A caching approach for XML based medical data

Athena I. Vakali - Andreas S. Pombortsis

Department of Informatics

Aristotle University

54006 Thessaloniki, Greece

avakali,apombo@csd.auth.gr

Abstract

This paper discusses the representation and caching of medical data structured and stored as XML documents. Caching of XML medical data is proposed towards improving the medical data accesibility and availability. The representation of XML medical data is a tree like structure and the caching process is guided by frequency of access of the XML medical data records. Experimentation is carried out for an artificial workload of patient records and both cache and byte hit ratios are evaluated. The proposed caching scheme is proven to be quite effective and beneficial for accessing the medical data.

Index terms: *medical data management, XML medical data representation, caching policies.*

1 Introduction

XML (e*X*tensible *M*arkup *L*anguage) has recently emerged as a new innovative standard for data representation and data transfer over the Internet. XML has been proposed as an innovative mechanism for medical data representation and management [4, 5]. XML is used in medical data representation due to its ability to define markups for specific types of data, thus it can be utilized as a markup for medical data. Here, we concetrate on medical data which are represented by XML markups stored as XML documents among various storage units under a common storage subsystem.

The idea of caching has been proposed in the past in relation to file system and operating systems. Research efforts have considered caching as a policy for reducing traffic and congestion to the storage medium. By introducing the caching policy we consider a cache area reserved on a primary storage level for faster access and request servicing. *File prefetching and caching* has been proven a quite effective technique for improving file access performance. Traditional

caching in a distributed file system is discussed in [2] and the most critical research issues in caching concern cache replacement strategies as well as cache consistency and validation. A web-based evolutionary model has been presented in [9] where cache content is updated by evolving over a number of successive cache objects populations.

This paper presents a model for caching medical data represented as XML documents. We propose a caching scheme for medical data stored as XML documents according to their popularity as identified by their frequency of access. The experimentation considered patient records as described in [7] for the medical data workload. The patients records have been chosen due to their organized structure which is the basis for a principle markup and advanced computer organization and processing. The remainder of the paper is organized as follows. The next section defines the medical XML data representation structure. In Section 3 the caching policies are presented whereas Section 4 has the experimetation details and the results for the proposed caching scheme ratios. Finally conclusions and future work topics are given in Section 5.

2 XML Medical Data Representation

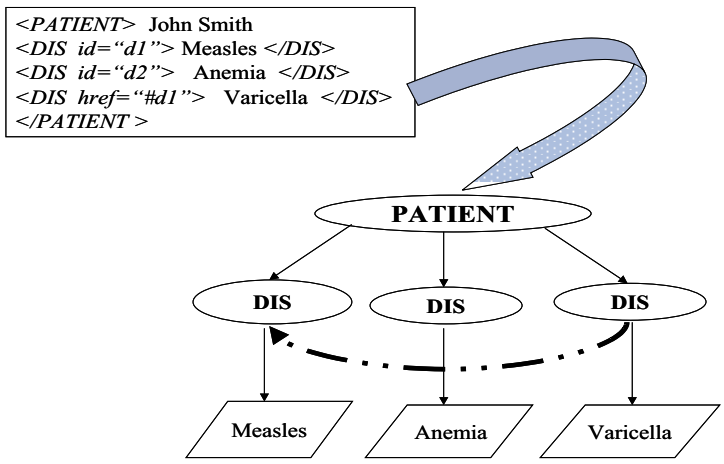

Figure 1: A tree structure for an XML medical data record.

The data model for our XML based data is considered as a linearization of a tree structure [3, 6]. At every node in the tree we might have links to several character strings or we might have a cross reference to another node of the XML tree like structure. The tree structure and the character strings together form the information content of an XML medical data document. Figure 1 presents an example of a fragment of a particular XML structured medical information record. Here, we have considered a patient's record for his reported diseases and their relations. This tree-like representation is a logical view of the XML medical document it is stored by a

corresponding physical structure. The physical structure is the considerd XML file identified as an XML medical document or object.

Each node might be requested and accessed by various users (clients) and we have both "popular" and "non-popular" medical information nodes (XML documents). Thus, there is a relative frequency estimated by the user requests and we we need to define the relative frequency of access of each XML medical object.

Definition 1 : The cached object's *relative frequency* is defined by

$$DynFreq_i = \frac{1}{a_i}$$

since a_i is the metric for estimating an XML medical object's access frequency (Table **??**). It is true that the higher the values of $DynFreq_i$, the most recently was accessed, since a_i is the parameter to identify the number of accesses to other objects since object i was last referenced.

3 The Caching Approach

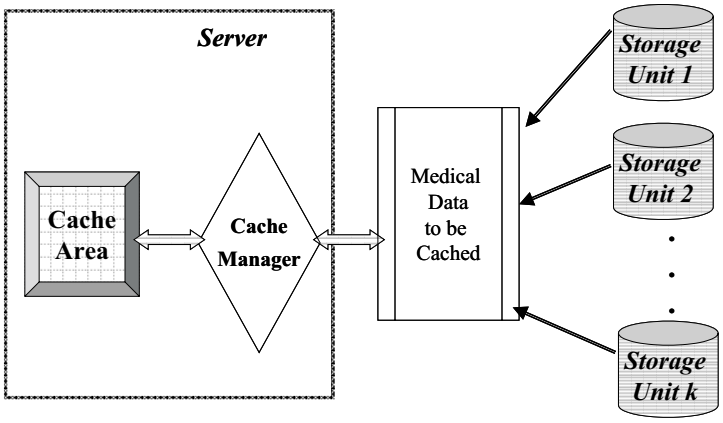

Figure 2: The XML medical data caching model.

