Annals of the University of Craiova, Mathematics and Computer Science Series
Volume 43(2), 2016, Pages 280–286
ISSN: 1223-6934

Adapting the semantic cache for CMIS eXtent

Mihai Stancu and Florin Slabu

Abstract. XML is the standard exchange format in web applications due to its portability and information exchange features. When applications communicate with each other or integrate information from several data sources, this standard is very useful. When considering a higher level application, the ECM Systems are the standard platforms for developing applications that manage all the unstructured electronic information inside an organization. CMIS is a new standard that unify the data exchange format between the existing ECM Systems. A new extension of this standard has been proposed to make possible the information exchange operations between ECM Systems and XML Databases. In this paper we try to enlarge the CMIS eXtent platform by adding a semantic cache module into the federated search applications. In this way, the CMIS eXtent can benefit from the advantages of the semantic cache systems.

Key words and phrases. XML, information exchange, document management, semantic cache.

1. Introduction

Recently, the major ECM vendors, seeking the interoperability of the ECM systems, developed a new standard called CMIS (Content Management Interoperability Services). This standard offers a common way to share and exchange information between multiple ECM systems. Considering the growing need for data exchange software and the fact that XML is the acknowledged standard for data exchange, a CMIS eXtent has been proposed [14] in order to exchange the information between ECM Systems and XML Databases as well. The rest of this paper is organized as follows. Section 2 offers a short description of the CMIS, the interoperability standard for ECM systems. Section 3 provides an overview of the CMIS eXtent for XML databases highlighting the key features of the CMIS Interface for xDB and the federated search application. Section 4 describes the advantages of the semantic cache and the implementation of a semantic cache module into the federated search application. Section 5 shows some performance experiments that have been done for the proposed CMIS eXtent. The article ends with conclusions an future work.

2. CMIS - an interoperability standard for ECM systems

Content Management Interoperability Services (CMIS) [2] is a specification for improving interoperability between Enterprise Content Management systems. OASIS approved CMIS as an OASIS Specification on May 1, 2010. CMIS uses Web services to enable information to be shared across Internet protocols, among document systems,

Received October 08, 2016.

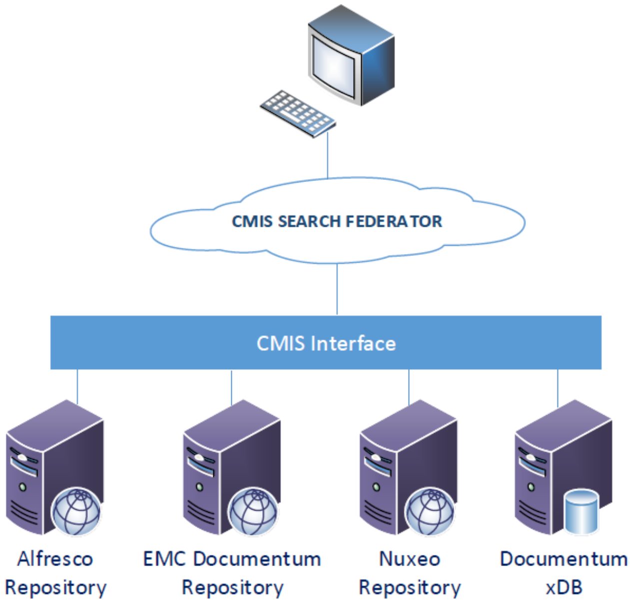


FIGURE 1. CMIS collaboration scenario.

publishers and repositories, within one enterprise and between companies. The CMIS standard provides a common data model, covering typed objects (like files or folders) with generic properties that can be set or read. In addition, there may be a checkout and version control facility, and the ability to define and create generic relations. The standard describes a set of generic services for modifying and querying the data, and several protocol bindings for these services, including SOAP and REST. The model is based on common architectures of document management systems.

