying systems software and hardware platforms. The realization of all of these capabilities, simultaneously, is significantly more difficult. Challenges are cumulative.

All service quality variables – including availability, job throughput (often referred to as "service time"), response time, data integrity, logging, security, and failover and recovery capability – are a great deal more difficult to maintain in a high-volume environment than in one with low- to mid-volume transaction workloads.

Challenges increase, moreover, in a manner that is closer to exponential than arithmetic. Maintaining 99.99 percent availability with a workload of 1,000 messages per second is not twice as difficult as with 500 per second. It may, depending on application and workload characteristics, be orders of magnitude more difficult.

Failure to allow for these effects is one of the most common causes of problems among organizations deploying new transaction-processing systems. Most industry experience with high-volume transaction processing has been in mainframe legacy environments. Designers familiar only with UNIX or Windows servers, relational databases, and tools and languages commonly employed with these often do not understand – and seriously underestimate – the difficulties of implementing such systems.

Reservation Systems

The challenges of delivering adequate performance and service quality for the workloads handled by large TPF-based travel systems are among the most daunting to be found in any industry.

A reservation system operates in near real-time. Fares must be current, and availability must be checked and reservations made extremely rapidly to avoid the same seat (or room, or vehicle) being assigned to multiple customers.

GDS databases contain up to 40 to 50 million fares, with 100,000 to 300,000 fare changes on a typical business day, and up to two million during specials. This routinely translates into billions of fare calculations per day. With the growth of Web shopping, workloads expand dramatically. Workloads originally based on searching for, generating, checking availability and presenting, say, 3 itineraries may expand to 10 or 20 itineraries. The effect is, again, near exponential.

This type of workload generates extremely high external I/O rates, and requires very rapid response times. But internal impacts are even greater. A GDS may, for example, hit peaks of 2,000 to 20,000 messages per second. Depending on workload mixes, interactions within the system itself – including communications within and between processors – may translate into the equivalent of 10,000 to more than 200,000 messages per second.

It is the larger figure – the internal throughput rate, rather than the external interaction rate – that is the real measurement of the performance a system must deliver. The impact on any platform will be enormous. Failure to allow for this effect contributed to the failure of early attempts to replace TPF. It is not clear that it is understood by users seeking to replace TPF-based systems today.

In order to support intra-system workload volumes of this magnitude, it is not sufficient simply to employ high-bandwidth fiber optic links. System architecture, and all hardware and software components within the system, must be optimized for extremely high-speed, reliable interaction.

One of the major implications is that dividing a system – e.g. by moving fare searching and pricing to a different platform – risks creating significantly latency problems even if high-bandwidth networks are used. If bottlenecks are to be avoided, the entire hardware and software complex of the new platform must be capable of at least the same level of throughput as TPF. It is, again, unclear whether this effect is properly understood among companies seeking to migrate TPF workloads.

The TPF operating system is built around an extremely lightweight process model – i.e. internal as well as external interactions generate minimal overheads. If the system is properly optimized, both capacity consumption and latencies are kept low. Most other operating systems generate significantly greater overheads. These may be further magnified by systems software, database and application structures, and by languages and types of middleware employed to implement them.

Replicating such capabilities is, as companies attempting to replace TPF systems have found, neither simple nor inexpensive. Even if platforms of choice are capable of handling TPF-equivalent workloads – which, for larger systems, is a still unproven proposition – it would be necessary to employ a great deal of processor power and run servers at low levels of utilization to avoid bottlenecks. Existing TPF-based systems can run efficiently at upward of 90 percent utilization.

Maintaining high levels of availability would, similarly, require extensive investment in database and server clusters. Logging and security, which are handled in an extremely efficient manner by TPF, would generate significantly larger system overheads in most other environments. These effects would push capacity requirements further. Companies that fail to understand and allow for these effects are likely to experience early cost overruns as well as performance disappointments.

Data Structures

It is commonly argued that a relational database is necessary to process Web fare searches. There is some logic to this. The relational model and the structured query language (SQL) that implements it were designed to handle complex queries. TPF data structures are geared to the shorter, smaller and faster data interactions that occur in transaction processing.

The argument that TPF cannot handle complex queries is, however, incorrect. TPF-based systems already handle large-scale Web shopping queries. The issue is thus one of relative efficiency rather than functional capability.

In reservation systems, databases experience extremely large numbers of concurrent accesses. The records searched for are relatively small, and search, itinerary, availability checking, presentation and booking processes are a great more time-sensitive than in most relational applications.

The overall workload model is thus close to conventional transaction processing. TPF data structures will, as long as they are properly optimized, be at least as effective, and probably a great deal more effective, than relational approaches for internal system processes.

Relational databases generate significantly higher system overheads, so that more processor power is required to avoid performance degradation – if, indeed, degradation can be avoided at all. This is particularly the case if two-phase commit protocols are employed. Data logging, assurance and management functions also generate significantly higher overheads than in a TPF environment.

These effects may not be serious – or may at least be acceptable – in low- to mid-volume transaction-processing systems. But they have a much greater impact on performance in the extremely high-volume, response time-sensitive environments that characterize high-end TPF workloads.

This is why most of the world's very large transaction-processing systems continue to run on mainframes employing hierarchical (e.g. IMS, IDMS) and flat file (e.g. VSAM) data structures. Similarly, and for the same reasons, most large NonStop-based transaction-processing systems employ flat files (Enscribe files) rather than the HP NonStop SQL database.

Multiplatform Architectures

In most companies that employ TPF, core TPF-based systems coexist with a range of different platforms, including UNIX and Windows servers, employed for other applications.