As mentioned earlier, we introduce the caching process for a considered medical data set represented by a corresponding number of XML documents or objects. Thus our problem statement is the answer to the following question :

"How to propose an effective caching of the XML medical objects in a certain cache area in order to improve access to the requested XML based medical data ?"

Figure 2 presents our model's architecture. We assume that the XML medical data might reside among various storage units which are managed by a common storage subsystem. We also consider a cache server which maintains a certain cache area of high access technology for the storage of the most frequently requested

Our caching policy could be identified by the following steps :

```
sort XML documents according to their Frequency of access
FT <-- define a Frequency Threshold
select <-- set of XML documents of greater than FT value
cache <-- place in cache the XML documents from select
          with the highest possible Frequency of access
          such that the cache is filled
```

With this caching algorithm there is a number of XML medical documents which will reside in cache according to their frequency of access, i.e the most popular medical data records will be stored in the reserved cache area. Thus, the next request for a popular medical XML document will be most likely serviced by the cache area at a faster rate.

4 Experimentation - Results

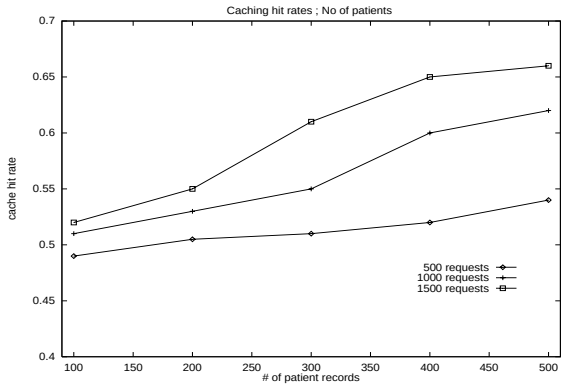

Figure 3: Cache hit; No. patient records.

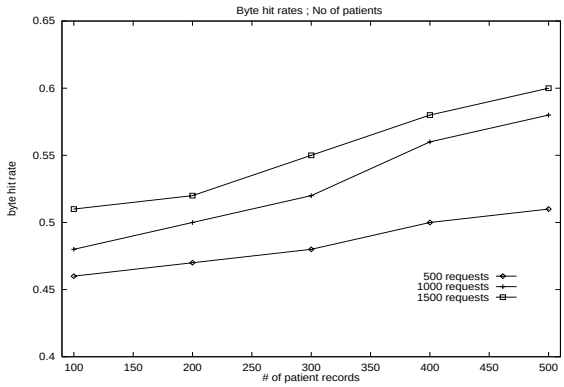

Figure 4: Byte hit; No. patient records.

The most populart performance metrics in Caching are cache-hit ratio and byte-hit ratio :

- **Cache hit ratio :** represents the percentage of all requests being serviced by a cache copy of the requested XML medical object, instead of contacting the original object's storage unit.
- **Byte hit ratio :** represents the percentage of all medical data transferred from cache, i.e. corresponds to ratio of the size of objects retrieved from the cache server. Byte hit ratio provides an indication of the network bandwidth.

The above metrics are considered to be the most typical ones in order to capture and analyze caching policies [1]. We have experimented on the proposed caching policy with an artificial workload of patient records of various sizes.

Figures 3 and 4 depict the cache and the byte hit ratios (respectively) for three different sets of requests, with respect to the number of patient records considered. More specifically, Figure 3 presents the cache hit ratio for $100, 200, \cdots, 500$ patient records and Figure 4 presents the byte hit ratio for the same workloads. As shown in these figures the cache hits increase as the number of patient records increases and these ratios are quite beneficial at rates reaching almost 70%.

5 Conclusions

We have proposed a caching policy for XML structured medical data. The scheme was experimented by considering an artificial workload of patient records structured as XML documents. We have considered these XML documents to reside among various storage units and we have applied the caching scheme over several sets of patient records which have been requested regularly. The caching scheme has been proven quite beneficial since we have resulted in high cache hit and byte hit ratios, thus the requested patient record was most likely found in the considerd cache area in faster rates with no need to contact the original storage unit. Furthermore, the applied cache replacement policy guarantees the accuracy and the confrontation of the cached to the originally stored medical data.

References

[1] C. Aggarwal, J. Wolf and P.S.Yu: Caching on the World Wide Web,*IEEE Transactions on Knowledge and Data Engineering*, Vol.11,No.1,pp.94-107,Jan-Feb 1999.

[2] M. A. Blaze: Caching in Large-Scale Distributed File Systems, Princeton University, PhD thesis, Jan 1993.

[3] B. Bos : The XML Data Model, *http://www.w3.org/XML/Datamodel.html*, 1999.

[4] H. C. Chueh, W. F. Raila, D.A. Berkowicz, G.O. Barnett: "An XML Portable Chart Format", *Proceedings of the American Medical Informatics Association Symposium AMIA'98*, Nov. 1998.

[5] A. K. Dubey, H. Chueh: "Using the Extensible Markup Language (XML) in Automated Clinical Practice Guidelines", *Proceedings of the American Medical Informatics Association Symposium AMIA'98*, Nov. 1998.

[6] C.C. Kanne and G. Moerkotte : Efficient Storage of XML data, *Technical Report*, University of Manheim, Germany, Dec. 1999.

[7] A. Rossi Mori, F. Consorti: "Structures of Clinical Information in Patient Records", *Proceedings of the American Medical Informatics Association Symposium AMIA'99*, Nov. 1999.

[8] A.S. Pombortsis : "Communication Technologies in Health Care Environments", *Medical Informatics*, Vol. 52, pp.61-70, 1998.

[9] A. Vakali: A Web-based evolutionary model for Internet Data Caching, *Proceedings of the 2nd International Workshop on Network-Based Information Systems, NBIS'99*,IEEE Computer Society Press, Florence,Italy, Aug 1999.