3. CMIS eXtent for XML databases

The CMIS eXtent, proposed in [14], present a collaboration scenario in which certain ECM repositories can be involve alongside with a XML Database that was also included. These data sources were accessed via CMIS standard, each one having a CMIS Interface for the communication with the main application that is a federated search. The global architecture of this collaboration scenario is depicted in Figure 1.

In this sample application, for demonstration purposes, the data sources access cases were simplified using only the EMC Documentum repository as the ECM repository involved in the collaboration scenario, and the EMC xDB as the XML database. The Figure 2 shows this sample colaboration case.

Considering that for each ECM repository, a CMIS interface is provided by the each ECM vendor, the new proposal has studied a CMIS interface that can provide the data access in the CMIS format for xDB data source. A sample of queriable data in xDB database is shown in Figure 3.

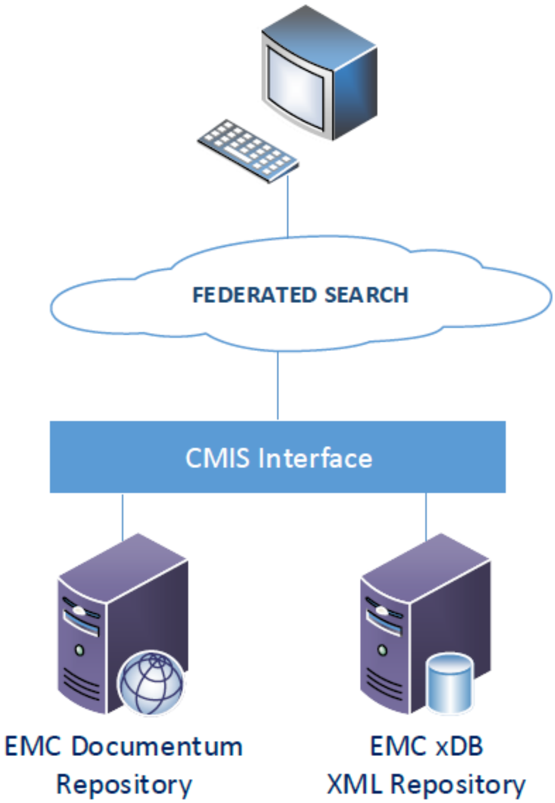
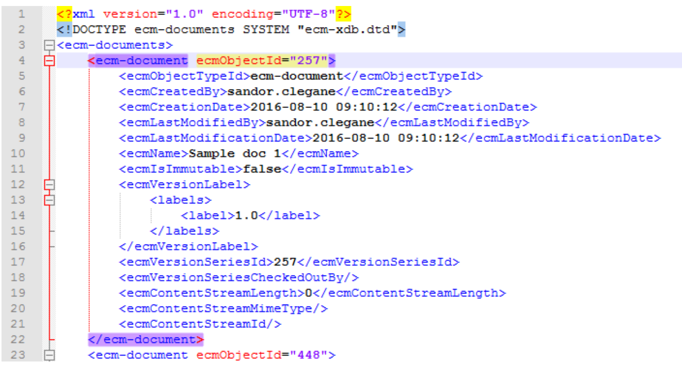
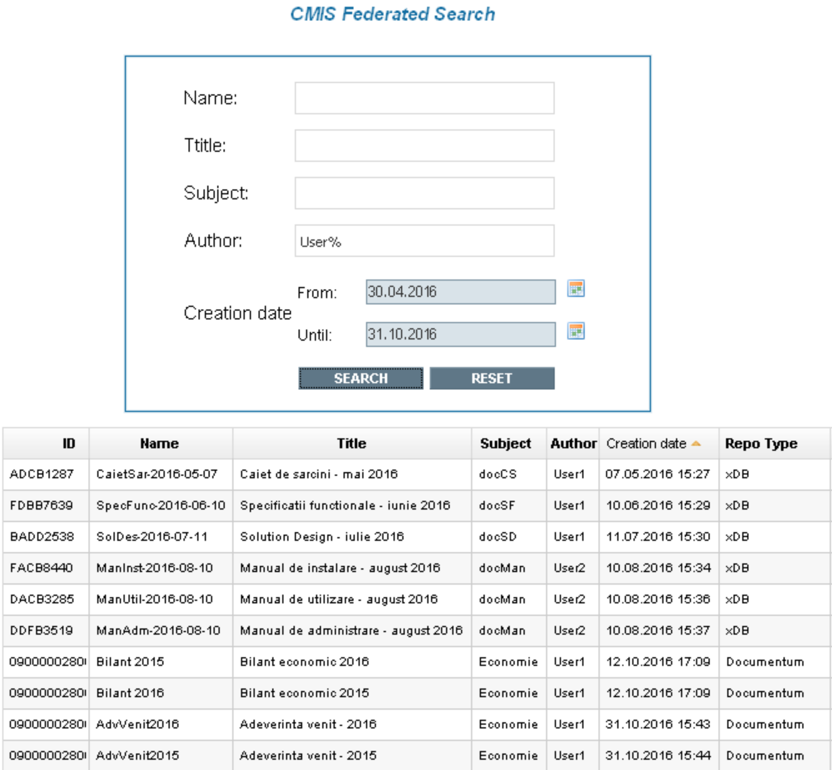


