Performance of XML Databases for Epidemiological Queries in Archetype-Based EHRs

Sergio Miranda Freire[a,b], Erik Sundvall[a], Daniel Karlsson[a], Patrick Lambrix[c]

[a]*Department of Biomedical Engineering, Linköping University, Linköping, Sweden*
[b]*Departamento de Tecnologia da Informação e Educação em Saúde, Universidade do Estado do Rio de Janeiro, Brazil*
[c]*Department of Computer and Information Science, Linköping University, Linköping, Sweden*

Abstract

There are very few published studies regarding the performance of persistence mechanisms for systems that use the openEHR multi level modelling approach. This paper addresses the performance and size of XML databases that store openEHR compliant documents. Database size and response times to epidemiological queries are described. An anonymized relational epidemiology database and associated epidemiological queries were used to generate openEHR XML documents that were stored and queried in four open-source XML databases. The XML databases were considerably slower and required much more space than the relational database. For population-wide epidemiological queries the response times scaled in order of magnitude at the same rate as the number of records (total database size) but were orders of magnitude slower than the original relational database. For individual focused clinical queries where patient ID was specified the response times were acceptable. This study suggests that the tested XML database configurations without further optimizations are not suitable as persistence mechanisms for openEHR-based systems in production if population-wide ad hoc querying is needed.

Keywords: Medical Record Systems, Computerized; Database Management Systems, Archetypes, XML Databases, openEHR

Introduction

An electronic health record (EHR) is a computer processable repository of information regarding the health status of a subject of care [1]. Much has been published about the potential use of EHRs to support healthcare, clinical-epidemiological studies, decision support systems and healthcare services management. International Standards have been proposed to establish the EHR definition, context and scope [1], the requirements of the EHR architecture [2] and models to communicate EHR extracts [3]. Despite all this, electronic health records are usually non interoperable, hard to evolve and do not fully meet the proposed requirements.

Health care is an area with some features that make it very complex for the development of EHR systems. For instance: there is a large number of evolving concepts and it is hard to achieve a consensus regarding comprehensive models for the EHR. To reduce the need for constant changes in the system persistence models, [4–6] propose the separation between the domain model and the reference model.

This separation of responsibilities is refined in the specifications developed by the openEHR Foundation [7], a multi level modelling approach, designed in order to build future-proof systems. The approach uses a stable reference model (RM) that can be implemented in software, and a flexible domain model expressed in "archetypes" and "templates"; these concepts are well explained in [8]. The RM is the model whose classes will be persisted and tends to be stable, i.e., its classes are intended not to change frequently. The archetypes give the semantic meaning to the objects that are persisted via reference model. OpenEHR's proposal is that structural changes and business rules are reflected in the archetypes rather than in the RM; this way there is no need to make changes in the persistence mechanism, be it relational, object-oriented, XML, etc. Furthermore the archetypes are created and edited primarily by domain experts, not programmers or informaticians.

The archetype-based multi level approach has opened a new horizon for research in medical informatics, besides inspiring standards development organizations [3] Several research groups have been working on several issues raised by the approach, for instance, representation of clinical guidelines [9], conversion of data stored in legacy systems to archetype-based systems [10,11], implementation of the specifications as open source [12], among others.

An important decision to be taken when developing systems based on the multi level modelling approach is the choice of persistence mechanism, so that performance and query requirements are met. Since the RM has a large set of classes that can form relatively deep hierarchies, a pure object-relational mapping may not be an efficient solution, this is suggested by the literature and discussions in the openEHR community [13,14]. Some openEHR-based open-source implementations have been made public recently [15–17], but their performances using realistic epidemiological data and queries have not been described.

The EHR data, generated according to the RM, can be serialized in several formats: JSON, XML, and others. Since there are several XML databases available, they can be used to store openEHR compliant XML documents. But in

order to be used in production, they must have good performance not only when querying for data about an individual (clinical query) but also for data about a whole population (epidemiological query), e.g., follow up or research. This is one of the most important secondary uses of electronic healthcare records and to the best of our knowledge, no study has been published with this kind of evaluation.

