Information Retrieval and Large Text Structured Corpora *

Fco. Mario Barcala[1], Miguel A. Molinero[2], and Eva Domínguez[1]

[1] Centro Ramón Piñeiro, Ctra. Santiago-Noia km. 3, A Barcia,
15896 Santiago de Compostela, Spain
barcala@freeresearch.org, edomin@cirp.es
[2] Dept. de Informática, Universidade de Vigo, Campus As Lagoas, s/n,
32004 Orense, Spain
molinero@uvigo.es

Conventional Information Retrieval Systems usually deal with plain text documents or with data files with a very elementary structure. Therefore, these kinds of system are able to solve queries in a very efficient way, but only taking into account whole documents, without distinguishing between different sections in those documents. This behaviour makes it impossible to support queries that should throw up a subset of documents which have some common characteristic or perform searches based on document sections.

In contrast with this, text corpora commonly employed nowadays are usually composed of a set of text files which in some way show a complex structure. So, building a classical Information Retrieval System to work with this kind of data will not benefit from this structure and results will not be improved.

In this article we test different technologies which can be used to build Information Retrieval Systems to be applied on large structured corpora, namely, technologies able to use the document structure to offer possibilities of filtering the search. In addition, and as a result of our work, we can propose the technology with the best flexibility-performance balance for use in building this kind of system.

There are several examples of Information Retrieval Systems using structured corpora [1] [2], and many others about searching methods on large text corpora [3], but there is no analysis or comparative study on technologies and architectures that can be used to build them. An overview of different generic technologies that could be applied to XML Information Retrieval Systems is shown in *XML Data Management, Native XML and XML-Enabled Database Systems* [4], but it is not about corpora.

In our analysis we suppose that the considered corpora are organized as a set of different XML (Extensible Markup Language) [5] files. This assumption does not imply an important restriction since every text corpora can be transformed to this format either automatically or manually, depending on the original structure of the text.

* Partially supported by Ministerio de Educación y Ciencia (MEC) and FEDER (TIN2004-07246-C02-01 and TIN2004-07246-C02-02), by MEC (HF2002-81), and by Xunta de Galicia (PGIDIT02PXIB30501PR, PGIDIT02SIN01E and PGIDIT03SIN30501PR)

Tamino [1] (a native XML indexer) [6], and Oracle[2] [7] (a relational database system which in its last versions integrates XML capabilities) are the main alternatives taken into account in our work. To perform our tests, we have taken into account the following two main criteria: query flexibility and query performance.

Query performance evaluation is based on processing time measures using a set of representative queries that covers a wide range of questions performed by typical text corpora users. To evaluate query flexibility we have considered the following aspects of the technologies under review:

- **Statistical capabilities**: The technology should allow the user to obtain numerical values at different levels. For example, capabilities to count the number of cases and documents which match each query.
- **Additional information**: It is also useful if the technology under review is able to offer supplementary information with the returned results. For example, returning sequences of terms which verify search criteria but also showing the author of relevant document, publisher name, etc., i.e., additional data taken from corpora files structure.
- **Match highlighting**: Technologies should be able to identify the term which caused the matching and highlight them among the retrieved results.
- **Kinds of search methods**: It is important to determine if a particular technology suppports only exact matches or it allows other kinds of matching (matching and indexing that is sensitive and insensitive to accents, boolean search, complete or partial matching, etc.).
- **Use of wildcards**: Support of advanced features such as wildcards, character substitution, suppport for regular expresions, etc.
- **Results browsing and navigation**: The technology should also allow us to go through the retrieved results, showing them page by page or even using more sophisticated ways.
- **Ordering**: Ordering capabilities offered by the technology are also relevant. Systems can allow single ordering criteria, several simultaneous orderings or impose other limitations.
- **Structural relationships**: This aspect refers to query flexibility when structural restrictions are included in queries. This covers questions like capabilities offered by the query language, and query complexity limitations.

The results obtained show that although XML native managers are a serious candidate for the near future, they present certain deficiencies, which limit query flexibility if we want to achieve a good performance. Only high performance relational database systems like Oracle can satisfy some kind of requirements, relevant to this kind of system.

[1] Tamino is a Software AG product.
[2] Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

References

1. María Sol López Martínez CORGA (Corpus de Referencia del Gallego Actual) In *Proc. of Hizkuntza-corpusak. Oraina eta feroa*, pages 500–504, Borovets, Bulgaria, Sept. 2003.
2. Mark Davies Un corpus anotado de 100.000.000 palabras del español histórico y moderno. In *Proc. of Sociedad Española para el Procesamiento del Lenguaje Natural* pages 21–27, Valladolid, Spain, 2002.
3. Mark Davies Relational n-gram databases as a basis for unlimited annotation on large corpora. In *Proceedings from the Workshop on Shallow Processing of Large Corpora*, Lancaster, England, pages 23–33, Lancaster, England, March 2003.
4. Akmal B. Chaudhri, Awais Rashid and Roberto Zicari *XML Data Management, Native XML and XML-Enabled Database Systems*, Addison-Wesley, March 2003.
5. XML In *http://www.w3c.org*, 27/10/2004
6. Tamino In *http://www.softwareag.com*, 27/10/2004
7. Oracle In *http://www.oracle.com*, 27/10/2004