

The Dienst OAI Gateway: A Preservation Gateway for a Legacy Protocol

Terry L. Harrison, Michael L. Nelson, Mohammad Zubair
Old Dominion University
Department of Computer Science
Norfolk VA, 23529 USA
+1 757 683 4817
{tharriso,mln,zubair}@cs.odu.edu

ABSTRACT

Though the Open Archive Initiative Protocol for Metadata Harvesting (OAI-PMH) is becoming the defacto standard for digital libraries, some of its predecessors are still in use. Although a limited number of Dienst repositories continue to be populated, others are precariously unsupported. The Dienst Open Archive Gateway (DOG) is a gateway between the OAI-PMH and the Dienst (version 4.1) protocol. DOG allows OAI-PMH harvesters to extract metadata records (in RFC-1807 or Dublin Core) from Dienst repositories.

1. INTRODUCTION

Dienst was once a popular, rich digital library (DL) protocol comprised of over 30 verbs [1]. A listing of Dienst service/verb pairs can be found in Appendix 1. The Open Archive Initiative Protocol for Metadata Harvesting (OAI-PMH) [2] represents some of the lessons learned during the deployment of Dienst, including a significantly decreased scope. As such, the OAI-PMH has only 6 verbs and introduces the division of responsibility for harvesters and repositories. Appendix 2 offers the 6 OAI-PMH verbs for comparison.

While many of the 100+ institutions that once used Dienst to participate in the Networked Computer Science Technical Reference Library (NCSTRL) have transitioned to the OAI-PMH [3], not all have. Some Dienst repositories continue to be populated and supported, while others are no longer being updated, and in the worst case, no longer being maintained. The NCSTRL website contains a listing of 126 Dienst archives once in existence [4]. Recent investigation reveals that only about 23 of them are functional today (Appendix 3). These archives represent just over 6,000 remaining records, yet some of them have been added as recent as February 10, 2003 [5]. Though this is a far cry from the heyday of Dienst, when NCSTRL contained over 20,000 digital objects, these Dienst implementations do represent valuable metadata, some of which is not available to current OAI-PMH digital library implementations.

The Dienst OAI-PMH Gateway (DOG) was created to allow OAI-PMH harvesters to extract data from existing and at-risk Dienst repositories (Dienst versions 4.1.x). DOG was initially created to allow for the harvesting of the ICASE Dienst repository for the OAI-PMH enabled NASA Technical Report Server [6], but the general nature of DOG allows it to be used for any Dienst repository, including those current DL projects based on Dienst, such as OpenDLib [7].

2. OAI-PMH VERB IMPLEMENTATION

DOG is a Java servlet and is available both as a demonstration service at ODU (running on a Tomcat 4.0.4 servlet server) and as a tar file for local implementation (dlib.cs.odu.edu). DOG allows a specified Dienst repository to appear as a normal baseURL to a harvester:

```
http://128.82.7.113:5187/dog/servlet/dataprovider/      (DOG)
dienst.iei.pi.cnr.itSLASHdienstSLASH/                (Dienst)
?verb=ListRecords&metadataPrefix=oai_dc               (OAI-PMH)
```

The URL of the Dienst server is imbedded the URL, so that the legally identifiable parameters of the request conform to the OAI-PMH. DOG parses out this URL and issues the appropriate Dienst verbs to the Dienst repository (Table 1), parsing the RFC-1807 [8] plain text results and returning an OAI-PMH formatted response. Metadata can be returned as either RFC-1807 or Dublin Core (DC) [9]. Appendices 4-6 illustrate these respective file formats. Figure 1 illustrates this architecture. The mapping from RFC-1807 to DC is modifiable if a local installation is chosen (Table 2). Additional documentation is available at <http://dlib.cs.odu.edu> showing how to make calls to DOG. A linked test site permits users to see these calls in action.

OAI-PMH request	Dienst 4.1 request(s) used
Identify (to seek sample Id) (to seek earliest datestamp)	Repository/2.0/List-Contents (requesting this value also checks if archive is up) Repository/2.0/List-Contents (This call is only made if a value has not been cached)
ListIdentifiers	Index/2.0/List-Contents (*note -- Dienst "?file-after" option intentionally not used due to ill-implementation by Dienst archives) Index version of Dienst verb used w. OAI-PMH due to added metadata requirements over OAI1.1
ListRecords	Index/2.0/List-Contents (*note -- Dienst "?file-after" option intentionally not used due to ill-implementation by Dienst archives)
GetRecord (oai - identifier)	Index/2.0/Bibliography/handle handle=identifier in Dienst format (*note: If http exception thrown, it may be because the handle is invalid or because the archive is down. To clarify this 404 error, a call to Repository/2.0/List-Contents is made and evaluated)
ListSets	no calls made (response hardcoded -- makes assumption that sets are not supported in Dienst archives)
ListMetadataFormats	no calls made (response hardcoded)
ListMetadataFormats (w. identifier arg)	Index/2.0/Bibliography/handle