It is expected that this will continue to be the case, as companies expand capabilities in such areas as Web applications, CRM, business intelligence (data warehouses, data marts and other types of solutions employed for data analysis applications), partner interfaces, and real-time marketing and service. Many of these will employ relational databases, along with object request brokers (ORBs), Java and other new technologies.

As long as SQL interfaces are in place, it will be possible to extend queries into TPF-based systems, and to extract TPF records and transfer these to relational systems either in batch mode or real time.

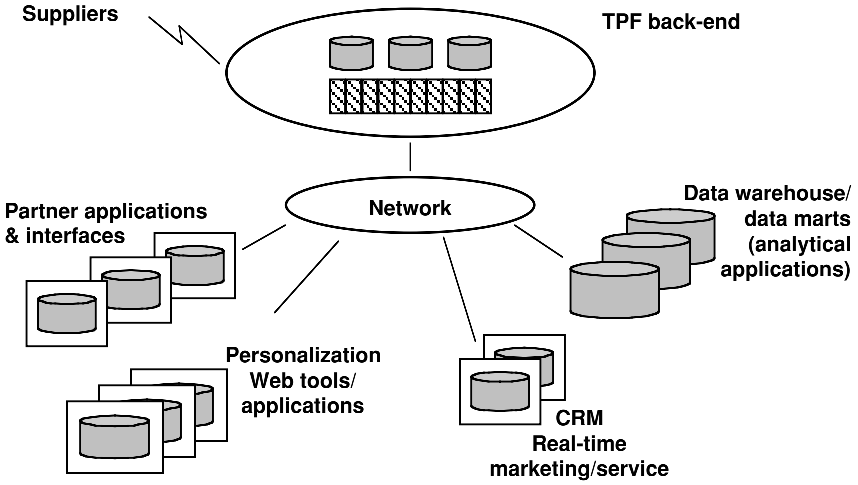
Interoperability with object-oriented systems, including CORBA- and Java-based systems, may be provided through IIOP. TPF may also interoperate through IP protocols, XML, Simple Object Access Protocol (SOAP) and others.

It is not necessary to embed new technologies into the transaction-processing system itself, and it would be unwise to do. Relational databases, along with ORBs, Java, artificial intelligence (for predicting and assisting user fare queries) and other high-end tools would all increase system overhead dramatically, as well as creating security and stability risks.

A less tightly integrated environment built around industry standards – such as that shown in figure 4 – will offer better performance and greater flexibility.

Figure 4

Representative Multiplatform Environment



This approach is a great deal more realistic than attempting to construct “megasystems” which would fuse the capabilities of a world-class GDS with a full suite of new technology implementations, including relational database, native CORBA and Java geared to Web users.

Such attempts would be among the most technologically ambitious projects ever undertaken in the travel industry, or indeed any other industry. It would be necessary to reconcile the characteristics of fundamentally different types of system architecture – transaction-processing, query-intensive and Web-oriented – and deal with the effects of massive performance degradation in one of the world’s highest-volume computing environments.

Under any scenario, this would be an extremely risky undertaking. But the degree of risk will be magnified if the NonStop platform were employed.

NONSTOP ISSUES

Risk Factors

Moving a large-scale, business-critical transaction-processing system from the mainframe environment to an alternative platform will – as more than a decade of industry experience has demonstrated – be a high-risk proposition under any scenario.

But moving to the NonStop platform represents certain unique risks that should be properly evaluated before any commitment is made. Three factors should receive particular attention:

1. ***Deployment experience***. NonStop systems are widely used for high-volume transaction processing. They are not, however, widely used in this role with the NonStop SQL database, which is employed by a minority of NonStop users primarily for data warehouse and operational data store applications.

   There is no real base of experience with large-scale, business-critical NonStop SQL-based transaction-processing systems. The most ambitious attempt to deploy such a system – by the California Department of Motor Vehicles (DMV) in the 1990s – was not completed.

   Any GDS implementation would be by a wide margin, the largest NonStop SQL deployment ever attempted. Its workload characteristics will also be significantly different to any previous NonStop SQL implementation of any size. The company conducted tests with a limited-function prototype of its new fare search and pricing application, using NonStop systems with approximately three percent of the processor capacity of the projected final configuration.

2. ***Vendor commitment***. The NonStop environment is supported predominantly by HP. There is little third-party applications software or tool support for the NSK operating system, NonStop SQL database, ZLE enablers and other essential NonStop software components.

   This in itself would not necessarily disqualify NonStop servers from a strategic role in travel companies. Similar comments can be made about TPF. But, despite much talk about "open systems," the reality is that moving from TPF to NonStop is a move from one non-standard environment to another. In both cases, users are fundamentally dependent on a single supplier for solutions, skills, and the platform's continued technological vitality.

   HP's commitment to the NonStop platform does not necessarily engage the company as a whole. NonStop systems are the responsibility of a business unit that represented approximately $2 billion in product and service revenues during 2001, equivalent to around three percent of the combined HP and Compaq revenue total. If the NonStop business does not meet HP goals, there is no obvious reason to believe that the company would underwrite its continuance.

   There are a number of precedents for such action by HP's current management team. In November 2001, for example, the company announced that it would phase out its proprietary e3000 midrange system. An earlier commitment to migrate the e3000 MPE operating system to Itanium was reversed. There were, according to industry estimates, more than 20,000 e3000 systems still in use at the time. Management cited poor growth prospects.

3. ***Itanium transition***. HP plans, starting in late 2004 to transition the NonStop environment from its current RISC CPU base to the Intel Itanium. It will be necessary to recompile all software written for MIPS-based models.

