See Also:

The UBC School of Library, Archival and Information Studies Student Journal

2015 - Spring

An Exploration of Archival Finding Aids Technologies and NoSQL Databases

Stephanie Fan - stephanie.y.fan@gmail.com

Stephanie is currently finishing her Master of Library and Information Science degree at the University of British Columbia. She is interested in information technologies and data.

Keywords: archival studies; information studies; NoSQL; MongoDB; finding aids; schema design; formats

Abstract:

While NoSQL databases, specifically document databases such as MongoDB, are not the answer for the majority of storage use cases, technical contributors to the major finding aids information systems should consider these new technologies. The conjunction of archival finding aids technologies and these new data stores has not yet been explored before given the highly siloed nature of the library, archival, and information sciences (LAIS) field and limited development resources that focus on this area. However, there are many similarities between the problems that LAIS system administrators face maintaining descriptive documents and the reasons for why this new class of non-relational databases was developed.

See Also:
2015 - Spring

See Also:
The UBC School of Library, Archival and Information Studies Student Journal
2015 - Spring

An Exploration of Archival Finding Aids Technologies and NoSQL Databases

Stephanie Fan - stephanie.y.fan@gmail.com

Introduction

Since the 1970s, relational databases and their associated enterprise management systems have been the dominant players in the market. Hierarchical and object-oriented databases have been commercially available but never saw the widespread popularity that systems such as Oracle, MySQL, or MS SQL Server experienced (solid IT, 2015).Yet in recent years, much buzz has been attributed to the “NoSQL” data stores that were developed to address the issues identified through the emerging Web 2.0 data landscape – namely, the need for storage solutions to support higher availability and greater scalability. As a point of clarification, the class of NoSQL databases typically refers to the non-relational storage of the data rather than the lack of the SQL query language. However, this muddling terminology eventually led Atzeni et al. to clarify that, “[t]he debate on SQL vs. NoSQL is as much a debate on SQL, the language, as on the relational model and its various implementations” (Atzeni et al., 2013, p. 65).

Finding Aids

Finding aids are documents that provide descriptions and details about a collection from the most general level to the specific item level. These were created to provide archivists information regarding which items are in the collection and locations

where the collection is stored.

Two content standards define the list of required or optional elements and rules for description: the General International Standard Archival Description (ISAD(G)), used internationally (International Council on Archives & Committee on Descriptive Standards, 1999), and Rules for Archival Description (RAD), used and maintained by Canadian archivists (Canadian Committee on Archival Description, 2008). In both standards, decisions were made to support flexibility in the rules so that the pre-existing range of formatting and descriptions across organizations could be more easily accommodated.

The same year that Carl Strozzi coined the term NoSQL, the Encoded Archival Description Document Type Definition (EAD DTD) Version 1.0 was released in August 1998 to provide an encoding standard for machine-readable finding aids. This was to support and extend the existing machine-readable cataloguing (MARC) records in use by libraries and archives. An interesting and forward-thinking decision at the time of EAD DTD development, the project directors chose to make Version 1.0 compliant with the emerging Extensible Markup Language (XML) as there was potential for these digitally encoded finding aids to be natively displayable by the popular Web browsers of the time (Library of Congress, 2013).

Following the development of a machine-readable standard, library, archival, and information sciences (LAIS) adoption and subsequent development of related description standards began to occur. The EAD implementation process has been well-documented (Combs, Matienzo, Proffitt, & Spiro, 2010; Higgins, 1998; Yaco, 2008); however, there has been very little focus on end-to-end architecture and schema design for this area outside of available software selection and software testing. As Mitchell (2013) argued, “there is still a pressing need for a continued

fundamental restructuring of the metadata models and records that serve as the foundation for these systems."

Archival Information Systems

Two major open-source archival information systems are currently in use in North America.

First, Access to Memory (AtoM) was developed with the focus on the following key features: standards based design—either the ISAD(G) or RAD standards, support for hierarchical description, multilingual interface and database content, persistent permalinks to resources and standardized import and export for descriptions and authority records. The technical architecture is built with a mix of open-source libraries. As it was developed in 2006, it requires a relational database for storage (Van Garderen, 2009).

Second, ArchivesSpace is implemented with a derivative standard of the ISAD(G) guidelines. The product provides encoding tools for their core data model, in JSON, which primarily stores to a MySQL server (ArchivesSpace, 2015).

NoSQL Databases

Depending on the data model involved, there are many flavours of NoSQL solutions. For example, these include key-value pairs, column-family, graph, and document databases (Indrawan-Santiago, 2012). In the context of this paper focusing on finding aids, which are simple documents, the author is solely concerned with the sub-class of document databases, which store documents in JSON interchange format.