This paper addresses the performance of XML databases that stores openEHR compliant documents in terms of size and response times to epidemiological queries.

Materials and Methods

Test database

This study used the database of the National Cervical Cancer Information System – SISCOLO – for the State of Rio de Janeiro, Brazil, from June 2006 to December 2009 that was subjected to a process of record linkage in order to identify records that belong to the same patient [18]. Through this process, all records belonging to the same patient were given the same integer identifier (uid field in the database) that was unique for each patient. Data for this database are collected from standardised forms for two ambulatory procedures requests: cervical pathological examination and pap smear. These forms generate two main tables in the SISCOLO database which contain respectively the results of histological and cytological examinations of women. Those tables, which from now on will be called "histology" and "cytology", were exported into corresponding tables in a MySQL [19] schema.

The cytological exam comprises the following sections:

- anamnesis;
- clinical examination
- cytological exam results
  - reasons for the rejection of the slice
  - type of epithelium in the sample
  - material adequability (true or false)
  - reasons for inadequability
  - benign cellular alterations
  - microbiology
  - atypical cells of indeterminate meaning
  - atypias in scamous cells
  - atypias in glandular cells
  - other malign neoplasias
  - presence of endometrial cells

Depending on the results of the cytological exam, the woman may be referred to perform a histological exam which is comprised of the following sections:

- cytological exam results;
  - atypical cells of indeterminate meaning
  - atypias in scamous cells
  - atypias in glandular cells
- colposcopy
  - result
  - procedure
- type of surgical procedure
- macroscopy
- microscopy
  - benign lesions
  - neoplasic or pre-neoplasic lesions
  - differentiation degree
  - tumour extension
  - surgical margins

The histology table has 7,477 records belonging to 6,238 patients. The cytology table has 2,471,088 records, belonging to 1,679,801 patients, and 5,316 of them have also records in the histology table. All identifying demographic data of patients, health professionals and organizations were removed from the database. In addition to this, the date of birth and the date of exam were modified by adding or subtracting a random number of days (in the intervals ±912 and ±100 respectively). This was done in order to obtain a virtually anonymized set of data without impacting the representativeness of the set.

OpenEHR XML documents

A set of 10 archetypes and 3 templates was designed from scratch in order to represent the contents of the SISCOLO database schema using the Ocean Informatics Archetype Editor and Template Designer [20,21]. Then the contents of the SISCOLO database was mapped to openEHR XML documents, according to the following steps:

1. XML files (EHR data instance examples) corresponding to each of the compositions: histological exam, cytological exam and administrative data were generated using LiU-EEE [22] that exposes the example instance skeleton generator from the openEHR Java reference implementation [12].
2. An XML document that represented a patient record with those compositions was generated using LiU-EEE.
3. This XML document was used as a basis for a Freemarker [23] template in order to map all records in the SISCOLO database to openehr XML documents. One XML document was created for each patient and it contained all histological and cytological exams for that patient.

Of the 6,238 women who have histological exams, 5,281 performed only one exam, 779 performed two exams, 138 performed 3 exams and 40 performed 4 or more exams. Of the 1,679,801 women who were submitted to cytological exams, 1,135,726 had one exam, 363,357 had 2 exams, 132,438 had 3 exams and 48,280 had 4 or more exams. Each exam generated one composition in the EHR. It can be seen then that the majority of women have only a few exams. In the very few cases where women have more than ten exams of each type, this may be due to errors in the probabilistic record linkage process.

For women who have histological exams, the openEHR XML documents vary in size from a minimum of 30 KBytes to a maximum of 606,2 KBytes, depending on the number of cytological and histological exams they were submitted to.

