Evaluating and Integrating Databases in the Area of NLP

Wahed Hemati, Alexander Mehler, Tolga Uslu,
Daniel Baumartz, Giuseppe Abrami

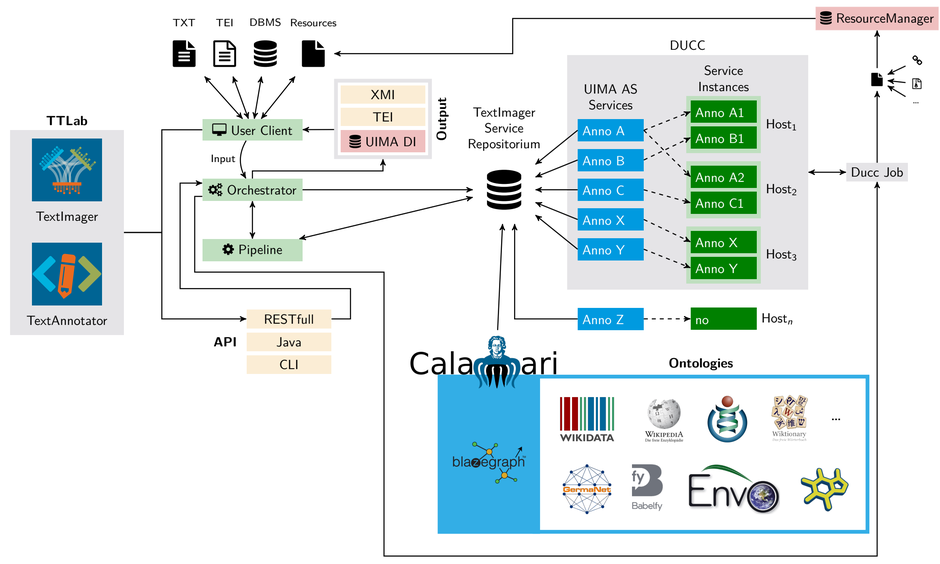
Goethe University Frankfurt | Text Technology Lab

QUALICO 2018
International Quantitative Linguistics Conference
5-8 July 2018
Wroclaw (Poland)

The Problem

- NLP is being used in more and more areas to perform (automatic) text analysis, e.g. medicine, biology, (digital) humanities, economy, legal theory, etc.
- Data for NLP tasks is often huge and diverse: graph-based models, document-based models (e.g. TEI), or combinations.
- We developed TextImager, a web service based NLP framework for big data (Hemati, Uslu, and Mehler 2016).
- NLP data is stored as XML-files (in our case serialized UIMA Common Analysis Structure UIMA-CAS) (Abrami and Mehler 2018).
- Further processing needs to always deserialize the thousands of XML-files, e.g. running quantitative analysis like: **TF-IDF** and **Type-Token-Ratio**.
- It is time consuming to: **add additional annotation layers**, or **query specific parts**.
- We propose a database system to be able to query annotations of documents according to the varying data models.
- The question: which existing database based on which paradigm performs best for which NLP tasks?
- Usually, databases are used for structured data and data for NLP is unstructured.
- We evaluate six different Database Management Systems (DBMS).
- Baseline: File System UIMA-CAS objects, serialized and compressed to XMI files (Grose, Doney, and Brodsky 2002).

TextImager

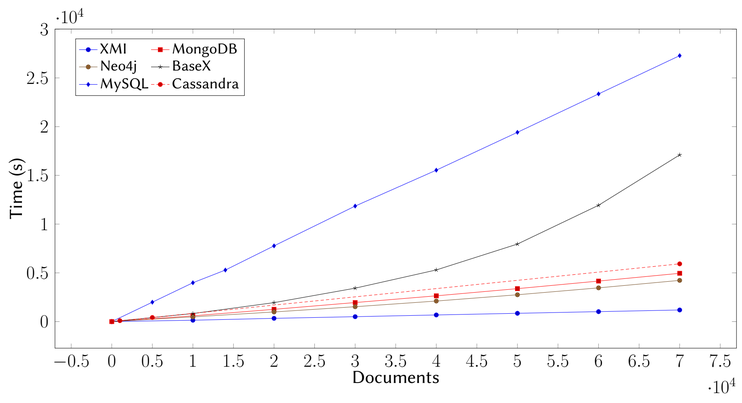


- Multi-User System
- Multi-Service System
- Multi-Pipeline System
- Multi-Server System
- Multi-Database System
- Multi-Visualization System
- Multi-Purpose System (Analysis, Synthesis, Depiction, etc.)

