IDC
Analyze the Future

Global Headquarters: 5 Speen Street Framingham, MA 01701 USA P.508.872.8200 F.508.935.4015 www.idc.com

WHITE PAPER

IBM's Secret Weapon: The Power, Flexibility, and Self-Managing Capability of Informix Dynamic Server

Sponsored by: IBM

Carl W. Olofson
February 2008

IDC OPINION

Informix Dynamic Server (IDS) represents a technology that's too significant to compromise or to try to blend into a more general product line. That's why IBM decided to boost investment in IDS development and promote it as a premier relational database management system (RDBMS) for high-volume transaction applications. IDS neither competes with nor represents an alternative to DB2. Rather, IDS represents an embeddable transaction-oriented RDBMS featuring valuable capabilities that distinguish it from DB2 in a significant way. Release 11 also includes features that help support IBM's initiative regarding Web 2.0 in general and its Information On Demand (IOD) effort in particular. In short, IBM has found IDS 11 to be:

- Ideally suited for demanding online transaction processing (OLTP) workloads
- Highly suitable for embedding in transaction-oriented ISV applications
- Capable of delivering both excellent performance and reliability and a high degree of deployment flexibility

IN THIS WHITE PAPER

This white paper briefly recaps the evolution of IDS as a leading RDBMS technology, how it became part of IBM's RDBMS product portfolio, and how its capabilities are leading to a renaissance of the product in the tough and competitive RDBMS market. It examines the special database technology requirements of high-volume transaction applications and shows how IDS addresses those requirements. The document discusses the capabilities that set IDS apart from the field in general and distinguish it from other IBM RDBMS offerings (including DB2). It also shows how IBM has recommitted itself to IDS in terms of technology investment as well as sales and marketing effort, and how customers are deriving benefits from the product.

SITUATION OVERVIEW

The Background of IDS

In 2001, IBM acquired IDS along with the other DBMS products of the former Informix Corporation. Most people familiar with the DBMS industry regard Informix as a failed company, and so they are tempted to come to the unwarranted conclusion that IDS is a failed DBMS. Nothing could be further from the truth. In fact, Informix's problems

were entirely due to business decisions and management actions, not any technical failure on the part of its products. When IBM acquired those products, the company was tempted to put them on a shelf, or migrate their users to DB2. IBM has found, however, that there is merit, from the perspective of broad market addressability and from the perspective of customer retention and business growth, in offering other RDBMS products to serve a variety of purposes. One such product that has been quite successful despite its lack of marketing promotion is IDS.

Since 1982, Informix had developed the RDBMS technology of IDS, building it as an efficient, transaction-oriented full relational DBMS that managed pools of session resources to deliver more efficient support of large numbers of concurrent sessions, used internal lightweight threads, and optimized SQL application plans for parallelization to deliver blistering transaction throughput. The company created a shared-nothing clustered capability and later acquired Illustra (founded by RDBMS pioneer Michael Stonebraker) to include object-relational support. Despite business missteps, including a certain lack of product focus on the part of Informix, IDS, with its reliability, excellent performance, and very low administration requirement, acquired a rapidly growing army of very satisfied, even fanatical users. Since IBM's acquisition of the product line, and despite a lack of external promotion, the IDS customer base has been loyal and devoted.

Although at one time IBM considered a plan to gradually merge the IDS and DB2 technologies into a single product line, the company has come to realize that this move not only is not feasible but also could compromise a powerful and successful product, IDS. Instead, IBM has found that IDS is especially good at meeting the needs of users who have high-volume transaction throughput requirements and who are resource constrained in terms of technical staffing. IDS is also an ideal candidate for embedding in data-intensive applications that typically feature high transaction volume and are sold to users who have little or no DBA staff.

In addition, in a gesture of ongoing commitment to IDS users, IBM is providing major support for the International Informix Users Group (IIUG) revival of the annual global meeting of Informix users as a separate and independently managed event.

Technology Challenges/Requirements and Criteria for High-Volume OLTP and Low-Volume Staff

As was previously mentioned, IDS is part of a portfolio of DBMS products offered by IBM. Why does this make sense? Because the various DBMSs tend to excel at solving different kinds of problems, and because no one DBMS is appropriate for each and every data management workload out there. Varied workload elements include complex query support; small memory footprint; support for complex data structures; object, document, and extended data type support; multivalued field support; and so on. IBM and its competitors offer products that address different combinations of these workload elements. Some are good for data warehouses, some for reporting databases, some for embedding in small devices, and some for rapid replication.

