

Bulletin

Case Study: Amdocs Implements IMDB Technology from TimesTen in New Generation of Applications

Analysts: Rob Hailstone and Levent Toros

IDC Opinion

What role can in-memory database (IMDB) technology play in commercial, performance-critical application products and what has been the experience of Amdocs in deploying the TimesTen product?

IMDB technology still occupies a niche data management role but offers considerable performance advantages over conventional disk-centric RDBMS products. Amdocs has retained its strategy of deploying mainstream RDBMS in support of its application products but has selectively deployed TimesTen IMDB technology to give it a competitive advantage in a performance-critical function. It has delivered real performance benefits that will help its customers control costs, and it has achieved this without compromising the stability of the applications.

Introduction

Amdocs is a leading provider of billing and customer relationship management (CRM) to the communications industry. In revising its products to suit the evolving industry requirements for profitable growth, improved customer loyalty and operational efficiency leading to reduced cost of ownership, Amdocs has addressed the need for increased performance and scalability while stabilizing costs. One of the factors in achieving this has been the implementation of TimesTen's in-memory database (IMDB) technology, which is at the core of the TimesTen Real-Time Event Processing System.

This paper describes the business issues being addressed, the technology deployed, the experiences of Amdocs, and the benefits achieved through the use of IMDB technology.

An Overview of the IMDB Market

The reason the database market evolved was to support the need for concurrent shared access to information from a variety of applications and users, and to support the logic required to retrieve and update selected pieces of information. Thus the DBMS evolved as a server to client applications, exposing itself through an application program interface (API). The mainstream DBMS market is represented by relational database management systems (RDBMS), which accommodate the SQL standards and support specific APIs such as ODBC, JDBC, and pre-compiled code such as Oracle's Pro*C

The primary performance constraint on RDBMS products has always been the need to find the appropriate entries on disk, and move them into memory from where they can be returned to the requesting application. Three primary mechanisms are used to improve the performance of this basic requirement:

- Indexing — several different techniques have been developed for maintaining index tables that prevent the need for the DBMS to search a large number of entries before finding the relevant data. There is a trade-off between the retrieval benefits of using multiple indices and the overhead of updating these when the underlying data changes.
- Caching/buffer management — because data usage normally falls into a pattern (as opposed to being entirely random), RDBMS products have evolved elaborate cache management capabilities that seek to keep the most frequently used data in memory buffers and avoid disk access altogether.

Quoting IDC Information and Data: *Internal Documents and Presentations*—Quoting individual sentences and paragraphs for use in your company's internal communications does not require permission from IDC. The use of large portions or the reproduction of any IDC document in its entirety does require prior written approval and may involve some financial consideration. *External Publication*—Any IDC information that is to be used in advertising, press releases, or promotional materials requires prior written approval from the appropriate IDC Vice President or Country Manager. A draft of the proposed document should accompany any such request. IDC reserves the right to deny approval of external usage for any reason.

Copyright 2002 IDC. **Reproduction is forbidden unless authorized.**
For additional copies please contact Meryl Salter +44 (0) 20 8987 7100.

Check us out on the World Wide Web!

<http://www.idc.com>

- Optimization — the process of matching the data request to the best possible means of retrieving the information (which may be fragmented across a number of data entries). This takes into account the availability of data already in the memory cache and the structure of indices that can be used to find the data.

All other performance enhancement techniques such as parallelization are built on top of these fundamental performance mechanisms.

As server hardware and operating system technology have matured to provide access to very large amounts of cost-effective main memory, DBMS technology has evolved to exploit ever-larger data caches. However, ultimate performance remains dependent on the architecture of DBMS products that are still primarily mechanisms for delivering disk-resident information to application programs.

Very high performance requirements can be met by building a DBMS architecture that does not depend on disk processing at all, except for recovery. This is quite different to implementing a conventional RDBMS and ensuring that all of the data will fit within the memory cache. If designed from the outset for in-memory processing only, a system built around IMDB technology can be relieved of a very large proportion of the code execution that burdens the mainstream products. In particular:

- Indexing — while still required to prevent serial scans of memory, it can be much simpler and more compact. For example, a conventional index stores a data value followed by the address at which the record can be located. When all the data is in memory anyway, there is no need to store the data value redundantly — only the address is required. While this sounds counter-intuitive, it represents a significant saving in processing and storage.
- Caching/buffer management — no longer exists as such since all of the data is in memory already. All of the processing as well as all of the disk latency associated with moving data from disk to memory (and vice versa) is removed.
- Optimization — is still required but becomes much simpler and is more likely to choose the fastest path, since the number of alternatives is much less than with a disk-based RDBMS.

