IDC
Analyze the Future

WHITE PAPER

Because Not All Data Is Flat: IBM's U2 Extended Relational DBMSs

Sponsored by: IBM

Carl W. Olofson
September 2006

IDC OPINION

The relational model of data management is the generally accepted model for storing and maintaining most business data. However, some data does not lend itself to maintenance in cross-referenced sets of flat lists, which is the paradigm under which the relational model operates. Some data involves recurring details or other nested structures that, if implemented using a pure relational model, would result in an inefficient collection of many tables and indexes requiring inefficient, multiply nested joins to access and update. The result is that the database for some kinds of data becomes, in the relational model, overly complex, slow, and difficult to administer.

This problem was recognized many years ago, prompting the development of various extensions to the relational model. While these extensions have never gone mainstream, they have proven quite valuable to those who need to manage complex collections of data, especially in commercially developed and delivered vertical applications. The most successful of these adaptations is sometimes called extended relational because it extends relational functionality by supporting the nesting of tables within tables without requiring extra indexes and joins for access. In areas such as manufacturing, logistics, and inventory management, this approach has proven invaluable, yet it is not well supported by the leading relational database management systems (RDBMS) products.

The IBM data servers, UniVerse and UniData, are examples of extended RDBMS products that have developed in response to such needs and continue to evolve today.

IN THIS WHITE PAPER

This white paper compares the strengths of the extended relational approach with the more commonly used pure relational approach and highlights various ways in which the extended relational approach provides an advantage over the pure relational approach in terms of development, performance, and maintenance of the database. It also discusses two products that held early leadership positions in this area, are still widely used, and are actively developed and marketed by IBM: UniVerse and UniData. This white paper discusses their suitability for use, especially embedded in vertical applications with special data management requirements.

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

SITUATION OVERVIEW

The Evolution and Significance of Extended Relational DBMSs

From the earliest days of business computing, there was a recognized need to manage data in forms other than fixed serial streams of formatted inputs and outputs. One of the earliest systems designed to provide structure and random access to data was the Generalized Information System (GIS), which was developed at TRW by Dick Pick and Don Nelson. Pick further developed this system into a multidimensional DBMS known as the Pick System, which was also an operating system (OS). A DBMS based on the Pick model was developed for Prime computers and integrated into the Prime OS as an optional data management layer called Prime INFORMATION. It was commonly believed in the late 1960s and 1970s that operating systems and database management systems were part of the value proposition of computers, so they were built together and offered along with the hardware as a single product. Both the Pick and the Prime DBMSs were integral components of their respective OSs and could not be run apart from them.

RDBMSs emerged in the mid-1980s following the theories of IBM Fellow Dr. Edgar Codd. They became the dominant mechanism for managing business data on midrange and mainframe computers. Unlike the Pick and Prime DBMSs, however, they were OS independent running on VMS and Unix. The OS independence, along with the simplistic representation of data in the form of tables having rows and columns, was a major factor in RDBMSs' becoming dominant as general-purpose databases. Yet despite the popularity of RDBMSs, some data management practitioners, particularly many ISVs looking to embed a database within an application, found the relational model inadequate, or at least inefficient, at storing data that consisted of complex hierarchies or nested structures. The Pick approach seemed superior for such applications.

To seize this market opportunity, a company called VMark produced a DBMS that was functionally equivalent to the Pick DBMS but could be run on Unix, and another company, Unidata, produced a DBMS (also called UniData) that was compatible with the Prime INFORMATION DBMS but also ran on Unix. Both products met with initial success due to features of simplicity, flexibility, performance, and finely tuned footprint characteristics (memory and storage requirements that could be tailored to the needs of the application) combined with self-managing capabilities that RDBMSs could not fully match.

