IDC
Analyze the Future

I D C T E C H N O L O G Y S P O T L I G H T

In the World of Digital Transformation, Data Is Not Flat: A Fresh Look at MultiValue Database

July 2017

Adapted from *The Shifting Data Management Landscape for Digital Transformation* by Carl Olofson, IDC #US41262717

Sponsored by Rocket Software

Digital transformation is opening enterprises to the idea that a pure relational database is not always the way to go in managing data. While many enterprises use data collectors for ingesting and ordering data, document databases for state or session data management, and wide column stores for narrow but deep data analytics, there is a range of data that is typically close to the user, is mostly tabular, and requires pliability in its definition and management. Thus users today are looking to different database management system (DBMS) forms. MultiValue (MV) DBMS is drawing interest because a great deal of data does not fit the flat table shape of relational but needs the formality of a schematic DBMS. From a data shape perspective, MV DBMS offers many of the same benefits as document databases, but with the manageability and scalable query performance that applications need. This Technology Spotlight examines the reemerging area of MultiValue databases and the role of these databases in facilitating digital transformation and looks at Rocket Software and its MultiValue database offerings.

Introduction

Digital transformation is expanding enterprise data management from systems of record to systems of engagement, from internal systems with external interfaces to blends of internal and external systems. It is also changing how we see and use data on a fundamental level. This transformation is beginning to impact IT organizations at all levels, moving enterprises from using data only in an internally controlled and managed form for the automation of fixed business processes to using data that is highly variable; that may come from external, uncontrolled sources; and that enables new modes of operation and analysis.

Transformation is being driven at the data level by a host of very different data management technologies that include such schema-optional (also called "NoSQL") data management systems as key-value (KV) stores, document databases, graph databases, and wide column stores. Data is being collected, cleansed, organized, and transformed in scalable data collection platforms such as Hadoop. It arrives in files but also in streams, and it sometimes demands immediate processing. Data comes from sensors, logs, mobile devices, and a wide variety of sources called the Internet of Things (IoT).

The data is used for many purposes, including:

- Enhancing data already in the data warehouse such as that used for business analysis
- Creating more specialized analytics such as network failure detection and fraud detection
- Managing state for highly interactive applications, including those for gaming, ecommerce, set-top boxes, and trading

- Driving real-time processes such as fleet management or shop floor control
- Enabling predictive analysis and machine learning

In today's climate of transformation, enterprises and software developers are reexamining their assumptions regarding the most appropriate database technologies for the problem at hand. For many cases of transactional data management, the pure relational model is not appropriate because it does not conform easily to the data's actual shape. For example, a data structure may have embedded data structures or lists of values. Such things violate the relational model, forcing the creation of multiple tables, with foreign key relationships and extra indexes. However, document databases also may not be appropriate because they lack the formalism of schematic structures that ensure structural consistency, a critical element for reporting.

A reasonable alternative is the MV DBMS. Like a relational DBMS, it supports table structures. Like a NoSQL DBMS, it supports embedded data structures and lists of data. It has a schema. It has indexes and supports several query models, including SQL. Several of these DBMSs were first developed decades ago, so they have the benefit of evolved capabilities and are "battle tested" in real-world scenarios, yet they have acquired the features required by modern databases deployed both on-premises and in the cloud. What also sets the MV DBMS apart is that it typically has an application development and deployment environment built in so that it can act as a data-centric application platform.

The MV DBMS can handle a wide variety of shapes, including tabular data such as that typically kept in a relational DBMS (RDBMS) but also data having nonrelational structures that may be derived from documents (such as JSON) or other formats, so that it can bring together a rich variety of data of varying structures. This capability facilitates the creation of composite services supporting both systems of engagement and systems of record and decision, thus coordinating information and presenting comprehensive views regarding the whole state of the enterprise.

Definitions

The following terms are used in the emerging database space as well as in describing relational and MultiValue databases.

- **Database:** An organized collection of data in a prescribed format that may be easily stored and retrieved and shared among applications
- **Schematic database:** A database that is governed by a schema, which defines its elements and structures, and so facilitates query and data sharing
- **Database management system:** A software system that manages a database; governs the contents of the database; enables the creation, reading, update, and deletion of the data; and ensures the integrity of both the data and its structures
- **Nonschematic database:** A database with no schema, although one may be attached to it optionally, and so the data structures are defined either within program code or encoded alongside the data (as tags or key-value pairs); also called a NoSQL database
- **Relational database:** A schematic database organized in a manner consistent with relational data theory into matrices called "relations" or "tables," with "tuples" or "rows" and "attributes" or "columns" (By rule, each table must have a primary key requiring a unique key value for each row. Every column value must be logically dependent only on that key. Every row-column intersection [or tuple attribute] may have one and only one value. Internal dependencies are not permissible. Thus, the data shape is flat, like a spreadsheet. Table rows may refer to rows in other tables through foreign key relationships.)