Other operations are similarly simplified (and simplicity can be transformed into performance and reliability). For example, performance does not suffer to the same extent if a large number of insertions, updates, and deletions cause the storage to become fragmented. There are few reasons to need to defragment the storage of an IMDB, and such an operation is, in any event, much quicker than that associated with defragmenting disk storage. The associated management tasks become simpler, and the whole product assumes a lighter administration profile with greater overall availability.

The usual objection to an IMDB is associated with recovery from failures. There is really little foundation to this. An IMDB product can log changes to a physical medium, and as long as the update to retrieve ratio is relatively light this will not impose a significant performance overhead. For example, TimesTen includes various options for logging changes to disk, and for replicating changes across multiple servers for failover protection and load balancing. Alternatively, the IMDB can be implemented for applications where rollback operations are not required and the database can be rebuilt in the event of a failure. An example of this would be its use within a cellular mobile switching center, where in the event of a failure, the database tracking the locations of mobile devices being serviced can be rebuilt piecemeal the next time each device communicates its presence with a base station.

The other objection, of course, is that the database size is inherently limited by the memory available at the server, and by the cost of the main memory in contrast with the much smaller cost of disk storage. The onset of 64-bit computing has relieved most theoretical limits on database sizes, but in practical terms this remains a barrier that is being progressively raised — but at a slower pace than the demand for online data. Hence a stand-alone IMDB has remained a niche solution within the broader enterprise market, and is more often used in conjunction with a back-office, disk-based RDBMS.

The fact that the capacity requirements of the data management market have continued to outstrip the storage capabilities available to IMDB products has caused the initial strong interest in the market to stagnate except in specific areas. What was, three or four years ago, a dynamic market with a number of innovative vendors has now lost most of these vendors through a process of company realignments, acquisitions, etc. The technology has generally not been abandoned but has gone undercover, being embedded as a native component of niche applications that have a very high performance requirement.

TimesTen, Inc.

<http://www.timesten.com/>

The most substantial of the remaining commercial vendors is TimesTen. TimesTen's Real-Time Event Processing System features an in-memory data manager with relational data architecture, which utilizes SQL as its API. It is therefore categorized by IDC alongside Oracle9i, DB2, and SQL Server. On this basis only, TimesTen falls a long way behind these market-leading products with a market share in Western Europe of less than 0.1%. However, this categorization based on functionality is not truly representative of the way in which the technologies are used. In many circumstances the IMDB approach is complementary to disk-based RDBMS products, as shown by this case study. In fact, TimesTen plays a central role in performance-critical applications deployed by very major organizations — in particular, within the telecommunications and financial services industries.

Because of the comfort factor of the traditional RDBMS, IMDB technology tends to be used as a last resort when an RDBMS has failed to deliver the required performance in a cost-effective manner.

IMDB technology is no different to other technologies in that in-house constructed solutions generally give way to commercial products over time.

Because of the comfort factor of the traditional RDBMS, IMDB technology tends to be used as a last resort when an RDBMS has failed to deliver the required performance in a cost-effective manner. In the industries where TimesTen is focused this is not uncommon, and prior to the availability of any commercial IMDB products, several of these organizations have had to resort to in-house built caching infrastructures.

IMDB technology is no different to other technologies in that in-house constructed solutions generally give way to commercial products over time. The build versus buy equation is stacked towards commercial products because of the combination of reduced support costs, adoption of standards, and ongoing performance and functionality enhancements. This is certainly true of data management in general and applies increasingly to the IMDB market.

TimesTen addresses a number of specific market areas:

- Embedded OEM technology — in which the IMDB is delivered as an integral part of the application, often to service real-time data requirements for equipment where little human intervention is required.
- Front-end cache — to a conventional RDBMS. TimesTen has created middleware to manage replication between Oracle databases and the TimesTen database. The TimesTen database can be located remotely from the main Oracle database and provide very high speed response to user requests — without compromising the data integrity provided by the conventional Oracle logging and recovery mechanism. This middleware was delivered initially as a separate product called Front-Tier, but the functionality has now been packaged in with the latest release of the TimesTen product.
- Standalone IMDB — implemented as a full RDBMS in support of performance-critical applications where the data volumes fall within the scope of the memory limitations.

