

Viewpoint

David Menninger

SVP &

Research Director

The Value of Multi-Model Databases

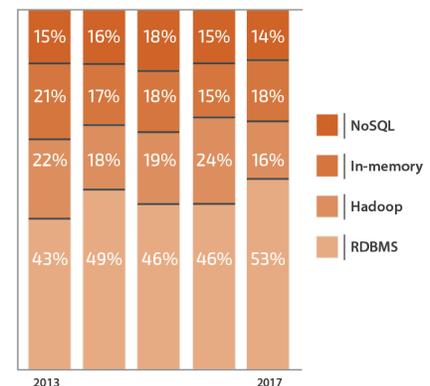
Relational database technology first appeared almost a half-century ago and for decades the database market appeared to be consolidating around the use of relational databases for most types of workloads, even if it wasn't very well suited to some of these workloads. More recently, though, as the big data market has exploded organizations have become less reliant on relational databases and appear more willing to use a variety of other database technologies including NoSQL databases to support their data processing needs.

As additional data models gained favor, relational database vendors responded by expanding their products to support a variety of data models, including multidimensional, document, in-memory, NoSQL, graph and others. Our research over the past five years shows widespread acceptance of multiple data model deployments. During this time, our research shows that the percentage of enterprises using different types of database technologies has remained relatively steady, and 40 percent of organizations have adopted a multi-model approach to their data processing needs involving two or more database technologies.

Rather than deploy multiple database products, organizations can deploy a single multi-model database for different types of data processing requirements. A multi-model database is a single database product that supports more than one type of data model — a document database that also supports a key-value data model, for example. For organizations using multiple database models, a multi-model database offers several potential advantages. There is no need to purchase, install and configure different products. Installing and managing a single database with multiple models requires less resources than installing and managing a collection of individual databases. Licensing costs for a single product can be lower than the license fees for a collection of individual products. And a single product with multiple data models provides easier access so users can experiment with different data models as different needs arise.

Multi-Model Deployments

A variety of models in use



Source: Ventana Research Benchmark Research
© Ventana Research 2018; All Rights Reserved



One of Two Approaches

Organizations generally take one of two approaches to multi-model databases: polyglot persistence with multiple database engines each handling a different data model or a single data store that introduces the data models at a higher level of the architecture.

Polyglot persistence offers the potential advantage of combining best-of-breed database engines into a single product. While easier to install and configure than multiple database products, organizations must still manage multiple copies of the data. If management of independent copies of the data is left to the users, it creates additional administrative, governance and consistency issues. If on the other hand management of the different copies of the data is built into the product, it creates additional overhead that could adversely affect performance.

A multi-model database with a single data store requires less administration because there is only one copy of the data. A single data store also eliminates consistency issues and can be governed and secured within a single framework. The data models are realized as layers or interfaces above the data store, allowing developers to use whichever interface is most appropriate. The vendors of these products typically create and manage a set of indexes to deliver performance comparable to individual database engines or multi-model databases using polyglot persistence.

Cost of ownership and performance are important considerations but organizations should keep other factors in mind as well. When evaluating multi-model databases, organizations should consider the variety of data models that the database supports. Also, consider which industry standard interfaces and languages are supported, such as SQL, Spark, JSON, ODBC, JDBC, and RESTful APIs. Some multi-model databases are also able to support transaction processing with ACID compliance. ACID is an acronym that stands for atomicity, consistency, isolation and durability, properties that guarantee validity of database operations even in the event of errors or failures such as problems with the network or power outages. For applications requiring multiple data models and transaction processing, a multi-model database may be the only option.

As our research shows, many organizations require support for multiple data models. Each organization needs to evaluate its own requirements. When weighing the options, consider whether a multi-model database can help reduce administrative and governance costs while providing the breadth of capabilities required to support your organization's applications.



David Menninger – SVP and Research Director, Ventana Research

David Menninger is responsible for the overall direction of research on data and analytics technologies at Ventana Research. He covers major areas including artificial learning and machine learning, big data, business intelligence, collaboration, data science and information management along with the additional specific research categories including blockchain, data governance, data lakes, data preparation, embedded analytics, natural language processing (NLP) and IoT.