- **MultiValue database:** A schematic database that has sets of records (or tables), wherein the records (or rows) in each set (or table) have a consistent field (or column) format (The records can refer to records in other sets. They may also contain fields that have lists of values, and record sets may be embedded within records. For instance, a sales record may contain a set of item records, one for each item in the sale, embedded within it.)
- **ACID:** Stands for atomicity, consistency, isolation, and durability (This is a standard concept for the assurance that when data is stored in the database, all the data involved in a transaction is either stored or not stored [there are no partial operations]; the data is logically consistent according to the rules in the schema; each update transaction is independent of, or isolated from, other transactions so there are no unexpected data combinations; and the resulting changes to the database are permanent until the data is affected by a subsequent transaction. The ACID property is considered essential for any database that is involved in business transactions.)

Benefits

The MV DBMS offers significant advantages for a variety of applications. Benefits include the following:

- The database record structure can conform to the actual shape of business data without requiring complex foreign key relationships among many tables. This also leads to efficiency in data storage and retrieval.
- Typically, an MV DBMS has a rich application programming element, which makes it easy to build and extend database applications. This element can involve a native programming language, which enables the easy integration of business logic into database operations. Some MV DBMS products also support common scripting languages such as Python and JavaScript.
- Most MV DBMSs have evolved to support a wide variety of APIs and RESTful interfaces for ease of integration and interoperation with larger systems. This capability, combined with the MV DBMS' support for integrated business logic, enables that logic to be published as services, which simplifies modernization and the process of digital transformation while lowering the total cost of ownership (TCO) of such transformation.
- The simplicity of MV data structures and the reduced need for complex intertable operations lead to far simpler database administration. Often, an MV database can be run without a database administrator (DBA).
- MV DBMSs have evolved over a long period. Extensive suites of applications have been built upon them, and these applications have evolved and modernized over time. There is also a substantial body of experience in building applications for MV databases.
- MV DBMSs were originally developed, and continue to be enhanced, with business applications in mind, so they tend to be easy to use for managing business application data.
- The initial cost of design and development for MV DBMSs was amortized long ago, so this technology tends to be more affordable than others that might be considered for the same job, including open source subscriptions.

Trends

The following trends are reshaping the DBMS landscape:

- Embrace of a wide variety of data types (Digital transformation involves the addition of systems of engagement, which require database architectures that are different from those used for internal applications and enterprise data analytics. This trend is driving adoption of large data collectors such as Hadoop with Spark and an emerging class of data managers called NoSQL.)
- Capture of data from sources previously not fully processed or captured, including data from the IoT and streaming data (Previously, the former could not be fully captured because of volume and variable formats, and the latter could not be captured and kept because it arrives too fast. Flexibility in data format adoption [especially with NoSQL] combined with extreme scalability and memory-optimized database management has enabled these sources to be supported.)
- Support of smart handheld devices and their apps and the need to manage that data in the moment while also capturing it for later analysis
- Desire to perform analytics on live data and to blend some analytics with operational application execution (This trend is satisfied in some cases by in-memory columnar compressed data in relational DBMSs and in other cases by technologies that enable live data analysis.)
- Requirement that operational data kept in document databases be analyzed in wide column, relational, or other databases and that, in general, data and data services be freely mixed and available (This trend is supported in part through the emergence of RESTful APIs and more flexible data formats.)
- Gradual adoption of cloud for data management, starting with backup and high availability/ disaster recovery (HA/DR) functions and continuing with test and development deployment in the cloud (Enterprises are setting up private clouds and looking to expand to hybrid formats, including for peak resource utilization [so-called "cloud bursting"], though all this points to the eventual migration of production data to the cloud.)
- Demand for data security and private data protection (Recent concerns about data breaches have combined with concerns inherent in the movement and conversion of data to generate anxiety regarding the ability of bad actors to see data they shouldn't see. This issue is being addressed aggressively by developers of a variety of data management technologies.)
- Emergence of a more significant role for developers (There was a time when IT managers and development managers would dictate which technologies would be used for which development areas. Now, the developers are specifying the tools and languages they prefer. This change has contributed to the demand for open source systems and the use of a number of preferred languages.)
- Use of a platform approach that integrates the previously mentioned features in a single system, usable by both technologists and end users (The trend is toward delivering functionality in a coordinated platform of capabilities rather than delivering a suite of tools, many of which are dissimilar. These capabilities include data governance [e.g., metadata management, data security], analytics, and database application development support.)