The replication middleware developed for use with a front-end cache architecture also delivers the ability to replicate from one TimesTen database to another resident on a separate server for hot-standby purposes. This low-latency replication makes TimesTen suitable for use in applications where there is a high availability requirement.

Background to Amdocs

<http://www.amdocs.com/>

Amdocs is an international provider of billing and CRM to the communications industry. Amdocs is a leading solution provider in this field with a 20-year track record and worldwide coverage.

Amdocs' applications cover the broad scope of telecommunications front- and back-office requirements. This includes pre-integrated products, customized solutions, and outsourcing services, aiming to deliver time-to-market advantages and support for next-generation services to its customers.

In 2001 (calendar), IDC rated Amdocs as the market share leader in the telecommunications billing software market with a market share of 17.3%, but with significant competition from Convergys/Geneva. In the equivalent service bureau market, IDC placed Amdocs in fifth position with an 8.4% market share but with higher growth than the lead players. Amdocs has a very high penetration into tier one telecom companies around the world. In Western Europe, the customer list includes Belgacom, BT, Deutsche Telekom, One2One, Orange, Telia, and Vodafone.

The telecom market within which Amdocs competes is undergoing rapid change and presents a number of specific challenges:

- The need for a reduced total cost of ownership (TCO)
- The need for more adaptable billing models
- The need to enhance customer loyalty, especially for prepaid customers
- Heightened sensitivity to customer service levels
- The need to position for 3G services.

The sum of these factors is a position that is well recognized throughout IT — the need to deliver higher performance, richer applications, greater service levels, and high adaptability — all at a reduced cost.

The sum of these factors is a position that is well recognized throughout IT — the need to deliver higher performance, richer applications, greater service levels, and high adaptability — all at a reduced cost.

It is against this business environment that Amdocs has built a new generation of its application portfolio and introduced the use of TimesTen IMDB technology.

A more complete description of Amdocs, its market position, competition and the market dynamics can be found in *Worldwide Telecom Billing Forecast and Analysis, 2002-2006* (IDC #27003, April 2002). The European market for telecom applications is described in *Telecommunication: Back to Basics* (IDC #LS26J, May 2002).

New Generation Telecommunications Applications

The primary Amdocs products impacted by the subject of this paper are Amdocs Enabler and Amdocs Mobile.

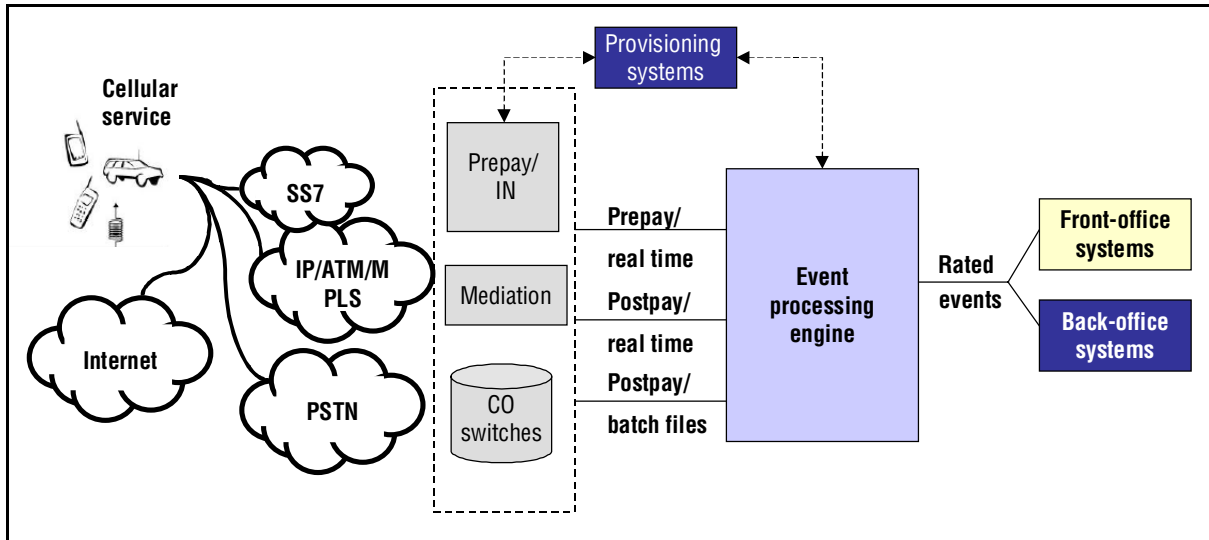
Amdocs Enabler provides the core functionality for rating and billing, supporting online and offline charging, and full convergent support for all payment methods — including “nowpaid” for real-time commerce transactions as well as prepaid and postpaid. Amdocs Enabler also provides product catalog and acquisition and formatting support.