Specifically, the author chose to focus on MongoDB's implementation of non-relational concepts as one of the two open-source archival information systems, ArchivesSpace, already uses a JSON data model, a JSON data store is a logical choice. Moreover, MongoDB is currently one of the most popular NoSQL databases and most literature featuring performance metrics of NoSQL systems references this particular product (Abramova & Bernardino, 2013; Floratou et al., 2012; solid IT, 2015).

Findings

Given the usage of EAD XML as a front-end output, both of the aforementioned major archival information systems require a layer of processing to store this content into a relational database. This translation is done during both initial storage and subsequent retrieval. The usage of tried and tested relational database storage mechanisms does offer additional data integrity checks; however, there is much more processing overhead to map these non-tabular documents to and from the various tables.

Building upon this idea, the different hierarchical levels of finding aids may result in sparse data for the RAD or ISAD(G) attributes when translated to a relational structure. This is particularly true due to the nature of storing data as a matrix and the storage space needed to denote empty cell values. In comparison, a well-designed document database will store the original document in a format such as JSON or XML without the null value buffers. Overtime, these tiny gains may be significant to database administrators.

Finally, as digital collections grow, database administrators will need to control for physical storage expansion. For relational databases, this diverges into two options.

See Also:
2015 - Spring

Vertical scalability, where a larger server box is purchased to maintain the single-node server configuration, is costly. Conversely, horizontal scalability is the implementation of a multi-node distributed architecture configuration. The major enterprise relational database management systems have developed solutions to implement a multi-node framework. Nonetheless, due to the nature of relational database information access, this is technically difficult to build. At the cost of relational storage and complex transactions, NoSQL databases were developed to solve the horizontal scalability issue and thus have built-in functionality for replication. This can be used to distribute high user loads to concurrently running servers or to better support server failover.

Discussion

The question arises as to why NoSQL databases should be considered as a backend storage solution now. Regrettably, live data migration is difficult and expensive. While many of the organizations who have implemented one of the archival finding aids systems are not running into storage issues at this time, this will likely be a problem to address in the future. Thus, it makes sense to perform a complete environment scan early on to develop solutions as early as possible to reduce the monetary and technical costs. This will likely also decrease the complexity of the archival data stores, as there may be a closer 1:1 mapping between the EAD XML data model and NoSQL database as opposed to the hierarchical and relational mapping that currently exists.

It is interesting to note that during the original ArchivesSpace development meeting to merge Archon and Archivists' Toolkit in 2010, the question of whether NoSQL databases should be used was brought up. However, the ArchivesSpace technical project team was wary of such "bleeding edge technologies" and decided against this development for the initial version (ArchivesSpace, 2010). Using MongoDB as

an example, version 1.0 was released six years ago. Still, development and interest is very much active and healthy for this product. Many industry developers are very interested in this area; consequently, finding developer resources that are familiar with this area would not be as difficult as it once was. This is partially due to how quickly NoSQL databases are gaining ground in the IT industry with the shift towards agile development practices and the need to handle larger volumes of data.

Tangentially, are NoSQL databases mature enough to spend precious development resources on this area? There is much buzz in this area given the development of new storage solutions in a relatively stable field. This interest is also fueled by large-scale industry players who are using these solutions in interesting ways. However, it is important to note that these databases were developed for different intended use cases. Relational databases offer much more robust functionality that seems to be beyond the requirements for an archival finding aid storage system as archival finding aids are primarily use cases of small scale reads and writes to the system.

Lastly, while relational databases can handle transactional queries well, free-text analysis is quite difficult to perform with this structure given the join processing required. In order to better support the current systems, open-source search platforms are often appended to the application mix to perform this function. Yaco (2008) reported that within the EAD implementation literature, one study found that fifty-six percent of surveyed respondent organizations had digitally encoded finding aids but did not release them in a searchable format due to the difficulty in incorporating extended search functionality (Yaco, 2008). For document databases, while likely not as robust as dedicated platforms, there is inherent functionality to handle text analysis.

See Also:
2015 - Spring

Future Research

Much more work is needed to investigate the breadth of NoSQL databases and optimal data model design. As NoSQL data stores are very young, there are issues that remain to be addressed, such as the additional need for security or improvements to data consistency (Nance, Losser, Iype, & Harmon, 2013). Due to the short timeframe of this study, a prototype of a NoSQL backend was not developed for performance benchmarking purposes.

Yet, most importantly, there is a lack of attention paid to the usability of digitized archival finding aids in literature. Much of the literature focuses on the implementation of systems or the transition to newer description standards. Future studies should focus on how non-LAIS end users are actually using these systems as well as how search and retrieval research would inform this area. It was difficult to determine in the study timeframe whether existing systems are deployed with enough functionality to use the content naturally. In general, being able to work with non-LAIS resources may be helpful in determining how storage schemas should be designed in the future.