IBM's UniVerse and UniData

In the 1990s, VMark and Unidata extended their products with support for SQL and the tables-and-columns paradigm while still offering nested structures and multivalued fields. They called their products "extended relational," "nested relational," or "multivalue" DBMSs. These products proved especially useful as embedded DBMSs, offering support for data structures that ISVs needed their software to manage more powerfully and efficiently than could be achieved using purely relational DBMSs. As a result, both UniVerse and UniData grew robustly. As the 20th century neared its close, VMark merged with Unidata to become Ardent Software. The company soon

took a different direction, pursuing the red-hot data movement software market with its extract, transform, and load (ETL) product DataStage while continuing to support and develop UniVerse and UniData. Ardent was acquired by Informix Software, and when IBM acquired the database software assets of Informix in 2001, UniVerse and UniData became IBM products. They are commonly referred to as U2.

Why Relational Is Sometimes Not Enough

The relational paradigm is based on mathematical set theory and represents an efficient means of cataloguing data. It should be borne in mind that Dr. Codd conceived of the relational model as a paradigm governing the relationship between the database and the application and not as an architecture for the physical organization and storage of data. The relational model has distinct limitations when the storage and retrieval of data are limited to the flat table structure that such a model demands. These limitations are particularly apparent when the data involves nested structures because nested data requires separate tables (in some cases) or cross-reference tables in order to support navigation among embedded hierarchies of rows (such as in a bill-of-materials structure). Such nested structures must be stored and retrieved with heavy reliance on indexes and using patterns of retrieval that are hard to optimize.

UniData and UniVerse, which natively support nested tables and multivalued fields, achieve much greater speed and efficiency in storing and retrieving such data. Although these constructs are not supported in a normalized database and are not well handled by most relational database management systems, multidimensional DBMSs such as U2 are designed to manage them extremely well. In addition, the U2 DBMSs support variable-length fields and files in order to deliver storage efficiency. Retrieval and storage of all related data in a single physical record eliminate the overhead that multiple tables incur from time-consuming joins and delays caused by lock contention. As a result, programmers and database developers realize significant performance and scalability benefits through simple and easy-to-understand data constructs. The ability to add fields to a data record without the requirement to rebuild the database makes the U2 DBMSs very popular with ISVs.

Meeting the Requirements of an Embedded DBMS

The requirements of DBMSs that are embedded in software products differ from those of general-purpose DBMSs in several key respects. The general-purpose DBMS must provide explicit database definition and management facilities for the database administrator (DBA), whereas the embedded DBMS does not because it manages data that is predefined specifically for the software product in which it is embedded. Because managing databases is a fairly sophisticated task, enterprises tend to favor the DBMS that is best known and understood by their DBA staff. By contrast, the embedded DBMS becomes part of the product in which it is embedded and requires no DBA to manage it.

A key basis for the evaluation of a general-purpose DBMS is its long-term direction, as well as its ease of use and its fit within the IT environment. The DBA staff must have members who are fully trained and competent to manage it over time. The embedded DBMS, by contrast, is invisible to the end user (or nearly so, at the embedder's discretion) and does not require an expert DBA staff to manage it. Because it is part of another software product and not visible to the user organization, its key evaluator is the ISV that is considering embedding the DBMS within its software product. Such an evaluator is mainly concerned with such issues as how efficiently it handles the product's internal data, how well it fits within the footprint and performance profile of the product overall, and how well its licensing fits within the pricing scheme of the end product. Ease of inclusion in the product, ease of application modification, and good support services are key elements as well.

In short, the requirements of an embedded DBMS are as follows:

- Its performance and efficiency characteristics must fit within those required by the software product in which it is to be embedded.
- Its footprint must be small enough to make it a reasonable component for the product in which it is embedded.
- Its pricing and licensing must be a good fit within the pricing and licensing model for the product in which it is to be embedded.
- Its vendor must provide excellent support to the ISV's development team and must provide excellent second-level support to back up the ISV's support organization.
- It must be utterly reliable because users will not be able to fix or troubleshoot problems.
- It must be capable of running within the product in which it is embedded without a DBA, adjusting its settings dynamically as needed or enabling the software within which it is embedded to tune its operation through an API.