Amdocs Mobile is integrated with Amdocs Enabler and provides additional functionality for customer management (including customers with hybrid accounts having multiple payment methods), as well as resource management, provisioning, and accounts receivable and collection.

Further information on these products is available at: http://www.amdocs.com/product_tree.asp?ID=98.

The overall breadth of the enhancements made to Amdocs' new generation of application products is outside of the scope of this paper. Although TimesTen is deployed in a number of application components in the Amdocs suite, the functionality that concerns us here is the event processing component (the rating engine) that forms a critical part of its real-time, convergent prepaid and postpaid billing solutions. Figure 1 shows the overall flow of information through this process:

Figure 1
Event Processing in Amdocs Applications



Source: IDC, 2002

Network usage events (e.g. IP usage) and circuit-switched based voice connections are initiated by any of a growing number of subscriber activities. Initially, the majority of these will be simple voice calls and SMS requests, but the system is designed to cater for an increasing number of data calls plus the use of online games and video-on-demand. Ultimately, each request will be routed to the Amdocs application environment via network elements and mediation systems. Each of these service requests must be processed by the event processing engine in order to determine the availability of funds, the amount to be charged, and which account to charge.

For conventional postpaid accounts, the peaks and troughs of usage are smoothed by accumulating requests into a number of batch files. These usage records (e.g., CDRs) are then pulled from the switches in batches to produce bills (typically several times per hour, but real-time CDRs may flow into the system depending on the mode of operation). However, requests made by users of prepaid devices must be processed and provisioned in real time in order to establish that sufficient funds are available before the call is allowed to progress.

In addition to prepaid billing models, postpay rating systems are increasingly requiring real-time processing features (in turn demanding a high performance, high availability application platform) because:

- IP network elements (e.g., routers) are not designed to store billing information and hence, if the service provider would like to bill the service on a per-usage basis, it needs to rate and bill the subscriber in real-time.
- Incumbent service providers are expanding their efforts to offer Web-based bill payment and presentment options to their customers. Real-time information presentment about the network service usage is becoming a key offering.
- Other critical and growing OSS and front-office systems ,including business intelligence, churn management, and fraud management, are demanding real-time information flow even if the service provider billing structure is mainly postpaid.

Amdocs is one of the few telecom billing vendors that offer a single integrated solution for prepay and postpay billing models, and this is likely to become an increasingly valuable competitive differentiator as providers increase the range of services on offer.

The output from the event processing engine consists of rated events that are used for back-office functions such as billing, and front-office functions such as CRM.

Depending on the nature of the event and on the specific account details, the logic performed by the event processing engine may be straightforward or very complex. The steps required are to:

- Format the incoming events from their initial structure to a standard layout
- Identify the event initiator — the actual identifier varying by the type of event being processed
- Identify who will pay for the service. This may be the initiator or could be a corporate account holder or other body
- Establish the most appropriate scheme. This will vary according to individual accounts, and must consider the number of free minutes available, call rates depending on time of day, special schemes such as “friends and family”, and whether the call is for an overseas destination, etc. Current usage will also need to be considered in order to establish the actual rate to be used where a tiered rate scheme is in operation
- For prepaid calls, establish the amount of funds available and match this to the anticipated call cost
- Create rated event records as an output stream for onwards processing

The adaptability of the event processing engine derives from being driven by rules tables and having access to subscriber-specific data. It follows that this represents a critical performance bottleneck in the ability to manage the vast number of subscriber activities. Amdocs has avoided this becoming a limit on scalability by designing the applications to use parallel event processing engines, so that scaling can be achieved by adding additional servers.

The ability to scale by creating a farm of event processing engines comes at a cost — both in terms of the cost of acquiring the server capacity and the ongoing cost of administration. These costs are contrary to the desire to deliver more for less described earlier.

In other words, by adding CPUs to the system the service provider can increase the number of events billed per second almost proportionally. However, the ability to scale by creating a farm of event processing engines comes at a cost — both in terms of the cost of acquiring the server capacity and the ongoing cost of administration. These costs are contrary to the desire to deliver more for less described earlier. In order to minimize the cost of scalability, the event processing engine must be made as efficient as possible.