> HP projects that recompilation will be a comparatively simple process. It will, however, inevitably be disruptive, particularly in that TPF replacement projects started in the near future will be well advanced. The fact that significant changes to underlying software will be required to run on a new CPU platform will tend to increase risks of bugs and instabilities.
>
> The potential risks of the MIPS-to-Itanium transition extend, however, well beyond the need to recompile. It is still far from clear that the Itanium program will deliver adequate performance for high-end NonStop applications. The difficulties of implementing the idiosyncratic NSK operating system on a relatively new, highly complex CPU design will be far from negligible. There are substantial risks of delays, shortfalls and glitches.
>
> There are, moreover, signs of a shift in Intel commitment. HP's strategy for Itanium migration was originally based on the proposition that Itanium would combine high-end, RISC-type performance with a "commodity" economic model. The central assumption was that Itanium development, engineering, manufacturing and support costs would be low because it would be produced for core Intel volume businesses.
>
> Intel's core PC business, however, is clearly more receptive to simpler designs that better preserve compatibility with x86 software. Responding to market demand, Intel is now emphasizing a new "commodity" line of Prescott CPUs that most industry observers expect will undercut demand for, and increase the costs of Itanium engines. Itanium economics would thus be a great deal less attractive for HP, and for the industry as a whole.
>
> HP could fall back on continued use of MIPS CPUs. But this would not be a viable long-term option. The MIPS platform is used primarily for low-end embedded controller applications, and its future evolution will be geared to these. It will continue to lag competitors in performance. Its longevity is also heavily dependent on a single large OEM customer – Nintendo – whose continued commitment cannot be regarded as certain.

Each of these factors represents significant risk. But their combined impact could be much greater. It is easy to envisage scenarios in which project failures, transition delays and higher-than-expected costs would undercut NonStop demand. This in turn could undermine HP's commitment.

NonStop systems are not a core HP business. The company's UNIX-based high availability solutions represent significantly greater revenue and profit potential than NonStop servers, and there would probably be a great deal more demand for ZLE solutions built around Oracle and mainstream HP server offerings. The NonStop business, NonStop SQL, or both could be phased out or spun off.

Such actions by HP would at least offer users clear-cut options. A potentially more disruptive scenario would involve a sequence of postponements. Itanium-based NonStop models might be delivered late, or with performance and stability problems that required users to stretch deployment of new NonStop-based systems beyond original expectations.

One of the most frustrating aspects of such delays would be uncertainty as to their duration. If HP itself were unable to adequately project milestones, users could be faced with a series of "leapfrogs" (e.g. the transition would be refocused from Madison to a next-generation Itanium CPU, then from that to the succeeding generation). This would make planning an erratic and uncertain process.

Itanium Transition

CPU Base

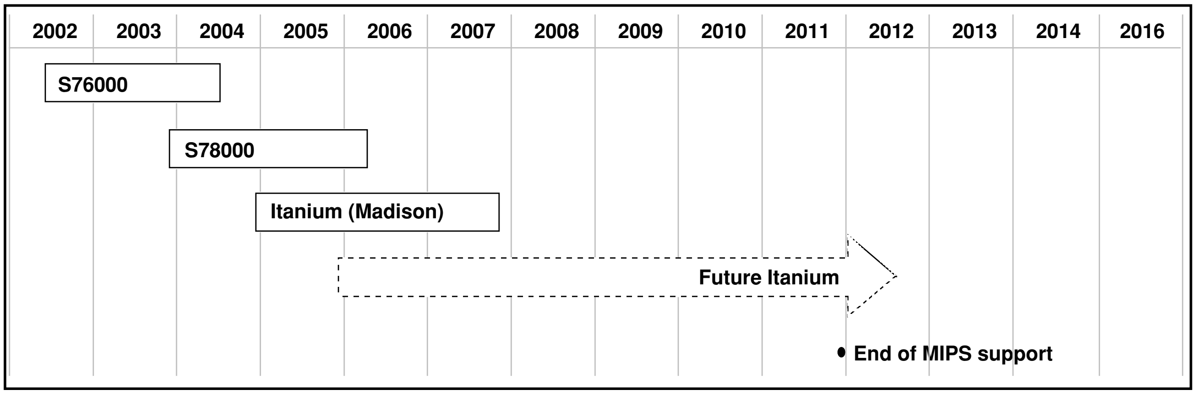
NonStop systems have, since the early 1990s, been based on OEM MIPS Technologies RISC CPUs. The performance of these has consistently lagged competitive RISC designs from Compaq (DEC Alpha), HP (PA RISC), IBM (POWER) and Sun (SPARC and UltraSPARC).

Following its acquisition of DEC in 1998, Compaq announced that the NonStop platform would transition to Alpha CPUs. In June 2001, this was changed to the Intel Itanium. Transition to the latter would start with new “G Series” NonStop models based on the Intel Madison CPU to be introduced in late 2004. These will be followed in 2005 and later by models based on next-generation follow-ons to Madison.

Only two further MIPS-based refreshes would be introduced during 2002 and 2003 (S76000 and S78000 models respectively). Support for MIPS-based architecture would be phased out by 2011. This schedule, which has been confirmed by HP, is summarized in figure 5.

Figure 5

Schedule for NonStop Transition from MIPS to Itanium CPU Base



Although the Itanium CPU platform is enabled for 64-bit addressing, this will not be immediately exploited by NSK. Early Itanium compilers and applications will continue to employ 32-bit addressing. According to HP, 64-bit application addressing will “be implemented in phases” in later NSK releases.