Evaluated Databases

Four XML database systems were evaluated in this study and compared with the performance of the original database in

MySQL (version 5.5.24): eXist (version 1.4.2) [24], BaseX (version 7.3) [25], Sedna (version 3.5) [26], and Berkeley DB XML (version 11g) [27]. This selection included the major actively maintained open source XML databases with XQJ and XQuery java interfaces.

Evaluation setup

The evaluation was done in terms of storage space and response times to a series of queries against each of the databases. Three datasets were built as subsets of the original SISCOLO database:

1. all EHRs containing both histological data and associated cytological data (6,238 records) - *siscolo6k*;
2. the same records as in 1 plus around 60,000 EHRs containing only cytological data (66,070 records) – siscolo60k;
3. the same records as in 1 plus around 600,000 EHRs containing only cytological data (604,367 records) - siscolo600k.

Each of these datasets were stored in each of the XML database systems. Then a set of population-based queries were created in SQL and equivalent ones in the Archetype Query Language (AQL) [29], which were translated to XQuery [30] through the LiU-EEE software.

In order to be realistic, the population-based queries were created following the analysis performed in an epidemiological study that evaluated the effectiveness of the SISCOLO screening programme [30]. In addition to the population-wide queries, three clinical queries were created to access, for each of a set of randomly selected EHRs, the whole content of the EHR, one *composition* section and one *evaluation* section within each EHR.

AQL does not yet have aggregation functions. Therefore all population-based queries were framed to return all record Ids that satisfied the query criteria. All epidemiological queries included a time interval in their selection criteria. The response times were evaluated both for an interval of four months and for a period of three years which included most of the data in the database. Box 1 shows an example of an AQL query and its corresponding translation to XQuery.

Box 1: A query example is to return all record ids that had a histological exam result indicating neoplastic lesions between 2006-01-01 and 2006-05-01.

In AQL it is expressed as...

```
SELECT e/ehr_id/value as ehr_id
FROM Ehr e
CONTAINS VERSION v
CONTAINS COMPOSITION c [openEHR-EHR-COMPOSITION.histologic_exam.v1]
CONTAINS OBSERVATION obs [openEHR-EHR-OBSERVATION.histological_exam_result.v1]
WHERE (EXISTS obs/data[at0001]/events[at0002]/data[at0003]/items[at0085]/items[at0033]/items[at0034] OR
EXISTS obs/data[at0001]/events[at0002]/data[at0003]/items[at0085]/items[at0033]/items[at0035])
AND c/context/start_time/value >= '2006-01-01T00:00:00,000+01:00'
AND c/context/start_time/value < '2006-05-01T00:00:00,000+01:00'
```

...which when translated to XQuery results in:

```
declare namespace v1 = "http://schemas.openehr.org/v1";
declare default element namespace "http://schemas.openehr.org/v1";
declare namespace xsi = "http://www.w3.org/2001/XMLSchema-instance";
declare namespace eee = "http://www.imt.liu.se/mi/ehr/2010/EEE-v1.xsd";
declare namespace res = "http://www.imt.liu.se/mi/ehr/2010/xml-result-v1#";
<res:xml-results>
<res:head><res:variable name="ehr_id"/></res:head>
<res:results>
{let $ehrRoot := //eee:EHR
for $e in $ehrRoot
for $v in $e/eee:versioned_objects/eee:versions
for $c in $v//*[@xsi:type='v1:COMPOSITION' and @archetype_node_id="openEHR-EHR-COMPOSITION.histologic_exam.v1"]
for $obs in $c//*[@xsi:type='v1:OBSERVATION' and @archetype_node_id=
 "openEHR-EHR-OBSERVATION.histological_exam_result.v1"]
where
(exists($obs/data[@archetype_node_id = 'at0001']/events[@archetype_node_id = 'at0002']/data[@archetype_node_id =
'at0003']/items[@archetype_node_id = 'at0085']/items[@archetype_node_id = 'at0033']/items[@archetype_node_id = 'at0034'])
or exists($obs/data[@archetype_node_id = 'at0001']/events[@archetype_node_id = 'at0002']/data[@archetype_node_id =
'at0003']/items[@archetype_node_id = 'at0085']/items[@archetype_node_id = 'at0033']/items[@archetype_node_id = 'at0035'])) and
$c/context/start_time/value  >= '2006-01-01T00:00:00,000+01:00' and $c/context/start_time/value < '2006-05-
01T00:00:00,000+01:00'
return
<res:result><res:binding name="ehr_id">{$e/eee:ehr_id/value}</res:binding></res:result>}
</res:results>
</res:xml-results>
```