Conclusion

Archival finding aids are simple documents for a particular body of work arranged in a hierarchical structure from the most general level to the most specific item level. The author found that NoSQL databases address some of the bottlenecks with current archival information systems. While ArchivesSpace and AtoM are actively developing systems to ingest and manage these documents, the new class of NoSQL database solutions would offer many efficiencies to the current storage processes. Ultimately, the major issues that NoSQL databases attempts to solve are problems related to scalability from high throughput user access and storage

growth. Thus, the development of the sub-class of document databases will be an interesting area to continue to follow for technical administrators.

See Also:
2015 - Spring

References

Abramova, V., & Bernardino, J. (2013). NoSQL databases: MongoDB vs. Cassandra. In *Proceedings of the International C* Conference on Computer Science and Software Engineering* (pp. 14–22). ACM. Retrieved from http://dl.acm.org/citation.cfm?id=2494447

Abramova, V., Bernardino, J., & Furtado, P. (2014). Experimental evaluation of NoSQL databases. *International Journal of Database Management Systems*, *6*(3), 01–16. doi:10.5121/ijdms.2014.6301

ArchivesSpace. (2010, June). Original technical architecture report. Retrieved from http://www.archivesspace.org/sites/default/files/ArchivesSpace_Technical_Architecture_Report-FINAL-07302010.pdf

Atzeni, P., Jensen, C. S., Orsi, G., Ram, S., Tanca, L., & Torlone, R. (2013). The relational model is dead, SQL is dead, and I don't feel so good myself. *SIGMOD Rec.*, *42*(2), 64–68. doi:10.1145/2503792.2503808

Canadian Committee on Archival Description. (2008, July 8). Rules for Archival Description. Retrieved February 15, 2015, from http://www.cdncouncilarchives.ca/archdesrules.html

Cattell, R. (2011). Scalable SQL and NoSQL data stores. *SIGMOD Rec.*, *39*(4), 12–27. doi:10.1145/1978915.1978919

Combs, M., Matienzo, M. A., Proffitt, M., & Spiro, L. (2010). *Over, under, around, and through getting around barriers to EAD implementation*. Dublin, Ohio: OCLC Research. Retrieved from http://www.oclc.org/research/publications/library/2010/2010-04.pdf

Floratou, A., Teletia, N., DeWitt, D. J., Patel, J. M., & Zhang, D. (2012). Can the elephants handle the NoSQL onslaught? *Proceedings of the VLDB Endowment*, *5*(12), 1712–1723.

Higgins, R. (1998). A case study of EAD implementation at Durham University Library Archives and Special Collections. *Archives and Museum Informatics*, *12*(3-4), 221–234. doi:10.1023/A:1009011404871

Indrawan-Santiago, M. (2012). Database research: Are we at a crossroad? Reflection on NoSQL. In *2012 15th International Conference on Network-Based Information Systems (NBiS)* (pp. 45–51). doi:10.1109/NBiS.2012.95

International Council on Archives & Committee on Descriptive Standards. (1999). *ISAD(G): general international standard archival description : adopted by the Ad Hoc Commission on Descriptive Standards, Stockholm, Sweden, 19-22 September 1999.* Ottawa: [Committee on Descriptive Standards].

Library of Congress. (2013) <EAD> Encoded Archival Description version 2002 official site | Development of the Encoded Archival Description DTD. Retrieved February 10, 2015, from http://www.loc.gov/ead/eaddev.html

Mitchell, E. T. (2013). Chapter 1: Metadata developments in libraries and other cultural heritage institutions. *Library Technology Reports*, *49*(5), 5–10.

Nance, C., Losser, T., Iype, R., & Harmon, G. (2013). NoSQL vs RDBMS – why there is room for both. In *Proceedings of the Southern Association for Information Systems Conference* (pp. 111–116). Retrieved from http://aisel.aisnet.org/cgi/viewcontent.cgi?article=1026&context=sais2013

solid IT. (2015, February). DB-engines ranking - popularity ranking of database management systems. Retrieved February 18, 2015 from http://db-engines.com/en/ranking

Van Garderen, P. (2009). The ICA-AtoM project and technology. Retrieved from https://www.icaatom.org/download/VanGarderen_TheICA-AtoMProjectAndTechnology_AAB_RioDeJaniero_16-17March2009.pdf

Wikipedia. (2009). NoSQL. Retrieved December 27, 2014 from http://en.wikipedia.org/wiki/NoSQL

Yaco, S. (2008). It's complicated: Barriers to EAD implementation. *American Archivist*, *71*(2), 456–475.