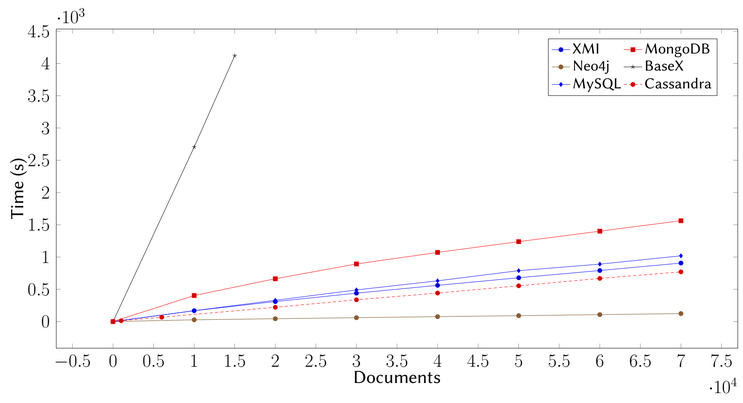
Databases

- MySQL: Open-source relational database management system, we use the NLP-related scheme from (Fette, Toepfer, and Puppe 2013).
- MongoDB: Scalable document-oriented NoSQL database, UIMA-CAS objects are serialized into binary-encoded JSON objects.
- BaseX: Light-weight XML document-oriented database, UIMA-CAS objects are serialized into XMI documents.
- Cassandra: Column-oriented NoSQL database, every annotation layer of the typesystem uses a separate table.
- Neo4j: Highly performant NoSQL graph database, each input text is represented as a node linking to all its token nodes whose syntagmatic order is mapped by token links.

Storage Performance



Reading Performance



Query Performance

	XMI	Mongo	Neo4j	BaseX	MySQL
POS	1,290.2	24.9	**13.8**	127.6	16.3
Lemma	1,287.0	27.5	57.0	185.9	**14.6**
Morph	1,299.1	26.3	48.0	132.3	**18.6**
Dep	1,490.2	371.4	**59.4**	2,349.7	87.7

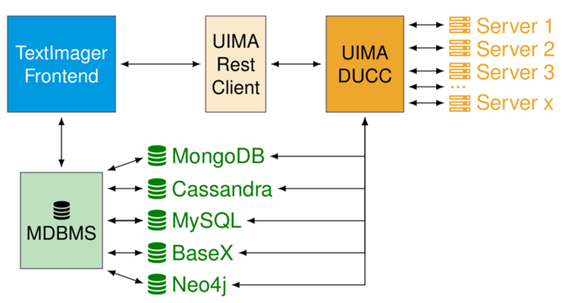
Data: 70,000 Wikipedia documents

Evaluation

- XMI: Extremely fast storing (just save the file), but also extremely slow querying.
- Neo4j: Fast storing, reading and relation querying.
- MySQL: Fast attribute querying and reading, but really slow storing.
- MongoDB: Consistently good performance, for all tasks.
- Cassandra: Performs ok, but not usable for our type of queries due to not supporting table joins for relations.
- BaseX: Relatively slow compared to the others.
- This shows no clear winner and depending on the task users should utilize different databases.
- Combination is needed: we introduce a web-based multi-database management system (MDBMS) integrated in TextImager.

MDBMS

- Provides a single interface for querying different databases.
- All data processed by TextImager is stored in multiple databases.
- Projects can simultaneously manage and query a variety of data using these databases:
  - Using the MDBMS interface using our combined databases, or
  - Directly accessing a single database, to use more specialized queries.



Abrami, G. and A. Mehler (2018). "A UIMA Database Interface for Managing NLP-related Text Annotations." In: *Proceedings of the 11th edition of the Language Resources and Evaluation Conference, May 7 - 12*. LREC 2018. Miyazaki, Japan.
Fette, G., M. Toepfer, and F. Puppe (2013). "Storing UIMA CASes in a relational database." In: *Proc. of UIMA@GSCL*, pp. 10–13.
Grose, T., G. Doney, and S. Brodsky (2002). *Mastering XMI: Java Programming with XMI, XML and UML*. Vol. 21. John Wiley & Sons.
Hemati, W., T. Uslu, and A. Mehler (2016). "TextImager: a Distributed UIMA-based System for NLP." In: *Proc. of the COLING 2016*.

Demo: https://textimager.hucompute.org/lab/database/

Github: https://github.com/texttechnologylab/textimager-database