The queries were executed ten times in MySQL in all three datasets. Due to their slow response times, the number of times the queries were executed in the XML databases was reduced. For BaseX, they were run three times for siscolo6k, two times in siscolo60k and siscolo600k. For Sedna, they were run two times for siscolo6k, a subset of the queries were run two times in siscolo60k and only once in siscolo600k. For Berkeley DB XML, the same subset of the queries was run three times in siscolo6k and only once in siscolo60k and siscolo600k. All databases were queried through the Java API for each database and the execution time was measured from the moment the query was sent to the database until the result set was returned. No display of or navigation through the results was performed.

The MySQL database was indexed by the EHR id. No indexes besides those that are already built-in in the XML databases were created, because we were most interested in ad hoc queries for which it is not known in advance which indexes should be used, and which is a very common use case in health care research. Thus, in the XML databases, no assumptions were made about the kinds of query that would be made. The evaluation was performed with a single user accessing the stand-alone database.

The evaluation was performed in a DELL desktop with an AMD Athlon 64X2 dual-core processor 5600+x2 with 3.9 GB RAM running Ubuntu 12.04 LTS.

Results

The sizes of each database in MySQL and in the four XML databases are shown in Table 1 for the three datasets described above and also for the complete dataset, described by "FULL" in the table. Besides requiring much more space than the relational database, it can be observed that there is a large discrepancy among the XML databases, with BaseX being the least demanding and Sedna the most space consuming.

Table 1 – Databases' size in GBytes according to the database system and the number of records it contains.

Database	Size (Number of Records)			
	6,238	**66,070**	**604,367**	**FULL**
MySQL	0.013	0.052	0.41	1.1
Generated XML files	0.56	2.8	23	60
BaseX	0.38	1.9	15.7	42
eXist	0.72	3.9	31.27	70
Sedna	1.7	8.4	70.9	181
Berkeley DB XML	1.5	7.0	57.4	137

The response times for each of the databases are shown in Figure 1.

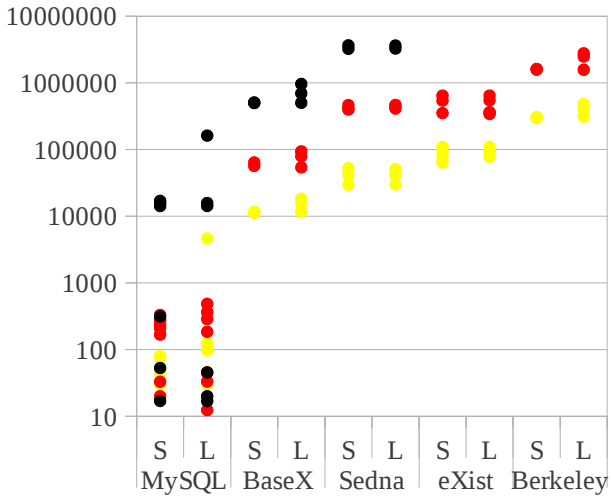

Figure 1 - Response Times for the epidemiological queries.
Yellow: siscolo6k, Red: siscolo60k,
Black:- siscolo600k,
S, L - short and long time intervals respectively
Y axis - Time in milliseconds and logarithmic scale