Use of TimesTen's IMDB Technology

The data requirements of the Amdocs applications have been built around the use of Oracle. The overall data requirements call for large databases as well as high volumes, and a mainstream RDBMS product such as Oracle is the natural way to address this requirement. Oracle has been shown to provide a good solution for these requirements and will remain the strategic DBMS for the Amdocs applications.

However, the event processing engine — as described previously — is very data-access intensive and represents a potential bottleneck and a significant potential cost. In the latest version of the application suite, Amdocs has therefore taken the step of replacing Oracle with TimesTen's IMDB technology for data management within the event processing engine, while retaining Oracle for other data management activities (such as the storage of large amounts of historical rated usage information).

The use of TimesTen in this functional component has had the impact of:

- Eliminating the latency due to disk access
- Reducing the time required to process each event
- Increasing the amount of work that can be accomplished by a single occurrence of the event processing engine
- Reducing the overall platform cost to support a given workload
- Reducing the administration overhead in maintaining the system for optimum throughput

The replication capability of TimesTen means that the requirements of a high-availability environment can be preserved. The use of TimesTen replication within Amdocs Enabler allows it to support full 24/7 operations, meeting the communication industry's need for the highest standards of availability in support of prepaid applications.

Impact of Implementing TimesTen

IDC has not been able to carry out a formal return on investment (ROI) study due to the early stage of deployment and the competitive sensitivity of this data. This section, therefore, has to be limited to a short description of the areas in which Amdocs report benefits. The associated costs (or lack of costs) are described in the next section.

Benefits fall into two primary groupings — those relating to reduced costs for the service provider and those relating to improved perception by the subscriber base.

Reduced Costs

Direct cost reduction is reported due to the ability to manage a greater workload with a less expensive platform. TimesTen claim data management performance advantages of approximately tenfold over a conventional RDBMS — even with all of the data cached in memory. Actual performance improvements will vary considerably depending on the profile of the data access for particular applications.

Amdocs reports an approximate overall gain of five-fold — enabling a single engine to absorb approximately five times the work load of the equivalent Oracle-based version before it becomes necessary to add a further engine for ongoing scalability.

In the case of Amdocs, the same platform supports both the data management and application logic for the event process engine so that the apparent performance advantage is diluted by the need to support processing logic as well as data access. Nevertheless, Amdocs reports an approximate overall gain of five-fold — enabling a single engine to absorb approximately five times the work load of the equivalent Oracle-based version before it becomes necessary to add a further engine for ongoing scalability. As described earlier, because this engine can be scaled by adding further platforms there is no ultimate ceiling to scalability either in this version or the earlier disk-based product.

Indirect cost reduction comes from the lower level of administration required to keep the TimesTen technology performing optimally.

Indirect cost reduction comes from the lower level of administration required to keep the TimesTen technology performing optimally. As with any disk-based RDBMS, the performance of Oracle is influenced by the skill of the database administrator (DBA) in managing the effective use of the physical storage. While DBA tools have become much more effective in recent releases, IMDB products avoid the problem entirely resulting in a low management-overhead environment.

Improved Subscriber Perception

Once again, due to the early stage of deployment, IDC has not been able to verify these claims with user interviews. However, subscribers are expected to see a number of direct benefits:

- Prepaid-postpaid convergence. By facilitating prepaid and postpaid event processing on a unified platform, Amdocs Enabler allows the operator to offer the same services and price plans for prepaid, postpaid, and hybrid solutions. This architecture also facilitates bundled prepaid-postpaid price plans and cross-channel discounts and benefits.

- Faster prepaid call connection. Event processing must be completed in order to verify sufficient funds exist before connecting a prepaid call. The event processing engine could represent a distinct bottleneck in this process. By improving the performance of this engine, Amdocs should be able to process peak loads without noticeable delay and without incurring substantial infrastructure costs.
- Current account status information. Account information is made available to subscribers through front-office systems that are fed with call information by the event processing engine. By improving the throughput and reducing the latency across this engine, subscribers will be presented with the latest account information and be able to make appropriate business decisions with greater confidence.