Design Complexity

Itanium incorporates features that are not only new to the Intel world, but also to the industry as a whole. It is a highly ambitious design. Its Explicitly Parallel Instruction Computing (EPIC) structure – jointly developed by HP and Intel – involves design compromises intended to combine high-end RISC-equivalent performance, backward x86 compatibility and commodity characteristics.

Experience with such concepts has not been encouraging. The HP-Intel cooperation began in 1994. HP initially expected to begin transition its PA RISC platform to the first IA-64 CPU, Merced, in the 1997 to 1998 timeframe. However, Merced performance proved to be inadequate, and the effort was refocused on McKinley. McKinley performance also proved to be inadequate, and the effort was refocused on the Madison CPU. Migration is expected to occur after 2003.

In its attempt to reconcile conflicting design goals, Itanium has become – by industry standards of microprocessor design – an unprecedentedly complex architecture. The EPIC structure, which breaks instruction code into multiple parallel streams, contrasts with the simpler use of a sequential instruction stream by Pentium CPUs.

The large number of performance variables inherent to Itanium design means that a significantly greater (relative to both Pentium and RISC engines) role is played by compiler technology in overall performance optimization. Itanium compilers must be larger, their designs more sophisticated, and their implementation more elegant than for other platforms. They also require exceptionally high levels of tuning.

Itanium's sensitivity to compiler-level optimization has been reflected in early benchmarks for this CPU. Initial results vary widely, and multiple iterations are typically required to exploit Itanium performance potential properly. According to some estimates, adequately tuning compilers may take two to three years. In addition, questions are raised about how viable highly tuned hardware and software configurations would be in realistic operating conditions.

NonStop Implementation

Most of the planned migrations from RISC to Itanium, including those of HP-UX, Linux and Open VMS involve transferring relatively simple symmetric multiprocessing (SMP) software architectures. NonStop migration, however, will pose more complex challenges.

It will be necessary to transfer a non-standard, significantly more complex dual-microprocessor architecture with highly proprietary lockstep code functions, and an equally non-standard massively parallel processing (MPP) operating environment. This would, under any scenario, be a highly challenging exercise. Itanium's complexity and instability render it a great deal more so.

Transitioning software, which must deliver 24x7x365 availability using relatively new, highly complex, largely unproven compiler technology, would be a risky undertaking under any scenario. Risks are compounded by the speed at which it is proposed to complete the transition, and by the unique characteristics of the NSK environment itself.

It will, in particular, be necessary to deliver lockstep capability with relatively new, still evolving compiler technology that is a great deal more complex than anything dealt with before in the NonStop world. Assuming that a stable implementation were achieved at an early stage of development, extensive testing and tuning would still be required to avoid software glitches and bottlenecks.

Uncertainties about Itanium performance are reinforced by the unique challenges of NonStop implementation. There is an obvious risk that G Series models will not deliver adequate performance, or at least will not do so until some time after migration to this platform begins.

HP has projected that Madison performance in an NSK environment will, circa 2004, be approximately 60 percent higher than for the MIPS-based S78000, and that a further 60 percent increase will occur with the Madison follow-on. Such projections must be regarded as speculative.

CASE STUDY

Overview

This case study is of a large GDS operator seeking to replace TPF for a new fare search and pricing application designed for Web shoppers. It is based on data supplied by multiple users of TPF-based systems, including GDS users, and its contents do not necessarily correspond to the experience of any actual individual company.

The case study highlights the issues raised in choosing between TPF replacement and enhancement. The company was dissatisfied with the limitations of a 30-year-old TPF-based system. However, these limitations were largely due to the company's reluctance to invest in this system, or in the skills of the personnel engaged in maintaining it.

The company decided to deploy the new fare search and pricing application on HP NonStop servers. It did not, however, seriously evaluate TPF enhancement opportunities. Moreover, its cost/benefit justifications for replacing the existing TPF-based fare search and pricing capability did not allow for the potential effects of these.

TPF replacement was originally conceived as a five-year project. Initial planning began in 1999, and a six-month "proof of concept" was conducted in cooperation with the then Compaq NonStop organization during 2000. Deployment was scheduled to begin in 2001, and to be completed by 2005.

This schedule has since slipped by at least twelve months, and further delays are likely. Performance bottlenecks have emerged, and the original system design is under review. Cost projections have been revised upward.

Although the company has to date focused on fare search and pricing, a larger management vision has affected decision-making. NonStop servers would eventually host not only the entire GDS, but also the core systems of the company's Web travel businesses. Volume efficiencies, 24x7 availability, and real-time information availability would help create a competitive edge, protecting the company from disintermediation and competitive cost pressures.

A certain irony is apparent. This vision could be achieved, more rapidly and cost-effectively, and with greater odds of success, if mainframes rather than NonStop servers formed the basis of the company's core infrastructures. That opportunity is open to its competitors.

