Chief Technology Strategist Notes

Volume 1, Issue 1
December 2013

Office of Technology Strategies (TS), Architecture, Strategy & Design (ASD)

A VA Executive's Guide to Databases

INTRODUCTION

Access to and use of data drives enterprise decision making at all levels. Whether you are a clinician, benefits manager or memorials executive, whether you have ever actually written a line of software code in your life or not, you have probably been in at least one meeting during which the conversation turned to databases, metadata, big data, structured data, unstructured data, and all sorts of other data.

Understanding data and databases, including the technical terminology and specialized language surrounding them, is critical to having the ability to make decisions that will better position the VA to meet its strategic goals. This paper highlights key topics and terms that people must understand in order to make informed decisions when faced with the nerdy and complex discussions about data and databases.

BACKGROUND

VA business offices use IT software (applications) every day to efficiently and effectively deliver services to our nation's Veterans. A basic understanding of how modern software works is important to understanding different types of databases.

Typical modern applications have different layers that reside on different servers. In general, modern software systems have three types of servers:

- Web Servers host the web pages users see and utilize to interact with the applications
- Application Servers host the software that pulls data out and puts data into one or more databases, and manipulates or changes the data based on user input
- Database Servers are where applications store or "persist" data

Modern software keeps these three functions separated to allow for the building of secure and robust applications that can be easily modified, updated, and maintained. This concept is often referred to as "separation of concerns" or a "three tier architecture".

DATABASES

Data can be stored or presented in many different ways. Understanding how databases function can help business leaders select the technologies that will deliver the best business results for their specific operational requirements.

A database is:

- A collection of data typically organized to support processes requiring certain information
- An application or software suite used to manage this collection of data, usually referred to as a database management system (DBMS)

Depending on how an application uses it, a database is programmed to provide the following capabilities:

- A way to Create, Read, Update or Delete data, often using what are called CRUD services
- The structure or model of the data stored therein
- "Administration," which includes things like security, access control, and performance

Where databases tend to differ is:

- The language or "syntax" used to Create, Read, Update or Delete data
- The structure or model they use to store data

Databases are most often categorized by the data model or structure they support, as well as the process or query language they use to store and retrieve the data. The two most common types of databases on the market today are Structured Query Language (SQL) databases and Not Only SQL databases.

SQL databases are also known as "Relational" Database Management Systems (RDBMS). An easy way to think about a SQL database is like an Excel spreadsheet in which every worksheet is a "table" and the workbook with all the worksheets is a "database." In fact, you can easily export an Excel workbook into a SQL Database. The SQL market is dom-

(Continued on page 2)

Technology Strategies

Defining OIT's "To Be" Technology Vision

This newly established office within OIT's Architecture, Strategy & Design (ASD) interacts not only with the ASD pillar offices, but also with multiple stakeholders within OIT and with strategic offices across the enterprise. TS works closely with IT and business owners to capture business rules and provide technical guidance as it relates to Data Sharing across the enterprise, specifically for inter-agency operability.

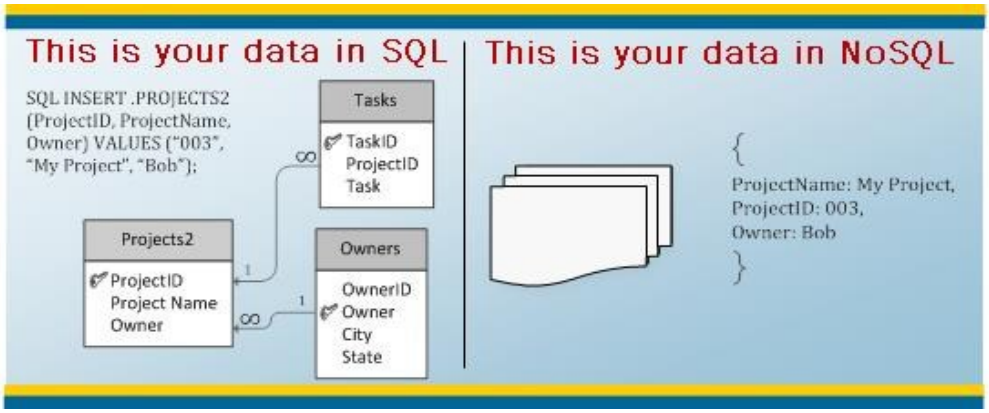
Chief Technology Strategist Notes

Volume 1, Issue 1
December 2013

A VA Executive's Guide to Databases, cont'd

inated by several well established products such as Oracle, MySQL, Microsoft SQL Server, and IBM FileNet.

NoSQL databases are more like a file cabinet of manila file folders full of cut sheets from all sorts of different catalogs. In a NoSQL database, these documents are "JSON documents". NoSQL databases are not new, but have just recently become popular in large part because JSON is the data format used by the web technologies used to mobile device applications. NoSQL has also become extremely popular because of its ability to aggregate large quantities of disparate data (sometimes referred to as "Big Data" or "Data Aggregation"). The NoSQL market is currently dominated by relatively smaller competitors, such as MongoDB, CouchDB, and MarkLogic. However, Oracle and IBM are now entering this market.



The figure shows two different data structures for Project 003 owned by Bob. The left portion shows Bob's project in SQL format using a rigid schema relating different data elements. The right portion groups Project 003 into a document format without requiring a rigid schema, resulting in simplified data access.

SQL	NoSQL
Established Standard Query Language (SQL)	No standard syntax for adding, changing, or deleting documents
SQL is best for situations in which you know exactly what data you need to store and share, and, the structure of that data is well known and doesn't change. In these cases, it provides extremely robust performance for applications that have high volumes of a limited number of well-known transactions. Specific SQL strengths include the ability to handle the following situations extremely well: • Exact data elements are well known and must follow established standards • Many users need to save and/or view the same type of information at the same time • Controlling how and what data is (i.e., "data consistency") is paramount	NoSQL is best for situations in which the structure of the data is either unknown or subject to frequent changes. It is also very strong when you need the ability to make "ad hoc" queries against data from many different sources. Specific NoSQL strengths include the ability to handle the following situations extremely well: • Exact data elements are unknown • Data sharing requirements are unknown • Lots of different people will want to use (i.e., "query") the data different ways • Data needs to be collected from multiple disparate sources.
In the commercial world, SQL is still used for everything, but is strongest in high volume "transactional" systems like ERP and Point of Sale systems.	In the commercial world, NoSQL is often used to aggregate customer or sales data from many different (SQL) systems. In DoD/Intel, NoSQL supports collection and assessment of huge volumes of intelligence data from many sources.

SO, WHICH IS BETTER?

Don't be misled by marketing jargon. All the major products are capable of providing ample scalability, security and performance for most applications. Selecting a database product is really about picking the technologies best suited to the specific business problem you are trying to solve. The structure of relational databases makes them able to find and read data very quickly. It also enforces a level of discipline with respect to data definitions and standards. The lack of structure makes NoSQL able to store data more quickly and enables "ad hoc" operations when you don't know exactly what you are storing, accessing or sharing.

Selecting a database requires understanding the attributes of both your business process/problem, and, the attributes of the various products. While the table to the left provides a simple comparison to help you make a more informed decision, don't hesitate to ask CTS (askCTS@va.gov) for assistance or more information.