FIGURE 2. Documentum - xDB collaboration case.



```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ecm-documents SYSTEM "ecm-xdb.dtd">
<ecm-documents>
    <ecm-document ecmObjectId="257">
        <ecmObjectTypeId>ecm-document</ecmObjectTypeId>
        <ecmCreatedBy>sandor.clegane</ecmCreatedBy>
        <ecmCreationDate>2016-08-10 09:10:12</ecmCreationDate>
        <ecmLastModifiedBy>sandor.clegane</ecmLastModifiedBy>
        <ecmLastModificationDate>2016-08-10 09:10:12</ecmLastModificationDate>
        <ecmName>Sample doc 1</ecmName>
        <ecmIsImmutable>false</ecmIsImmutable>
        <ecmVersionLabel>
            <labels>
                <label>1.0</label>
            </labels>
        </ecmVersionLabel>
        <ecmVersionSeriesId>257</ecmVersionSeriesId>
        <ecmVersionSeriesCheckedOutBy/>
        <ecmContentStreamLength>0</ecmContentStreamLength>
        <ecmContentStreamMimeType/>
        <ecmContentStreamId/>
    </ecm-document>
    <ecm-document ecmObjectId="448">
```

FIGURE 3. Queriable XML data in xDB.



ID	Name	Title	Subject	Author	Creation date	Repo Type
ADCB1287	CaietSar-2016-05-07	Caiet de sarcini - mai 2016	docCS	User1	07.05.2016 15:27	xDB
FDBB7639	SpecFunc-2016-06-10	Specificatii functionale - iunie 2016	docSF	User1	10.06.2016 15:29	xDB
BADD2538	SolDes-2016-07-11	Solution Design - iulie 2016	docSD	User1	11.07.2016 15:30	xDB
FACB8440	ManInst-2016-08-10	Manual de instalare - august 2016	docMan	User2	10.08.2016 15:34	xDB
DACB3285	ManUtil-2016-08-10	Manual de utilizare - august 2016	docMan	User2	10.08.2016 15:36	xDB
DDFB3519	ManAdm-2016-08-10	Manual de administrare - august 2016	docMan	User2	10.08.2016 15:37	xDB
0900000280	Bilant 2015	Bilant economic 2016	Economie	User1	12.10.2016 17:09	Documentum
0900000280	Bilant 2016	Bilant economic 2015	Economie	User1	12.10.2016 17:09	Documentum
0900000280	AdvVenit2016	Adeverinta venit - 2016	Economie	User1	31.10.2016 15:43	Documentum
0900000280	AdvVenit2015	Adeverinta venit - 2015	Economie	User1	31.10.2016 15:44	Documentum

FIGURE 4. The returned results of a federated search.