Today's Development Challenges

As a new generation of software products and components is being developed for "real time" and integrated enterprise, Web-based, and service oriented architecture (SOA) implementations, developers are pushing against limitations of the relational paradigm once again. While RDBMS vendors have made enhancements that extend their capabilities beyond the strict limitations of relational data management, the "extended relational" approach is proving to be the right solution for some software vendors that need to manage complex data structures in an "on-demand" world.

U2 and XML

The extended relational DBMS, with its multivalued fields and nested structures, is a perfect fit for XML, which is a data organization text structure that is hierarchical and supports lists of values. IBM has extended the U2 products to provide numerous features that enable XML documents to be stored, searched, and retrieved quite easily in the database. These features include XMAP and DOM APIs to facilitate standardized support for managing XML data in the database.

XML is also the base data structuring technique for exchanging data between Web service requestors and providers. Such exchanges are governed by protocols defined using Simple Object Access Protocol (SOAP). The U2 products provide facilities that make it easy to define SOAP structures and publish data access capabilities as Web services with full Web Services Description Language (WSDL) definitions. U2 subroutines may also be published as Web services. These capabilities do not require developer knowledge of SOAP or WSDL.

The U2 products offer object-oriented access through a feature called UniObjects. This facility includes an abstract class called UniXML, enabling developers to quite straightforwardly define object structures that correspond to XML document structures for inclusion in .NET and Java programming frameworks. This facility also supports the exposure of U2 data through ADO.NET. (ADO, or Active Data Objects, provides application data services as objects in a .NET environment.)

U2 Within IBM

IBM has identified UniData and UniVerse as key data servers within its Information Management division. The company believes these servers will help drive its "Information as a Service" initiative, aligned within a framework that includes DB2 (for general-purpose data services); WebSphere Information Integrator; IBM Content Manager; and services for managing, publishing, and distributing information as a service from users' in-house applications, major packaged applications, and databases managed by non-IBM software.

The key to understanding this approach is to accept that most IT installations will continue to be heterogeneous and that in-house IT increasingly will need to integrate with outsourced IT services. Addressing the need to make such environments coherent and sensible from an information access and management perspective requires embracing an approach to database management that assigns the right technology to each element of a framework that includes corporate data, applications and tools, internal data, XML documents, and Web service data and that creates the metadata that makes sense of it all. The U2 products include the technology that allows them to play an important role in many of those areas.

FUTURE OUTLOOK

The Embedded DBMS Market

IBM's U2 products had a 9.4% share of the $1.2 billion embedded DBMS market in 2004, in fourth place within this highly fragmented market space. Competitors are numerous, and many are customized to address specific types of embedded data management challenges. IDC analysts forecast this market to grow to $2.2 billion in 2010, for a compound annual growth rate (CAGR) of 11.2%. This CAGR is more than three percentage points higher than the CAGR for the relational DBMS market, which is the fastest-growing functional DBMS market. This finding shows the growing opportunity for embedded and embeddable DBMS products, and IBM's U2 products are in a position to take full advantage of it.

IBM's Commitment to UniVerse and UniData

There has been considerable discussion of IBM's commitment to the U2 products. With so much emphasis on DB2, one might be excused for believing that the U2 products are "on the shelf," just collecting maintenance revenue from a static and shrinking base. IBM has made it clear, however, that this is not the case. Recent moves to support SOAs, Web services, .NET and Java object frameworks, and XML underscore the significance that IBM assigns to these products, particularly within its broader Information as a Service initiative.

The fact is that relational DBMS products are not well suited to address certain data management problems, particularly in an embedded database context. IBM is aware of this issue and knows that the U2 products are powerful arrows in its quiver if it wants to be able to address the complete range of data management issues that face IT organizations today. In fact, IBM states that "data servers are not a commodity any more than the information they contain but are key business assets. Data servers have unique capabilities which are leveraged for their business value and IBM offers a full spectrum of data servers to meet a spectrum of needs. The unique capabilities of UniVerse and UniData are successfully providing value across a wide array of vertical industries."