Background

GDS History

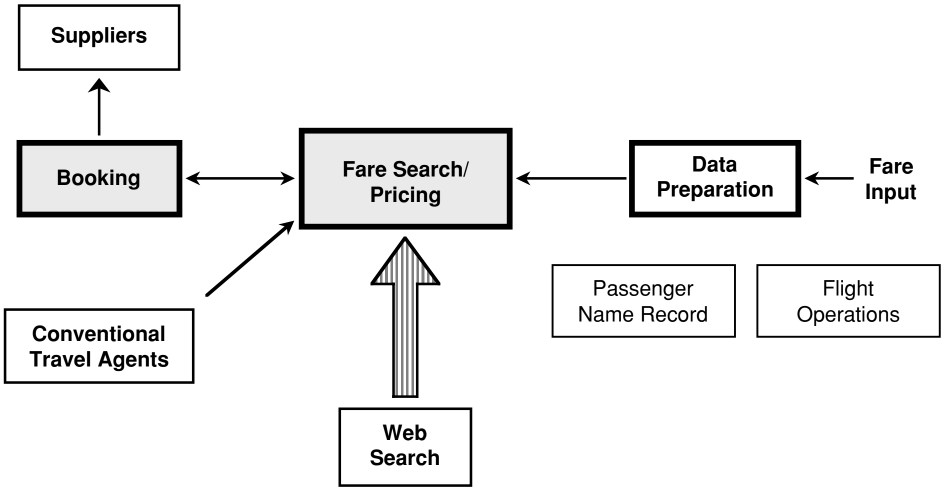
The company's GDS was originally developed in the 1960s, and was later substantially upgraded during the 1970s and early 1980s. Thereafter, enhancements were largely incremental. By 2000, it consisted of more than 10 million lines of code, primarily written and maintained in assembler, but including some C++ components.

The system was designed for, and for most of its history has been used predominantly by conventional travel agents accessing the system via terminals, and by airlines and a range of other suppliers, including hotel, car rental and railroad companies. By 2000, the system supported more than 200,000 terminals, including PCs equipped with terminal emulation and graphical user interfaces (GUIs).

The system is illustrated in figure 6.

Figure 6

Overall GDS Structure



Key components may be summarized as follows:

- ***Data preparation***. A non-TPF, mainframe-based subsystem was built around the IMS hierarchical database. It handled loads of airline fare data from the Airline Tariff Publishing Company (ATPCO), Société Internationale de Télécommunications Aéronautiques (SITA) and other sources, and applied fare rules – which were relatively simple, and involved organizing fares by city pairs (origin and final destination). Loads typically occurred 7 to 12 times per day.
- ***Fare search/pricing and booking***. These were the primary TPF-based subsystems. Based on criteria specified by users, the fare search/pricing subsystem identified flights, generated itineraries (including multiple stages where appropriate) and calculated fares. Normally, three itineraries were generated, priced and presented to travel agents.

  The booking system processed reservations. Although functionally distinct from the fare search/pricing subsystem, there were high levels of interaction between the two as availability of fares was checked with the booking system database.
- ***Passenger name record (PNR) and flight operations***. These were smaller TPF-based systems that maintained and updated passenger and flight records respectively.

The system was designed for travel agents inputting relatively simple criteria, and for airline staff such as customer service and gate agents performing simple queries and transactions. With these user communities, performance and functionality was – and remains – satisfactory.

Web Shopping Impacts

In the mid-1990s, the company began to diversify aggressively into Web travel services. This had major GDS impacts in two areas:

1. ***Fare load times***. These became unacceptable. The existing data preparation subsystem – a nearly 30-year-old system that could not be easily modified – was creating delays of up to two hours before current data was available for searching. This was an obvious negative for a business operating in "Web time."
2. ***Fare search workloads***. The "look-to-book" ratio (i.e. the number of searches conducted before a reservation was made) increased radically as users shopped for lower fares or better connections. For competitive reasons, the company was obliged to increase the number of itinerary options presented from 3 to 9, and it was recognized that it would be necessary to increase this to 20 or more in the future.

To reduce delays in making current data available, the company began to bypass the legacy data organization system, loading fare data and rules directly into the TPF-based fare search/pricing subsystem. This shift exacerbated the impact of higher look-to-book ratios and larger numbers of itineraries, and dramatically increased the workload impact on the subsystem, which was now performing complex calculations for which it was neither designed nor optimized.

Average TPF instruction path lengths increased from 7 to 10 million to 20 to 30 million for domestic fare calculations while path lengths for more complicated international fares could be several times longer. The result was rapid escalation of the amount of mainframe processor capacity required. By 1998, capacity consumption measured in MIPS was growing at upward of 40 percent per year, and this continued during 2000 and 2001.

If this trend had continued, the company's processor capacity requirements would have grown over the next five years from approximately 8,500 used MIPS to more than 30,000, with major increases in hardware, software and other costs. The impact would in practice have been greater, as additional idle capacity was maintained to allow for failover. Although less directly affected, MIPS consumption for the booking subsystem also escalated as the number of checks for fare availability increased.

Management was concerned that the capacity limits of the existing base of TPF mainframe processors would soon be reached. The company employed relatively old IBM and compatible machines, and was reluctant to invest in newer technology.

There was also some dissatisfaction with application maintenance productivity (little new development had occurred for more than a decade). The company's TPF programming staff primarily employed manual assembler coding techniques. An attempt was made to introduce more effective C++ tools and practices, but was not pursued after initial results proved to be disappointing.

Replacement Project

General

In the late 1990s, the company began to look for alternative approaches to fare searching and pricing for Web shoppers. Management projected that the proportion of the company's business generated by conventional travel agents would decline from approximately 60 percent in 2000 to 35 to 40 percent by 2005. Most growth was expected to occur through the company's Web travel agency arm. A target of 20 percent annual growth for the company's core GDS business over this period was set.

Planning began with the assumption that it would not be possible to meet these demands with the existing TPF-based system, and that it would be necessary to develop a new application and host it on a different platform. The decision to employ HP NonStop servers was made at an early stage, and was based on criteria that are discussed in more detail below.