The prototype platform of the CMIS interface with support for xDB database provides also a web application as the user interface for data querying. Figure 4 shows the federated search form and the returned results.

4. Semantic cache module for CMIS Interface

A semantic cache mechanism manages the client's cache by using semantic regions of cache. These regions are somehow similar with the pages used in the page cache method and correspond to a tuples aggregation. However, unlike the pages, the semantic regions have a dynamic format and size. When a new query is addressed to the client, this query is broken in two parts as follows:

- Probe query - the part of the query whose result can be served directly from cache.
- Remainder query - the part of the original query whose results are not contained in the local cache and must be received from the server.

The results obtained from these two queries are then combined in pages and then served to the client. The establishing of the probe query is done by using the query containment theory [8].

This cache mechanism, like all cache systems, has a limited storage memory. When the regions stored in the cache system exceed this limit, a cache replacement strategy must be considered.

Partial cache replacement strategies. An efficient partial cache replacement strategy was studied in ACE-XQ project [7]. This strategy uses utility values to refine the representation of different constituent fragments. Thus, each query descriptor has attached a table with XPath values for each element returned by the query. Along with these XPath values, the table also stores each particular utility value. Thus, when a cached query contains the probe query, the particular utility values for all the elements corresponding to the probe query will be updated incrementally. When the cache storage is full, instead of removing a complete region, only the elements with the smallest utility value will be eliminated. After this removing operation, the query descriptor will be updated correspondingly.

The formula propsed in [13], used to compute the utility value, is:

$$(\Delta_t * freq * hits)/t_0 \tag{1}$$

whereΔ_t is the time period for the element being present in the cache system, $freq$ is the frequency of his utilizations, $hits$ is the utilizations number, and t_0 is the initial loading time of the cache system.

In some cases, when XML data contain elements that have IDREF attributes, referring some other elements, a new improved formula has been proposed [13]:

$$((\Delta_t * freq * hits)/t_0) * (k * N_{ref}/N) \tag{2}$$

where N_{ref} is the number of its references, N is the total number of elements, and k is a relevance factor of the references.

The meaning of this new formula is that when an element is referred by some other elements, its importance can be considered bigger than in a normal scenario. In this case, these elements it is preferred not to be removed from cache within an eventual replacement operation.

We have developed a semantic cache module for the federated search application and we have used the partial cache replacement strategy described above. Considering that the queriable XML data (Figure 3) contains IDREF attributes, the second formula was used for computing the utility values. The new semantic cache module was deployed into the CMIS Interface for xDB as can be seen in the Figure 5.

5. Experimental studies

Some experiments was made in [15] using the Neoload performance analysis tool. In the experimental scenario three CMIS queries were addressed both on Documentum and xDB repositories. The results of these experiments highlight better response times for Documentum CMIS Interface. However, the differences are not significant and can be explained by the query optimization used by the underlying relational databases. More, when comparing the native access times to data sources, the results are similar, which mean that an optimal CMIS implementation for xDB can be reliable in this kind of collaboration scenario.

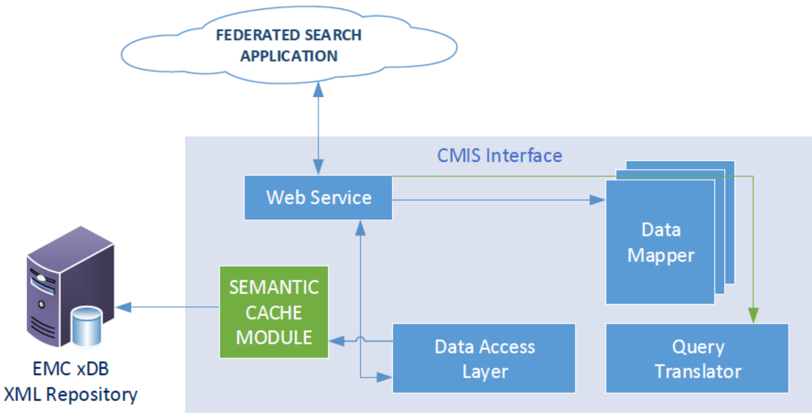


FIGURE 5. The architecture of the CMIS interface for xDB including a semantic cache module.