IDS is good at delivering high-volume OLTP with minimal administrative overhead. The strength of its position in this space is demonstrated by the fact that despite a lack of sales and marketing, and despite a series of management and corporate changes that would have caused most product users to lose heart, IDS has not only held onto but also grown its user base.

This combination of performance and a low requirement for database administration is key to selling IDS as an embedded DBMS (that is, a database engine embedded inside an application) because such applications are typically sold in a context in which it's understood that a database administrator (DBA) is not required. Even companies that buy IDS explicitly, especially smaller ones, often lack the technical resources necessary to manage most other DBMS products. This fact is exacerbated by the manifest lack of available DBA talent, something that IDC has repeatedly found to be true in the course of researching database management software.

While every leading DBMS vendor is enhancing its product to make it more self-managing and easier to administer, IDS has been able to offer these characteristics for quite some time due to the elegance of its internal architecture in this regard.

All Database Workloads Are Not Alike

Database management systems are used to handle a wide variety of workloads, and each makes special demands on the DBMS that are best addressed by distinct attributes. It is for this reason that a variety of DBMS products has emerged, each having unique strengths and establishing great success, if not dominance, within its area of special capability. Examples of various workloads that require specialized technology include the following:

- Data warehousing, which is best served by a DBMS with strong complex query optimization and a solid index structure, built-in features that enhance analytic functions such as online analytical processing (OLAP) and a variety of tuning options that help the DBA adjust the performance of the data warehouse based on its data structure and patterns of use.

- Small footprint, including in-memory DB (IMDB) support, which places a premium on small size and quick and easy execution but typically lacks advanced functions and sophisticated query support.

- Specialized data structure support, including support for complex objects and multivalued fields.

- Complex transaction support for large amounts of data that generally includes support for very large databases (VLDBs) and also supports complex multiple join update transactions.

- High-volume OLTP support with little administrative overhead and a high degree of reliability.

IBM offers DBMS products that address a number of the preceding workloads, and although IDS can address aspects of a number of them, it addresses the last workload most fully.

The Special Needs of High-Volume OLTP

Why is it so hard to support high-volume OLTP in a low administration environment? Let's consider what has to happen when an update transaction takes place on the database. First, the row to be updated must be found, usually by searching the index (assuming random updates, which means that a table scan would be horribly inefficient). If the update requires a change to one or more indexes, or if the action is a row insertion (which would always require a key value insertion into an index for a fully normalized database), then the new value has to be placed in the index, which usually requires at least a small amount of reorganizing of the index, and space must be found on disk for the new row data or, if rows are being versioned, for the update data to an existing row. Meanwhile, other transactions are being processed from other users. Most DBMSs can handle low to moderate volumes of update transactions because they have time to make the adjustments to the indexes and disk space allocation before dealing with the next update. Only a few can handle very high volumes because in those cases, they must deal with indexes that are being altered and reorganized and with disk space allocation, even as other searches and updates are taking place.

Now, most of the DBMSs that can handle this type of workload do so because they have tuning features that enable the DBA to optimize their performance. Some also have self-optimizing features, although these DBMSs typically require at least some monitoring if not occasional adjustment or correction. Ideally, one wants a DBMS that has an internal architecture that is so adept at high-volume OLTP that little tuning or self-tuning is required in the first place. Such a DBMS would need to have a very efficient indexing architecture, a well-conceived means of laying out data on disk in an automated manner, and excellent thread management to deal with record locking and transaction resource management.

The Challenge of Embedding High-Volume OLTP DBMS Software in Applications

The need for a self-managing DBMS, requiring little or no administration, is especially acute for embedded DBMSs and for databases that are to be deployed in locations where DBAs are not available to manage them. Such DBMSs must also have virtually no unplanned downtime, since there is no technician available to effect recovery. When one combines these requirements with that of high-volume OLTP, the list of DBMSs available on the commercial market that can fill the bill is very short indeed.