Development of the new fare search and pricing application was conducted by a team of approximately 60 programmers, including retrained TPF assembler programmers – the company experienced little difficulty in retraining these – along with developers with previous C++ experience. The application was developed primarily in C++ using the HP version of Microsoft Visual Studio.

The company had originally planned to use Java extensively in the new application. However, performance and reliability problems were experienced, and management later characterized use of this language as "unsuccessful."

Company plans called for a staged, multi-year phase-over from the existing TPF-based fare search/pricing subsystem to the new application on NonStop servers. It was expected that, in the first year (originally 2001, later 2002) approximately 20 percent of the fare search/pricing workload would be migrated to 12 NonStop servers with 16 CPUs each. By 2005, the installed base would increase to more than 2,000 CPUs, equivalent to at least 126 servers with 16 CPUs each.

It was expected that the entire GDS would eventually be migrated to NonStop servers, which would also be employed for other applications, including support for the company's Web travel agency business.

The company's decision to commit to NonStop had been made, and plans were already advanced, when Compaq announced plans to transition to the Intel Itanium CPU. There were concerns about need to recompile the new application. A plan for joint development by HP and the company of a hybrid-architecture platform including MIPS and Itanium CPUs was floated. It was hoped that this would avoid recompiling. But the concept does not appear to have been technically realistic.

Platform Selection

The decision to employ the HP NonStop platform was, according to the company, based on the following criteria:

- ***Availability***. Management recognized that UNIX or Windows servers could not provide 24x7 availability, and that the only realistic options were NonStop servers or zSeries mainframes.
- ***Performance and scalability***. To verify that the NonStop platform could meet the company's needs in these areas, a customized "stress test" was conducted. The test was regarded as particularly important because no existing NonStop users could be identified running this type of workload, with comparable volume, on NonStop SQL.

  The test involved running a limited function prototype of the new application on a configuration of four 16-way NonStop servers equipped with the NonStop SQL database, using a simulated air pricing and shopping workload. Performance and availability were found to be satisfactory. Test results were extrapolated to confirm that NonStop architecture would be capable of scaling efficiently to handle the company's long-term needs.
- ***Total cost of ownership (TCO)***. Initial calculations by the company indicated that savings relative to the existing TPF-based system could be realized through lower hardware and software costs; higher productivity (i.e. fewer personnel) for application development and maintenance; and fewer operations personnel.

Comparisons were based on existing TPF capacity utilization versus projected NonStop configurations; on relative costs of developing and maintaining the new application in C++ and assembler; and on existing TPF operations staffing compared to best practices norms for NonStop systems.

A consulting firm was retained to validate the company's TCO conclusions. The firm, which is known for its view (unique among industry analysts) that NonStop is the industry's lowest-cost server platform, confirmed that NonStop TCO would be lower than for TPF.

- ***Open system***. The company defined an open system as a "platform that permits integration and interoperation of many different applications using a standardized format." The NonStop environment was considered "open," while TPF was not.

  In practice, management was concerned to employ tools and practices that would enable applications to be delivered more rapidly than with manual assembler coding. With this objective, and with a view to application portability and interoperability, the company also specified certain industry standards. These included C++, POSIX, SQL, TCP/IP, Internet Mail (SMTP, POP, IMAP), Web serving, IIOP, XML, SOAP and MQSeries middleware.

- ***Zero latency enterprise***. Management was impressed by the capabilities of HP's ZLE scenario, and saw the ability to generate and exploit real-time information as potentially highly relevant to the company's future competitive application requirements.

The company also identified a number of areas in which it would like to see improvements in the NonStop environment. These included 64-bit addressing (HP does not expect full 64-bit capability to be implemented until well after the migration to Itanium has begun); faster processors; greater availability of third-party applications; improved development tools; and reduced footprint.

Efficient TPF Scenario

Enhancement Options

In developing its plans, the company did not evaluate TPF enhancement options. A number of opportunities were thus neglected.

First, redesigning the existing TPF subsystem could have reduced the costs of fare search and pricing workloads. The new application, designed to run on the NonStop platform, reduced the capacity required for fare calculations by more than 40 percent. Even larger capacity reductions could have been achieved through relatively simple optimization techniques for legacy TPF code. Such techniques were later implemented by the company.

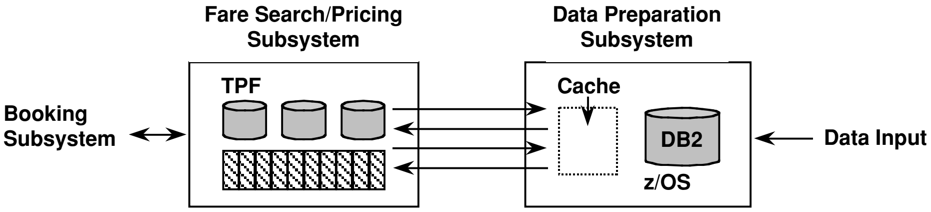
Early experience with the new system showed that response times for domestic fare searches averaged approximately eight seconds, about the same as for the existing system. In other words, the poorly optimized TPF fare search/pricing subsystem was delivering performance similar to that of the state-of-the-art NonStop system. A properly optimized TPF system would show significantly better performance.

Second, further capacity reductions could have been achieved, and load delays could have been eliminated by replacing the legacy data preparation subsystem. This approach had in fact been proposed at one point, but had been rejected by management. It would have reduced load times by at least the same amount – 75 percent – achieved by moving this function from TPF.

A new data preparation subsystem could have been implemented on a zSeries system or systems using DB2 under z/OS or Linux. For example, a TPF extension, TPF Application Requester (TPFAR), could have been used to fetch DB2 data stored in cache. This approach, illustrated in figure 7, would have enabled use of high-speed communications within or between zSeries system images, resulting in lower latency than would be generated by the NonStop scenario.