Architecting Applications for Adaptability

The costs of implementing a new approach such as this fall into four major categories — direct costs (e.g. product license costs), ongoing administration costs, application conversion costs, and costs associated with the risks involved. Of these, the direct costs are measurable, and compared to the expected benefits, are not large. Ongoing administration costs have in fact been reduced by the conversion of the applications to use the TimesTen technology, since the inherent simplicity of IMDB products means that the data management features can be embedded into the applications in a completely transparent manner. There are no new skills needed by those deploying Amdocs applications in order to manage the TimesTen environment, and the Oracle administration load has been reduced.

The costs associated with application conversion and with associated risks can be mitigated very considerably by using an appropriate application architecture.

Amdocs applications use a layered architecture whereby the logical application layer is independent from the data access logic, and the data access logic is itself independent from the underlying DBMS.

Amdocs applications use a layered architecture whereby the logical application layer is independent from the data access logic, and the data access logic is itself independent from the underlying DBMS. This architecture had been deliberately chosen for the adaptability offered and the freedom from lock-in to any particular technology, and it is this that has eliminated much of the potential risk. Little use is made by the applications of proprietary DBMS features such as stored procedures, so the freedom of choice has not been compromised significantly.

The stages in converting the event process engine to use TimesTen were to:

- Analyze the areas of applications that perform data access
- Ensure these areas of code are properly wrapped and thus isolated from the application logic
- Switch the data access to use TimesTen.

Since the applications use standard ODBC access to the database, no other changes are required. In particular, the application logic remains unchanged so that there is no risk to the stability or behavior of the code.

This architecture also addresses the different issue of potential risk due to the relatively small company size of TimesTen. While IDC does not anticipate any issues with respect to the ongoing evolution or support of the TimesTen product, it has to be acknowledged that the company is not in the same league as the major RDBMS vendors. Amdocs has shown that the replacement of the underlying DBMS for a critical part of the business functions need not be a traumatic exercise, and there is no reason why this replacement should not be bi-directional if future requirements dictate a different approach. Just as Amdocs avoided creating a lock-in to Oracle by adopting this layered approach, this same approach would enable the redeployment of Oracle if this were deemed to have business benefits in the future. No lock-in has been created to TimesTen.

It should be emphasized that Oracle is still the preferred deployment DBMS for the Amdocs applications and is still regarded as the best fit with the overall data management requirements. The exception to this is the event processing engine described in this paper. However, the application architecture provides the potential for greater deployment of IMDB technology in the future should this be seen to be to the advantage of Amdocs' customers.

In this particular set of applications, the choice of a "best fit" DBMS is not a simple one, with no single product providing all the features required across the suite. The architecture chosen by Amdocs provides the ability to use different products to deliver the best fit with the data needs of individual components.

Conclusions

As the market for 3G telecom services steps up, service providers will be faced with a growing number of subscribers who use multiple services with mixed prepay and postpay agreements. The ability to create a common billing process will become increasingly important as a service differentiator. The ability to manage an increasing need for real-time rating plus batch processing of large, time-sensitive workloads will continue to drive the search for higher performance at lower cost.

The use of IMDB technology within the event processing engine has been shown by Amdocs to deliver this improved cost/performance, and it has been incorporated into the application products in a manner that does not impose any additional risk to stability or adaptability. Because the technology has been encapsulated in a manner that is not visible to users or administrators, no additional skills are needed and overall administration costs are also reduced.

This combination of performance improvement and cost reduction is exactly the right match with the requirements of the telecommunications providers, and should help Amdocs to retain its position as a leading provider of applications to this market.

This combination of performance improvement and cost reduction is exactly the right match with the requirements of the telecommunications providers, and should help Amdocs to retain its position as a leading provider of applications to this market.

The combination of IMDB technology and conventional disk-based RDBMS is unusual but effective, and is a good example of the use of the most appropriate technology to address the

problem at hand. IDC believes that there must be many other possibilities for using IMDB technology to address specific performance-critical, data-intensive processing without needing to re-evaluate the strategic DBMS decision.

The IT market has regarded RDBMS products as commodity items for several years (with the obvious exception of high-end usage such as in a large data warehouse). However, there are relatively few occurrences of applications being moved to an alternative technology. In most cases the potential benefits of such a move are less than the cost in terms of application modification and risk to stability. Amdocs has shown that such a move can be justified if the business benefits are sufficiently strong and the application architecture has been designed with technology independence in mind.

Document #: AD29J

Publication Date: October 2002

Published Under Services: European Application Deployment and Integration Platforms; European Infrastructure Management Services; European Utilities and Telecom Applications; Next-Generation OSS and Billing; European Industry-Specific Applications