CHALLENGES/OPPORTUNITIES

Barriers to the success of the U2 products include some reluctance in the marketplace to accept nonrelational approaches to database management. Fortunately for IBM, the limitations of relational data management and the need to extend relational DBMSs with functionality to address these limitations have been addressed not only by IBM with extenders for DB2 and blades for Informix Dynamic Server but also by Oracle with its object support and by other RDBMS vendors. These developments have served to open up the market to extended relational, or multivalued, database concepts. A key technology driver has been the emergence of XML, which has a hierarchical structure that is clearly not best supported by a purely relational approach.

Another challenge is the perception, as mentioned above, of U2 as a second-class level of product within the IBM portfolio. IBM has demonstrated its intention to keep faith with its customers and partners in a variety of contexts. U2 clearly illustrates this ongoing commitment. IBM's best move in this regard, apart from ensuring successful and satisfied customers and partners, is showing how U2 fits into its broader strategic plans by including U2 in its Information as a Service initiative.

CONCLUSION

The relational paradigm is important for enterprise data management, but it does not represent the best approach for all types of data management, especially where such data requires nested structures or multivalued fields. For this reason, a number of nonrelational or extended relational DBMSs have found great success, particularly UniData and UniVerse from IBM.

IBM's U2 products have a rich history, but more importantly, they are a key fit in growing areas of data management going forward, including XML, Web services, and standardized object frameworks. These products do not share the limitations of purely relational DBMSs and have shown their worth over many years of use in a wide variety of applications.

The U2 products feature characteristics that are key to success in the embedded DBMS world, including:

- DBA-less, self-managing operation
- Performance and efficiency suitable for embedding in a range of software products
- Reliability sufficient to give partners the confidence to embed the technology in their products without worrying about support issues

IBM's commitment to these products has been demonstrated in its ongoing investment in their development and its ongoing effort to support U2 partners and customers. IBM's greatest shortcoming in this regard could be said to be its lack of promotion of these products, leaving them as perhaps the best-kept secret of the DBMS world. IBM is moving to address this issue, however, by bringing the U2 products to light as important elements of its information management product family and strategy.

CASE STUDIES

APT Solutions Limited

APT is a United Kingdom–based firm that provides IT services, specifically membership and subscription systems, to the not-for-profit sector. Its clients include large trade unions and professional institutions in the United Kingdom and Australia. APT has been in operation for 18 years. It initially worked with a Pick-based system but switched to UniData in the early 1990s.

Development Flexibility and Efficiency

APT feels that the flexibility of UniData has given it the ability to provide custom data management for its clients that would be difficult, if not impossible, to accomplish with a relational DBMS. For instance, its application includes a large member file, which is a component of the database in which 60–70 facts may be recorded about each member. Clients often expand that number to 120–130 items, which are not coded into the application or schema in advance. UniData can handle that customization without requiring a schema change or any sort of reformatting of the data files. In fact, APT indicated that it is easy to add or change data fields at will, without requiring schema adjustments or data unload and reload operations.

APT likes the convenient way that addresses can be stored as multivalue tables, avoiding the unnecessary complexity and inefficiencies of the sorts of table structures needed to handle mailing addresses in a normalized relational database. The company also claims that it can make changes to applications that affect data structures four to five times more quickly using UniData than it could with a normalized relational database. As a result, APT feels that it achieves a great deal of efficiency in the development process due to the underlying architecture of UniData.

How UniData Is Implemented by APT

The database is meant to support large volume transaction processing. APT has found it to be reliable and fully recoverable. Its stability is such that one of APT's clients reported that the database server, which was running Windows 2003 Server and supported 120 users, was down only once in the past four years, and that downtime was for scheduled maintenance. One reason for the server's reliability is its ease of administration (and therefore lack of opportunity for human error); APT described the database as largely self-managing.