Figure 7

Alternative zSeries-based Data Preparation Approach



These approaches could, if an early start had been made to planning, have been implemented in a fraction of the time – probably less than two years, compared to three to four years – required to deploy the NonStop-based solution. It would also have been possible to enhance booking and other system components in a comparatively rapid and inexpensive manner.

Third, TPF application development and maintenance costs could have been reduced. “Time to market” for new GDS applications and functions could have been improved by employing C++ and visual tools comparable to those adopted for NonStop programming. There is a great deal of evidence from other TPF users that assembler programmers can – if the process is properly managed – be retrained, and practices changed to deliver significantly higher levels of productivity.

The company’s own experience confirms this. TPF programmers were in fact retrained to work on the new C++ application on NonStop servers, and the results were later characterized by the company as satisfactory. There is no reason that the same results could not have been achieved in a TPF environment, if a commensurate investment in skills and personnel development had been made.

The broader argument that TPF is not “open” is specious. The TPF environment supports all of the company’s specified standards, and meets the company’s own definition of “open systems.” It does not directly support a relational database, ORB or Java, but these are not functionally necessary for the application, and would in practice cause severe performance degradation. The company’s own experience with Java confirms this effect.

Fourth, savings in operating costs could have been realized within the TPF environment by replacing aging mainframe systems and upgrading operational support tools and practices. The cost structure for an efficient TPF environment with reduced capacity consumption and fewer processors would have been significantly different.

Long-term Goals

The company’s long-term goals for new applications and real-time data exploitation could be met in a simpler manner. The process would, again, have been more rapid and inexpensive than that actually pursued by the company.

The capabilities of the HP ZLE scenario are not unique to the NonStop platform or to NonStop SQL, and can be implemented using DB2, Oracle or other relational databases. Real-time data transfers between TPF and these could be implemented using TPFAR, which conforms to Distributed Relational Database Architecture (DRDA) standard for SQL data interoperability.

An advantage of this approach would be that, unlike NonStop SQL, DB2 and Oracle are de facto standards that enjoy industry-wide support from independent software vendors (ISVs) and third-party services firms. Skills are also readily available for these.

Comparative Costs

Hardware and Software

The hardware and software costs of an efficient, properly optimized TPF scenario for fare search and pricing subsystem are calculated as follows.

First, TPF capacity consumption is reduced by 70 percent, due to the effects of code optimization and a redesigned data preparation subsystem. The efficient TPF scenario assumes 20 percent capacity growth per year over a five-year period, while the inefficient TPF scenario assumes that used capacity increases at the historical rate of 40 percent per year.

As shown in figure 8, required capacity in MIPS for the efficient TPF scenario is significantly lower than would otherwise have been the case.

Figure 8

MIPS Consumption for TPF Scenarios

TPF Scenario	Year 1	Year 2	Year 3	Year 4	Year 5
Inefficient	8,500	11,900	16,600	23,324	32,654
Efficient	2,550	3,060	3,672	4,406	5,288

TPF capacity requirements are translated into processor acquisitions based on MIPS consumption rounded to the next-largest z900 model as shown in figure 9. It is assumed that all first year capacity is purchased new. The new data preparation subsystem employs the IBM z/OS operating system and DB2 database.

Figure 9

Number of Systems for NonStop and Efficient TPF Scenarios

Scenario	Year 1	Year 2	Year 3	Year 4	Year 5	End of Period
NonStop	60 x 16	+12 x 16	+15 x 16	+18 x 16	+21 x 16	126 x 16 (2,016 CPUs)
Efficient TPF	2 x 1C6 z103*	+z103	+z103	+z104	+z1C4	7 systems (6,221 MIPS)

*New data preparation subsystem

For the NonStop scenario, the first-year configuration is based on the company's estimate that it would take 12 NonStop servers with 16 CPUs to handle 20 percent of the TPF fare search/pricing workload. This is prorated to 100 percent of the workload, for 60 servers with 16 CPUs each.

It is assumed, again, that all first-year systems are purchased new, and that 20 percent capacity growth occurs each year over a five-year period. Configurations are rounded to the next largest increment of 16 CPUs.

Hardware acquisition costs for the efficient TPF scenario are based on a "street price" of $2,270 per MIPS for loaded zSeries configurations, while maintenance costs are based on four percent of cumulative hardware acquisition value per year. Software costs are based on $1,000 per millions of service units (MSU) per month for TPF processors and $112 per MSU per month for z/OS, DB2 and CICS software for the new data preparation subsystem.

For the NonStop scenario, acquisition costs are based on HP published list prices, discounted by 50 percent, for S74000 processors and NSK and NonStop SQL software. Hardware maintenance and software support costs are based on 4.8 and 5 percent of cumulative acquisition value per year respectively.

A number of comments should be made about these calculations. First, they are for fully operational first-year configurations, adjusted for annual workload growth. In practice, NonStop systems would be phased in over a multi-year period, and the existing TPF system would continue to function until NonStop deployment was completed. The actual cost structure would thus be more complex.

Second, the efficient TPF scenario assumes that new zSeries processors are purchased for the first-year configuration, and in subsequent years. Hardware costs are thus significantly higher than if the company had continued to use existing capacity. In practice, requirements could have been met over the five-year period without purchasing new processors.

Third, calculations for both scenarios are based on current technology platforms as of April 2002. zSeries configurations and costs are based on z900 processors introduced by IBM in October 2000, rather than the more powerful models introduced in May 2002. NonStop configurations and costs are, similarly, for MIPS R12000-based S74000 rather than newer R14000-based S76000 models.

