

: *What is a document database?*

by Dan Power

Editor, DSSResources.com

Increasingly unstructured or semi-structured documents are the drivers of new, novel decision support systems. In the expanded framework (Power, 2002), DSS linked to a document database are called document-driven DSS. A number of approaches can be used to store and retrieve documents for decision support including: 1) storing the documents in directories or files, 2) using a RDBMS to store documents or some document metadata with a link to the complete document, 3) using a document-oriented database. Document databases are a type of "NoSQL" or XML database. Documents may be stored using markup, XML, PDF and Microsoft Office formats.

Documents are the organizing structure in a document database. Conceptually a document is similar to records or rows in relational databases, but they are less structured. Documents do not adhere to a standard schema with structured fields, sections, slots, parts, or keys. According to Krishnan, "Document-based databases do not store data in tables with uniform sized fields for each record. Instead, each record is stored as a document that has certain characteristics. Any number of fields of any length can be added to a document."

Wikipedia has the following example of a document:

```
FirstName="Bob", Address="5 Oak St.", Hobby="sailing"
```

Another document could be:

```
FirstName="Jonathan", Address="15 Wanamassa Point Road", Children={{Name:"Michael",Age:10},  
{Name:"Jennifer", Age:8}, {Name:"Samantha", Age:5}, {Name:"Elena", Age:2}}
```

"Both documents have some similar information and some different. Unlike a relational database where each record would have the same set of fields and unused fields might be kept empty, there are no empty 'fields' in either document (record) in this case. This system allows new information to be added and it doesn't require explicitly stating if other pieces of information are left out."

Cattell (2011) identifies 3 types of NoSQL data stores -- key-value store, document store and extensible record store.

: *What is a document database?*

According to Cattell, "Key-value Store provide a distributed index for object storage, where the objects are typically not even interpreted by the system: they are stored and handed back to the application as BLOBs."

"Document Stores provide more functionality," according to Cattell. For example, "the system does recognize the structure of the objects stored. Objects (or documents) may have a variable number of named attributes of various types (integers, strings), objects can grouped into collections, and the system provides a simple query mechanism to search collections for objects with particular attribute values."

Finally, "extensible Record Stores, sometimes called wide column stores, provide a data model more like relational tables, but with a dynamic number of attributes, and like document stores, higher scalability and availability made possible by database partitioning and by abandoning database-wide ACID semantics."

According to Ayende Rahien, "A document database is, at its core, a key/value store with one major exception. Instead of just storing any blob in it, a document db requires that the data will be store in a format that the database can understand. The format can be XML, JSON, Binary JSON (MongoDB), or just about anything, as long as the database can understand it."

Ayende notes "A document database is schema free, that is, you don't have to define your schema ahead of time and adhere to that. It also allow us to store arbitrarily complex data. If I want to store trees, or collections, or dictionaries, that is quite easy. In fact, it is so natural that you don't really think about it."

A major limitation of document databases is limited query capabilities. Ho (2009) notes "Many of the NoSQL DB today are based on the DHT (Distributed Hash Table) model, which provides hash table access semantics. To access or modify any object data, the client is required to supply the primary key of the object, then the DB will lookup the object using an equality match to the supplied key." Developers need to organize the indexing.

For more on NOSQL databases, check Power [What is Hadoop?](#).

: *What is a document database?*

Finally, Rick Osborne (2010) reviewed two non-relational database alternatives, Apache CouchDB and MongoDB. Each has advantages and he concludes "a document-oriented database is just what it sounds like: a database of entire documents."

References

Cattel, R., "Relational Databases, Object Databases, Key-Value Stores, Document Stores, and Extensible Record Stores: A Comparison," February 2011 at URL <http://www.odbms.org/download/RickCattell.pdf> .

Dean, J. and S. Ghemawat, "LevelDB: A Fast Persistent Key-Value Store," July 27, 2011, at URL <http://google-opensource.blogspot.com/2011/07/leveldb-fast-persistent-key-value-store.html> .

Ho, R., "Query Processing for NOSQL DB," November 28, 2009 at URL <http://horicky.blogspot.com/2009/11/query-processing-for-nosql-db.html> .

Jones, R., "Anti-RDBMS: A list of distributed key-value stores," January 19, 2009 at URL <http://www.metabrew.com/article/anti-rdbms-a-list-of-distributed-key-value-stores.p>

Krishnan, H., "Document Oriented Databases," Geek Snack, Monday, June 8, 2009 at URL <http://geeksnack.blogspot.com/2009/06/document-oriented-databases.html> .

Osborne, R., "SQL or NoSQL?" February 14, 2010, at URL <http://rickosborne.org/blog/2010/02/sql-or-nosql/>.

Power, D. J. Decision support systems: Concepts and resources for managers. Westport, CT: Greenwood/Quorum Books, 2002.

Power, D. J., "What is Hadoop?" DSS News, Vol. 12, No.23, November 13, 2011 at URL <http://dssresources.com/faq/index.php?action=artikel&id=235> .

: *What is a document database?*

Rahien, A., "That No SQL Thing – Document Databases," April 11, 2010 at URL
<http://ayende.com/blog/4459/that-no-sql-thing-document-databases>.

http://en.wikipedia.org/wiki/Document-oriented_database

Author: Daniel Power

Last update: 2011-11-26 11:56