Database workloads vary according to the needs of APT's clients. The smallest is a three-user system, while the largest supports 350 concurrent users plus a Web interface. The latest system, about to go online, is scaled to support 5,000 concurrent Web users. The databases are implemented on both Unix and Windows servers; the Unix servers include AIX, HP-UX, and Solaris.

Database sizes also vary; the smallest in terms of memberships has around 3,000 to 4,000 memberships, whereas the largest has around 2 million active records. The largest database, in physical terms, is estimated at between 50GB and 60GB, but APT believes that if it were implemented on a relational DBMS, the size would be closer to 200–240GB.

UniData Development and Support

APT indicated that although there had been some frustrations in dealing with its vendor in the past (Unidata, then Ardent, then Informix, and now IBM), the experience has been very positive, especially following the IBM acquisition. Since that time, IBM has proven a good partner, offering a much more structured support environment. APT was pleased to find that over the various corporate transitions, the technicians have stayed the same, and that now, in addition to that consistent experience and expertise, IBM offers much better administration of the relationship and strong marketing support.

As UniData has been assumed by successively larger vendors, both the available toolset and the ability of the vendor to support and develop the product have grown. APT feels that each vendor made "enormous" investments in the technology and that UniData has always been well planned and has a good future road map.

Datatel

Datatel is an ERP system vendor that provides integrated information management solutions for higher education. Datatel Colleague, the company's flagship product, encompasses a collection of applications for student enrollment, financial management, human resource management, institutional advancement, financial aid, and other functions. Datatel serves more than 725 colleges and universities, including two-year and four-year schools with enrollments ranging from 500 to 60,000 full-time equivalent students.

How UniData Is Implemented by Datatel

Datatel offers its product with several DBMSs, including UniData, as well as leading relational DBMSs. The company uses stored procedures in all of these products and uses its own internally developed language together with code generators to write the stored procedures in a DBMS-neutral format and then generate stored procedures in the language that is appropriate for each DBMS.

Although Datatel reports good performance with each DBMS, the company says that it gains efficiencies from UniData's flexibility that provide it with an advantage over relational DBMSs. This is particularly true with something called a status table, which tracks a student's status with respect to each course that a student adds or drops. In UniData, the status table is nested within the student table. Relational DBMSs require a separate table for the status, plus a cross-reference table and a set of joins via foreign key relationships.

Datatel also likes UniData's flexibility, which allows better support for variable-length character strings and makes adding fields to a record very easy. Also, with UniData — unlike with relational DBMSs — such structural changes do not require a database reorganization.

Most of Datatel's accounts run the UniData version of its product, including its largest client, which supports eight institutions on a single database. This client has tracked millions of students, covering such areas as enrollment, class registration, grades, billing, financial aid, and transcripts. The database is "hundreds of gigabytes" in size. It runs on a Sun UltraSPARC with 16 processors and is capable of keeping up with the 60,000 students who register during the three-day registration period.

UniData Development and Support

Datatel has worked with UniData for 18 years. The company says that it has always received excellent support, and it has dealt with pretty much the same people during that entire period. Datatel reports having been worried with each acquisition that there would be downsizing, that effective practices would be abandoned, and that key people would leave. In each case, however, its fears were unfounded. In fact, key senior people were often promoted rather than let go.

Datatel feels that it has direct input into what the UniData team plans for the DBMS and a close relationship with the software engineers. In general, the company believes the UniData embedded DBMS model allows IBM's U2 team to get very close to its ISVs, including Datatel. Datatel appreciates the regular release schedule that the UniData team has maintained. It reports that support for non-IBM platforms has been excellent. Datatel is looking forward to using the SOAP interfaces in the next release and also mentioned embedded XML support and support for C#, .NET, and Java as key to the company.

Datatel continues to grow, recording its best year ever in 2005. The company's business is centered almost entirely in North America (and the Caribbean), yet it established a large number of new accounts last year.