In practice, users would tend to purchase latest-generation technology models of both platforms over the five-year period. There is, however, no obvious reason to expect that price/performance evolution would, over the period as a whole, be significantly faster for one platform relative to the other.

Personnel Costs

Personnel costs for the two scenarios are based on the full time equivalent (FTE) staffing levels shown in figures 10 and 11.

Figure 10

Application Development and Maintenance Personnel

Scenario	Development	Maintenance
NonStop	Fare pricing systems 100 FTE years Visual Studio/C++	Fare pricing systems 12 FTEs/year Visual Studio/C++
Efficient TPF	Code optimization 25 FTE years VisualAge/TPF Data preparation 30 FTE years C++	Core system Assembler: 5 FTEs/year VisualAge/TPF: 15 FTEs/year Data preparation 2 FTEs/year C++

Figure 11

Administration, Operations and Support Personnel

Scenario	System & Data Administration	Operations & Technical Support
NonStop	30 FTEs	23 FTEs
Efficient TPF	20 FTEs	15 FTEs

It is assumed that development for both scenarios is conducted in C++ using HP and IBM visual tools respectively. For the efficient TPF scenario, it is assumed that code optimization is performed using the IBM VisualAge/TPF tool set, and that maintenance is performed using a combination of C++ and assembler.

Costs are based on average annual salaries of $66,000 per year for all application development and maintenance personnel; $60,000 and $40,000 per year for NonStop systems and data administration, and operations and technical support personnel respectively; and $54,000 and $40,000 for the same categories for zSeries systems. Salaries are increased by 30 percent per year to allow for benefits, bonuses, training, travel and related items.

A detailed breakdown of overall costs for the two scenarios is presented in figure 12.

Figure 12

Detailed Costs Breakdown: NonStop and Efficient TPF Scenarios

SCENARIO	Year 0	Year 1	Year 2	Year 3	Year 4	Year 5	Total
NONSTOP							
Hardware acquisition	–	37,620	7,524	9,405	11,286	13,167	**79,002**
Hardware maintenance	–	1,806	2,167	2,618	3,160	3,792	**13,543**
Software acquisition	–	12,390	2,478	3,098	3,717	4,337	**26,020**
Software support	–	620	743	898	1,084	1,301	**4,646**
Subtotal ($ thousands)	–	**52,436**	**12,912**	**16,019**	**19,247**	**22,597**	**123,211**
Application development	8,580	–	–	–	–	–	**8,580**
Application maintenance	–	1,030	1,030	1,030	1,030	1,030	**5,150**
Admin/operations/support	–	3,536	3,536	3,536	3,536	3,536	**17,680**
TOTAL ($ thousands)	**8,580**	**57,002**	**17,478**	**20,585**	**23,813**	**27,163**	**154,621**
EFFICIENT TPF							
Hardware acquisition	–	7,258	1,469	1,469	1,895	2,032	**14,123**
Hardware maintenance	–	301	349	408	484	565	**2,107**
Software	–	5,407	6,751	8,095	9,811	11,647	**41,711**
Subtotal ($ thousands)	–	**12,966**	**8,569**	**9,972**	**12,190**	**14,244**	**57,941**
Application development	4,719	–	–	–	–	–	**4,719**
Application maintenance	–	1,888	1,888	1,888	1,888	1,888	**9,440**
Admin/operations/support	–	2,184	2,184	2,184	2,184	2,184	**10,920**
TOTAL ($ thousands)	**4,719**	**17,038**	**12,641**	**14,044**	**16,262**	**18,316**	**83,020**

ABOUT THE INTERNATIONAL TECHNOLOGY GROUP

ITG sharpens your awareness of what's happening and your competitive edge
. . . this could affect your future growth and profit prospects

The International Technology Group (ITG), established in 1983, is an independent information services and management consulting firm specializing in timely analyses of current developments and future trends in information systems technologies and markets.

Particular attention is given to analyzing trends in emerging technologies and markets. ITG's staff and supporting consortium of international analysts are well experienced and skilled in creative assessment techniques and research methodologies.

Services provided to clients are designed to provide factual data and reliable documentation to assist in the decision-making process. The information provided establishes the basis for developing tactical and strategic plans. Important developments are analyzed and practical guidance is offered on the most profitable ways to respond to the changes influencing and shaping complex IT deployment.

A broad range of services is offered, furnishing clients with the information necessary to complement their internal capabilities and resources. Customized programs are designed to provide a continuous source of technology, product and market information involving delivery of various combinations of the following services:

Status Reports	In-depth studies analyzing important topics and future industry trends
Flash Reports	Timely documents covering latest developments and significant market activity
Management Briefings	Periodic interactive planning meetings with management to address tactical issues
Telephone Consultation	Immediate response to clarify conclusions and review current informational needs
Executive Presentations	Scheduled interactive strategic meetings with decision-makers to examine future directions

The firm's **Enterprise Information Systems** service is particularly noteworthy. This service covers various architectures, technologies and techniques that may be effectively deployed in order to achieve the goals of enterprise-wide information systems. Scenarios being proposed by leading system suppliers and software vendors are objectively evaluated. Clients include systems suppliers, software vendors, service firms, financial institutions, government agencies, trade associations and end users.

International Technology Group

4546 El Camino Real, Suite 230
Los Altos, California 94022-1069
Telephone: (650) 949-8410
Facsimile: (650) 949-8415
Email: info-itg@pacbell.net