How IDS Rises to the Challenges and Addresses the Requirements of High-Volume OLTP

Because IDS was developed to serve the needs of smaller enterprises with limited staff but big database challenges, the product has evolved naturally to address the needs of high-volume OLTP with a limited administrative requirement. Specifically, the requirements met are as follows:

- **High-performance OLTP.** The buffer management and indexing features of IDS, combined with internal thread management, enable this DBMS to offer a high-performance OLTP capability.

- **Low administration.** Because the design of IDS enables long-term operation without administrative intervention, IDS has required very little staff involvement to operate well, according to customers contacted by IDC. These capabilities are augmented by an API introduced in release 11 to permit applications in which the DBMS is embedded to perform programmatic administrative activities to deliver a zero administration environment.

- **Robustness.** A uniform story that IDC has heard from all the customers interviewed is that IDS runs so reliably that they could not even provide unplanned downtime statistics. To protect against exegetical factors (such as power failure or some site disaster), IDS offers a high-availability option called High Availability Data Replication (HDR), enabling a reliable mirrored database with failover support.

All in all, IBMers like to compare the choice of IDS over more heavily optioned RDBMS products to the choice of certain kinds of cars that have fiercely loyal customers because they are so reliable. IDS, they say, is like those reliable cars that don't get a lot of attention but keep running and performing well. IDC can attest to the loyalty and commitment of the IDS customer base.

IDS as an Embedded DBMS: Resiliency, Agility, and Invisibility

As mentioned earlier, IDS' robustness and zero administration features combine with a resource-stingy architecture to form a product that seems well suited to its use as an embedded DBMS. IDC has spoken to a number of vendors that embed IDS in their products and that have attested that their customers are not aware that IDS is inside, nor need they care.

Key Features of IDS Release 11

IBM has announced that the key features of IDS release 11 are in the following areas:

- **High availability, scalability, and performance.** These features combine to deliver benefits not only in terms of reliability of the application but also in terms of lower recurring hardware costs due to lower system utilization than before and lower staffing costs because expensive DBA resources can be dedicated to application development and improvement, which are high-value activities, rather than to database tuning and recovery.

- **Enhancement of high-availability features to include distributed data access and availability.** In release 11, HDR is expanded to support multiple replicas. A replica may be implemented as a remote standalone secondary (RSS) or as a group of remote replicas, one of which is the failover node. These are available individually as read-only databases for query and reporting and may be accessed simultaneously by multiple systems using the shared disk option, in which case they are called shared disk secondaries (SDS). The IDS feature that includes the SDS technology is called the Continuous Availability Feature (CAF) and supports deployment on a variety of server platforms including the IBM BladeCenter.

- **Zero administration.** Using the zero administration features, including the new administrative API, ISVs can embed IDS and ensure that most routine administrative situations are handled programmatically, which extends the range of applications for which IDS is suitable as an embedded DBMS. According to IBM, a number of large IDS users including several large U.S. retailers and a major Canadian drug retailer each report over 1,000 database instances running under the management of a single DBA.

- **Application development.** IDS has a host of new and improved development tools to ease database development and reduce the need for special knowledge of or expertise with IDS on the part of the developer.

- **Security.** IBM has added label-based access control (LBAC) to IDS to greatly enhance its ability to hide sensitive data from prying eyes.

- **Complex data support.** IDS release 11 features hierarchical data support in the Node DataBlade (which steps outside the relational model to provide explicit support for complex structures such as employee structures and parts implosion/explosion structures).

- **Web 2.0.** IBM has been making a big push to support Web 2.0 through features of various products that deliver the ability to support blending content, mixing data and content, and providing better support for a wide variety of data types in the database. IDS offers XML Publishing (providing SQL interactions and results in terms of XML commands and documents), the Basic Text Search DataBlade (which provides the ability to integrate basic text search functions into queries), and an implementation of the Open GeoSpatial Consortium® Web Feature Service (OGCWFS, which enables a standardized search of geospatial and geodetic data using a form of XML called GML, or Geography Markup Language, as a Web service).

FUTURE OUTLOOK

Demand for data management continues to grow. In addition, the range of database applications is increasing, and many of these applications will need to be deployed in situations where DBA support is not a realistic option. These applications will need to manage a wide variety of data types (including geospatial, various forms of XML, and complex structures) and will integrate content into their operations. The new classes of applications under development today are component-based, service-oriented architecture (SOA) applications that are designed to collect, hold, process, and transmit data within an integrated framework. Speed and reliability are of paramount concern.