6. Conclusions and future work

This paper has dealt with the CMIS standard of the ECM Systems and the CMIS eXtent platform that takes advantage of the CMIS interoperability features in the information exchange processes between ECM systems and XML databases. The implementation of such an extension provides us with the interoperability between an EMC Documentum repository and a XML-native database EMC xDB. Additionally, we have developed a new module into this platform that takes advantage of the semantic cache systems.

The possible future work will include some additional experiments in order to measure the improvements brought by this new semantic module. Also, the implementation of the CMIS interface for other XML databases will provide a better future analysis.

References

[1] S. Abiteboul, P. Buneman, D. Suciu, *Data on the Web: From Relations to Semistructured Data and XML*, Morgan Kaufmann, October 26, 1999.

[2] AIIM - Association for Information and Image Management, *CMIS - Content Management Interoperable Services*, Jul 09, 2009. Available at `http://www.aiim.org/Resources/Standards/Articles/CMIS`.

[3] AIIM - Association for Information and Image Management, *What is ECM Enterprise Content Management*, 2001. Available at `http://www.aiim.org/What-is-ECM-Enterprise-Content-Management`.

[4] V. Apparao et al., *Document Object Model (DOM) level 1 specification*, 1998, W3C Recommendation. Available at `http://www.w3.org/TR/REC-DOM-Level-1/`.

[5] S. Boag et al., *XQuery 1.0: An XML query language*, November 2003. Available at `http://www.w3.org/TR/xquery/`.

[6] J. Borden, L. Martin, K. Staken, *XML:DB Initiative for XML Databases*, 2003. Available at `http://xmldb-org.sourceforge.net/index.html`.

[7] L. Chen, E.A. Rundensteiner, ACE-XQ: A CachE-aware XQuery Answering System, In: *Proceedings of the 5th International Workshop on the Web and Databases (WebDB)*, Madison, WI, 31–36, 2002.

[8] C. Chekuri, A. Rajaraman, Conjunctive Query Containment Revisited, *ICDT*, 56–70, 1997.

[9] J. Clark, *XSL transformations (XSLT) specification*, (1999). Available at `http://www.w3.org/TR/WD-xslt`.

[10] JCP - Java Community Process, *JSR 225: XQuery API for Java (XQJ)*, June 24, (2009). Available at `http://jcp.org/en/jsr/summary?id=225`.

[11] D. Megginson et al., *Simple API for XML (SAX)*, (2000). Available at `http://www.saxproject.org/`.

[12] OASIS, *Content Management Interoperability Services (CMIS)* Version 1.0, May 1, (2010). Available at `http://docs.oasis-open.org/cmis/CMIS/v1.0/os/cmis-spec-v1.0.html`.

[13] M. Stancu, Semantic Cache - Optimizing the Semistructured Data Queries, *Annals of the University of Craiova - Mathematics and Computer Science Series* **38** (2011), no. 3.

[14] M. Stancu, CMIS eXtent: a bridge from ECM systems to XML Databases, *Journal of Knowledge, Communications and Computing Technologies* **3** (2011), no. 2.

[15] M. Stancu, Extending CMIS Standard for XML Databases, *Annals of the University of Craiova - Automation, Computers, Electronics and Mechatronics Series* **8(35)** (2011), no. 2.

[16] W3C, *XML Schema*, October 28, (2004). Available at `http://www.w3.org/TR/xmlschema-2/`.

[17] W3C, *XML Path Language (XPath)* 2.0, (2003). Available at `http://www.w3.org/TR/xpath20`.

(Mihai Stancu, Florin Slabu) Department of Computer Science, University of Craiova, 13 A.I. Cuza Street, Craiova, 200585, Romania

E-mail address: `mihai.stancu@yahoo.com, florins@central.ucv.ro`