All XML database systems perform poorly compared to the relational database for the epidemiological queries and there is also a large discrepancy among them as far as the response time goes. MySQL has response times varying from µs to hundreds of ms with one query reaching a peak of 100s one time for the larger dataset. BaseX has the best performance of all XML databases in all datasets but the response times are around 10s for the 6k , 60s for 60k and 500s for 600k dataset for the short time interval. The response times increase significantly for the long time interval. Sedna comes second in the rank of XML databases with similar response times for both the short and long time intervals, but much slower response times than BaseX in all datasets. eXist is slower than Sedna for the smallest dataset, but with comparable response times for the 60k dataset. Berkeley DB XML was the slowest of all with response times two orders of magnitude higher than BaseX. No data was obtained for eXist and Berkeley DB XML for the 600k dataset because the response was taking too long and the program was aborted.

The average response times for the clinical queries were between 10 and 200ms for the BaseX, eXist and Berkely DB XML, as well as for MySQL. The response times depended on the sizes of the individual records.

Discussion

The openEHR XML documents are very verbose; this is caused by the inherent verbosity of XML and by the openEHR RM. The RM has a deep tree structure and it stores both codes and description for terminological entries. More information is also added to the openEHR data such as context, auditing, archetype ids an so on, which was not present in the anonymized SISCOLO database. The size of

the three sets of XML documents are respectively 556 MBytes, 2.8 GBytes and 23 GBytes. Therefore it is not a surprise that the sizes of the XML databases are much larger than the corresponding SQL database. However it is interesting to notice that the XML database systems differ greatly in the size of the generated databases with BaseX being the most space saving of all and Sedna and Berkeley DB XML requiring around 3 times more space than BaseX. eXist ranks second in this aspect.

The response times of the XML databases for the epidemiological queries leave much to be desired as compared to the sql database. This is in accordance with the results from the literature [31]. There are also large differences among the XML databases, being BaseX again the most responsive of all and eXist the least responsive. The response times are very high, even for the smallest dataset. In a realistic scenario with concurrent access to the databases, the response times would be even worse.

Each XML database has its own built-in indexing mechanism. With Sedna the query syntax should be modified to indicate which index to use. In the epidemiological scenario, this is not useful because usually it is not easily known in advance which index would be most helpful to the query execution.

The way the openEHR archetypes are designed and the nature of data values that are stored in the database make the automatically generated indexes in the databases inefficient. The archetypes usually have many attributes with the same value, for instance almost all archetypes have an archetype node id equal to “at0001” and the database used in this study has mainly coded values with few options to choose from. This makes xml text and attribute indexes point to a huge number of entries in the database, leading to long inspection of documents in order to return the results. How to best handle querying of the relatively deep openEHR tree structures, often with repeated path segment identifiers, is an interesting topic for future research.

The XQueries were not handwritten but produced by LiU-EEE AQL parser. Possibly these queries could be rewritten so that better response times could be obtained, but this is open to investigation.

It would be better to run each of the queries the same number of times for each scenario but, due to the slow response times, the number of repetitions was reduced and, in some cases, some queries were omitted. However, it has been observed that the response times do not change very much for successive runs of the same queries and the order of magnitude of the response times is not lost when we limit the number of repetitions. Besides the queries that were left out had response times similar to the queries that remained.

Although useful as an educational tool for teaching the openEHR specification and implementation to students and newcomers, the results of this study put into question the usability of XML databases as a persistence mechanism for openEHR-based systems intended for ad hoc population queries. Even in the clinical scenario, it is common to perform population-based queries, for instance, a doctor that asks for the records of all patients that he/she is going to attend that day. Therefore other alternatives should be investigated, such as the one proposed by Beale [32], Arikan [15] XML shredding [33] or other architectures such as Column stores, e.g. Hadoop/Hbase (hadoop.apache.org), Cassandra (cassandra.apache.org); Document store, e.g. Terrastore (code.google.com/p/terrastore); Key Value or Tuple stores, e.g., AmazonSimpleDB (aws.amazon.com/simpledb), Graph Databases, e.g., Neo4j (neo4j.org), InfoGRID (infogrid.org); and RDF triple stores e.g. Allegro Graph (www.franz.com/agraph/ allegrograph).