The following trends are addressed specifically by MV DBMS:

- Demand for analytics against live data (The common practice of isolating data into separate databases for transactions and analytics is giving way to a desire to perform business analytics on live data.)
- The need for interaction with a wide variety of systems through open APIs
- The move to the cloud (This challenges DBMS vendors to provide systems that can exploit the virtualization and elastic scalability of resources.)
- Demand for data security, including good access control, encryption, and log security audits
- A developer-driven world (In such a world, providing support for and access to the right sets of development languages and environments is key to ongoing success.)
- A platform approach (Historically, MV DBMSs have included development environment and management features. These features need to be enhanced with security, HA/DR, and analytics support.)

Considering Rocket Software

Rocket Software offers MV DBMS products based on the proven principles of MultiValue data management developed in the late 1960s. These products have evolved to provide a range of capabilities for application software providers and end users who need to manage complex sets of data in a manner that is simple and easy to understand and that requires little formal administration and support. Products include Rocket D3 and the company's U2 offerings, UniVerse and UniData. They can run in virtually any computing environment, including Linux, Windows, Unix (AIX), and the leading public cloud environments.

These products share several common characteristics. For instance, they are self-contained, meaning that they require little in the way of setup of complexes of software programs to operate. Their design enables the user to install and set up the database and then just let it run.

These MV DBMS products also manage data in business structures more efficiently than most RDBMS products, enabling them to consume less CPU and less storage than any relational equivalent. This translates into less datacenter space and power savings. In addition, because of the intuitively simple nature of database structures, and the optimized way they are managed, these products also require little in terms of DBA, network administrator, or storage administrator time, delivering significant staff time savings.

The data structures match the way that business is transacted, so it is simple to align the data with its corresponding workflow. For workflow that involves other data systems, these products also feature tools that accommodate digital transformations. And just like any production DBMS, they offer full HA/DR support.

Rocket's MV DBMS products address the previously described trends in the following ways:

- **Demand for analytics against live data.** The design of the products enables analytic queries in the active database without disturbing transaction performance.
- **The need for interaction with other systems through open APIs.** Rocket Software's MV DBMSs feature RESTful services.
- **The move to the cloud.** Rocket's MV DBMSs have been deployed in both private clouds and public clouds, the latter including AWS, Azure, and Google Cloud. Multiple form factor capability also provides support for mobile devices.

- **Demand for data security.** Rocket's MultiValue application platform strategy includes standard authentication, automatic data encryption, and audit logging.
- **A developer-driven world.** Rocket MV DBMSs support two popular development language environments: Python and JavaScript. Python support is fully integrated with the native BASIC programming language embedded in MV DBMSs.
- **A platform approach.** Rocket's platform strategy includes metadata management, Rocket Discover for self-service analytics, Rocket CorVu NG for business intelligence (BI) development, HA/DR, provisioning, application life-cycle management, and other features.

Challenges

Cloud platform providers and other large data management technology providers are constantly changing attitudes about how data should be managed and optimized, especially for the cloud. As applications change and evolve, they are driving the need for supporting a greater variety of data formats and transformations. In this time of constantly changing data management models, competing approaches to handling schematic data in complex shapes, with simplicity and leading environment interoperability, will inevitably arise. It is for Rocket Software to continue to develop its MV DBMS technology, pushing the outside edge of the envelope while continuing to satisfy its customers to ensure ongoing success.

Conclusion

As enterprises engage in the process of digital transformation, they are increasingly examining their assumptions regarding DBMS technology and are open to choosing the right technology for each job. There are various database requirements, and no one technology can address them all. However, although RDBMS remains a common choice for transactional data, the growing need for flexibility in development and for a database that fits the shape of the data being managed represents a clear opportunity for examining other options. An MV DBMS may be the right choice, especially for applications that require deployment flexibility, ease of administration, and multifaceted operation.

Rocket Software's MV DBMS offerings operate in a low administration mode, with features and tools designed to make business application development quick and simple. The products support a variety of environments, including those of the leading cloud platforms, as well as the main development languages. Therefore, Rocket's offerings warrant serious consideration for database applications that require simplicity of development and deployment and ease of operations, but with the data governance features one would expect of an enterprise DBMS. To the extent that Rocket Software can address the challenges described in this paper, the company has a significant opportunity for success.

A B O U T   T H I S   P U B L I C A T I O N

This publication was produced by IDC Custom Solutions. The opinion, analysis, and research results presented herein are drawn from more detailed research and analysis independently conducted and published by IDC, unless specific vendor sponsorship is noted. IDC Custom Solutions makes IDC content available in a wide range of formats for distribution by various companies. A license to distribute IDC content does not imply endorsement of or opinion about the licensee.

For more information on IDC, visit www.idc.com. For more information on IDC Custom Solutions, visit http://www.idc.com/prodserv/custom_solutions/index.jsp.

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