The integrated application environment of the future will involve a wide variety of problem spaces and data workloads and will require a wide variety of DBMS products to support it. This means that the range of types of required DBMSs is growing, not shrinking. An important DBMS in this scenario is one that is efficient, requires little administration, yields high throughput, and is always on. Based on IBM's information and on conversations with customers, it may be safely said that IDS has these characteristics. Furthermore, release 11 features suggest that IDS' future development is vectoring in the right direction to meet the needs of an integrated application environment.

CHALLENGES/OPPORTUNITIES

The DBMS marketplace remains dynamic, despite the apparent decrease in the number of competitors. Leading DBMS vendors that once offered just one product each now feature a range of DBMS products, bringing to an end the era in which any DBMS vendor could argue that all a company's data could or should be stored on one database, or even one DBMS. New specialized competitors continue to emerge, with products that address part of the IDS value proposition. Both new and established DBMS vendors will continue to provide vigorous competition for IBM's IDS.

CONCLUSION

When it comes to databases, one size most definitely does not fit all. Different types of databases, and different types of database workloads, demand different DBMSs. IBM, for example, offers a variety of DBMSs, most of which are essentially relational. Each serves a different database management problem space.

The problem space that IDS addresses is that of the high-volume, low maintenance, resource-stingy DBMS for deployment in transactional applications meant to be deployed in environments in which DBAs are scarce or nonexistent, downtime tolerance is very low, and yet transaction volumes are high. In short, IDS ought to be considered for any database implementation with the following requirements:

- High-volume throughput
- Flexibility of deployment
- High-availability features and capabilities
- Specialized features including spatial support and security
- Low or no administration

CASE STUDY

NetworkIP serves the prepaid phone card industry with innovative telephony and stored value services, including issuing accounts and tracking of prepaid calling cards; managing the business processing necessary to connect calls; tracking usage and managing the cost and transaction records for each account ID; and performing additional services such as quality assurance, billing, and card recharges. IDS is the backbone of NetworkIP's system, driving the databases that manage all the data necessary to perform these services. NetworkIP handles over 2 million calls each day and performs as many as 100 database accesses per call, with additional database processing once the call is connected.

NetworkIP's data is distributed over three types of databases — account, transactional, and content management. The company manages 17 servers hosting the various database engines. Twelve servers run with a hot standby database configuration using HDR. The other five servers run in a distributed load-sharing cluster and utilize Enterprise Replication (ER). Failures are extremely rare, and so far they have been caused only by hardware problems, not software problems. When failures have occurred, failover has taken place automatically within a few seconds.

The seven account databases, each with 120GB, are extremely active, managing over 120 million live accounts. The account databases, which are absolutely mission critical to NetworkIP, are Dell 2650 servers running dual CPUs with 2GB of RAM each. The databases are stored on external drive enclosures.

All these NetworkIP databases are managed by a single DBA, Andrew Ford, who spends virtually no time on routine maintenance; such issues are handled automatically. He estimates that half his time is spent monitoring the databases and looking for ways to squeeze a little more performance out of them. Ford spends the rest of his time working with developers, helping them with new functionality, enhancements, and ways to improve their SQL. He also spends time on his own development projects.

Ford said that during the transition period from Informix Corporation to IBM, and during some of the shifts in emphasis within IBM, he was uncertain about the level of commitment to IDS customers. Since IDS 10, however, he has been extremely pleased with his relationship with IBM and the support he has received, though he rarely needs it. He was astonished at how the IBM channel partner helped NetworkIP save money by guiding him to the right combination of licenses to suit his needs. He spoke glowingly of the documentation, saying in particular that the online documentation is "a wealth of information."

Ford has worked with Informix technology at NetworkIP since 1999. Prior to that, he had spent a fair amount of his time in "reactive mode." Now, he is "heavily proactive," driving new development projects. He plans to move the servers to blade architecture on IBM hardware and intends to use Tivoli Storage Manager instead of manual methods of maintaining files and backups. Ford is proud to point out that he was the first DBA to be certified in IDS 11.