It is important to point out that the Archetype Query Language does not provide constructors for creating aggregation queries. In the clinical-epidemiological context this is an essential requirement and the Archetype Query Language needs to be enhanced in order to enable interoperable such queries. It is also necessary to add the “Distinct” construct so as to avoid the return of the same values more than once in the query results.

Conclusions

The XML database systems were considerably slower and required much more space than the relational database. For population wide epidemiological queries the response times scaled in order of magnitude at the same rate as the number of records (total database size) but were orders of magnitude slower than the original relational database. For individual focused clinical queries where patient ID was specified the response times were acceptable. This study suggests that the tested XML database configurations without further optimizations are not suitable as persistence mechanism for openEHR-based systems in production if population-wide ad hoc querying is needed.

Acknowledgements

SM Freire received a scholarship from CAPES Foundation (Proc. 4055/11-0). The authors thank the Head of the Cervical Cancer Screening Program of the Health Secretary of Rio de Janeiro State for allowing access to the original SISCOLO database .

References

[1] ISO. TR 20514 - Health informatics - Electronic health record - Definition, scope and context [Internet]. International Organization for Standardization; 2005 p. 27. Available at: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=39525

[2] ISO. IS 18308 - Health informatics -- Requirements for an electronic health record architecture [Internet]. International Organization for Standardization; 2011 p. 25. Available at: http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=52823

[3] ISO. IS 13606: Health informatics — Electronic healthcare record communication — Part 1: Reference Model [Internet]. International Organization for Standardization; 2008 p. 83. Available at:

http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=40784

[4] Johnson J. Generic Data Modeling for Clinical Repositories. J Am Med Inform Assoc. 1996;3:328-39.

[5] Nadkarni P, Marenco L, Chen R, Skoufos E, Shepherd G, Miller P. Organization of Heterogeneous Scientific Data Using the EAV/CR Representation. J Am Med Inform Assoc. 1999;6:478-93.

[6] Chen R, Enberg G, Klein G. Julius - a template based supplementary electronic health record system. BMC Med Inform Decis Mak. 7(10).

[7] openEHR. The openEHR Foundation [Internet]. [Accessed 2012 jul 28]. Available at: http://www.openehr.org

[8] Beale T, Heard S. OpenEHR architecture overview [Internet]. [Accessed 2012 jul 20]. Available at: http://www.openehr.org/releases/1.0.2/architecture/overview.pdf.

[9] Chen R, GeorgiI-Hemming P, Ahlfeldt H. Representing a Chemotherapy Guideline Using openEHR and Rules. Stud Health Technol Inform. 2009;150:653-7.

[10] Moner D, Maldonado J, Boscá D, Fernandez J, Angulo C, Crespo P, et al. Archetype-Based Semantic Integration and Standardization of Clinical Data. Proceedings of the 28th IEEE EMBS Annual International Conference. New York; 2006. p. 514-5144.

[11] Chen R, Klein G, Sundvall E, Karlsson D, Ahlfeldt H. Archetype-based conversion of EHR content models: pilot experience with a regional EHR system. BMC Med Inform Decis Mak. 2009;9(33).

[12] Chen R, Klein G. The openEHR Java Reference Implementation Project. MEDINFO 2007. Kuhn et al.; 2007. p. 58-62.

[13] Muñoz A, Solominos R, Pascual M, Fragua J, Gonzalez M, Monteagudo J, et al. Proof-of-concept Design and Development of an EN13606-based Electronic Health Care Record Service. J Am Med Inform Assoc. 2007;14(1):118-129.

[14] openEHR Foundation. openEHR Technical Discussion List [Internet]. [Accessed 2012 jul 27]. Available at: http://lists.openehr.org/pipermail/openehr-technical_lists.openehr.org/

[15] Arikan S. openEHR REFerence Framework and Application [Internet]. [Accessed 2011 may 4]. Available at: http://opereffa.chime.ucl.ac.uk/introduction.jsf

[16] Atalag K, Yang H. From openEHR Domain Models to Advanced User Interfaces: A Case Study in Endoscopy. 2010 Health Informatics New Zealand Conference [Internet]. Wellington; 2010 [Accessed 2011 may 5]. Available at: http://www.openehr.org/wiki/download/attachments/18513934/Atalag_HINZ2010-Paper.pdf?version=1&modificationDate=1291667587000

[17] Pazos P, Carrasco L, Machado F, Simini F. Traumagen: historia clínica electrónica con acceso a estudios radiológicos digitales especializada en la atención de pacientes gravemente traumatizados. CAIS - JAIIO 2010 [Internet]. 2010 [Accessed 2009 nov 10]. Available at: http://www.slideshare.net/pablitox/proyecto-traumagen-cais-jaiio-2010

[18] Freire SM, Almeida RT de, Bastos E de A, Cabral MDB, Souza RC, Silva MGP. A record linkage process of a cervical cancer screening database. Computer Methods and Programs in Biomedicine. 2012; 108:90–101 .

[19] MySQL database [Internet]. Oracle Corporation; [Accessed 2010 jun 1]. Available at: http://www.mysql.com

[20] Ocean Archetype Editor [Internet]. Ocean Informatics; [Accessed 2012 jul 28]. Available at: http://www.openehr.org/svn/knowledge_tools_dotnet/TRUNK/ArchetypeEditor/Help/index.html

[21] Ocean Template Designer [Internet]. Ocean Informatics; [Accessed 2012 jul 28]. Available at: http://wiki.oceaninformatics.com/confluence/display/TTL/Template+Designer+Releases

[22] Sundvall E, Nyström M, Karlsson D, Eneling M, Chen R, Örman H. Applying Representational State Transfer (REST) Architecture to Archetype-based Electronic Health Record Systems. Unpublished Manuscript; 2012.

[23] Freemarker - Java Template Engine Library [Internet]. [Accessed 2012 jul 28]. Available at: http://freemarker.sourceforge.net/

[24] http://exist-db.org/exist/index.xml [Internet]. [Accessed 2012 jul 28]. Available at: http://exist-db.org/exist/index.xml

[25] BaseX:The XML database [Internet]. [Accessed 2012 jul 28]. Available at: http://basex.org/

[26] Sedna: Native XML Database System [Internet]. [Accessed 2012 jul 28]. Available at: http://www.sedna.org/

[27] Oracle Berkeley DB XML [Internet]. Available at: http://www.oracle.com/technetwork/products/berkeleydb/index-083851.html

[28] openEHR Foundation. Archetype Query Language [Internet]. [Accessed 2012 jul 28]. Available at: http://www.openehr.org/wiki/display/spec/Archetype+Query+Language+Description

[29] W3C. XQuery 1.0 [Internet]. 2010 [Accessed 2012 jul 28]. Available at: http://www.w3.org/TR/xquery/

[30] Bastos E de A. Estimativa da Efetividade do Programa de Rastreamento do Câncer do Colo do Útero no Estado do Rio de Janeiro [M. Sc. Thesis]. [Rio de Janeiro]: Universidade Federal do Rio de Janeiro; 2011.

[31] Green J. A Comparison of the Relative Performance of XML and SQL Databases in the Context of the Grid-SAFE Project. University of Edinburgh; 2008.

[32] Beale T. Node + Path persistence [Internet]. 2008 [Accessed 2012 jul 28]. Available at: http://www.openehr.org/wiki/pages/viewpage.action?pageId=786487

[33] Strömbäck L, Freire J. XML Management for Bioinformatics Applications. Computing in Science & Engineering. 2011;13(5):12-22.

Address for correspondence

Sergio Miranda Freire

Department of Biomedical Engineering

Linköping University

SE-581 83 - Linköping – Sweden

e-mail: sergio@lampada.uerj.br