Table 1. Dienst requests used to generate metadata for OAI-PMH request

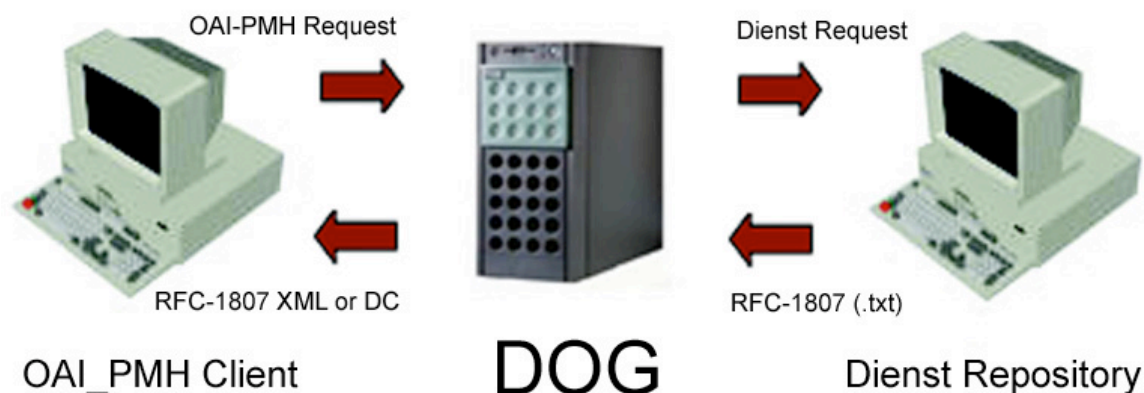


Figure 1. DOG process flow

3. VERB IMPLEMENTATION

DOG's architecture is loosely based on the Arc dataprovider developed at Old Dominion University (ODU) [10]. Several objects are created to handle a call to DOG. The "dataprovider" handles the bulk of the metadata processing, including OAI-PMH error handling and response generation. The "simpleIdentity" object acts as a repository of both hard-coded and

retrieved data and has basic functionality to do such tasks as parse URLs and format Dienst handles from OAI identifiers. The “xmlTool” object is used exclusively by the “dataproducer” and facilitates the building of the DOM tree for the .xml response. The “Connect” object handles connections to the Dienst archive, returning results as a String to the calling method. The servlet discovers (via the “simpleIdentity” object) which OAI verb has been called and passes this request to the “dataproducer”, which orchestrates method calls (named after each OAI verb) to fulfill the request.

3.1 Identify

The Identify method extracts much of its required element content from hard-coded variables or from parsing the request URL, with the exceptions of earliestDatestamp and sampleIdentifier.

The earliestDatestamp element was added in OAI-PMH 2.0 and guarantees to harvesters a lower bound on record datestamps. Datestamp maps to the Dienst “entry” field, required in RFC-1807, for each record. To discover the earliest date is an expensive operation and requires the parsing of all records for a given Dienst repository. To reduce this overhead, the resulting value is cached so that the operation need only be performed once per repository. This operation is handled by “simpleIdentity” as it is the object tasked with population of such variables. As with all datestamps, RFC-1807 format (May 31, 2003) is converted to ISO-8061 format (2003-05-31).

Values for the sampleIdentifier element are intentionally not cached, as the Dienst call needed to extract this from the repository also provides a status check on it, returning an http exception if the archive is no longer available. Some escaped character sequences had to be created to accommodate characters in some Dienst repository identifiers that did not conform with OAI-PMH character restrictions for the sampleIdentifier element content (fig . 2). One such case is the underscore in repository identifiers such as nctrl.mit_ai. To resolve this, the “_” char is replaced with “UNDERSCORE”.

```
<simpleType name="sampleIdentifierType">
  <restriction base="string">
    <pattern value="oai:[a-zA-Z][a-zA-Z0-9\-\_]*(\.[a-zA-Z][a-zA-Z0-9\-\_]+)+[a-zA-Z0-9\-\_!\~*'\(\);/\?:@&=+\$,%]+"/>
    <!--meta ., \, ?, *, +, {, } (, ), [ or ] -->
  </restriction>
</simpleType>
```

Figure 2. The sampleIdentifierType does not permit underscores in the repository name portion of the identifier [11].

3.2 ListSets

Since the Dienst collection service was never widely adopted, most Dienst implementations have no equivalent to the OAI-PMH concept of sets. Thus, the response to a ListSets request is hard coded to be the “noSetHierarchy” error.

3.3 ListMetadataFormats

Since Dienst responds with rfc-1807 and OAI-PMH requires support of Dublin Core, DOG has been designed to support both. A response indicating this has been hard-coded into DOG and is automatically generated unless an identifier argument has been included. This option requires checking Dienst to see if metadata for the given identifier exists and producing the necessary error, if it does not exist.

3.4 GetRecord

Getting a single record from the Dienst archive is relatively simple, as the request maps well into Dienst protocol. (Table 1) The OAI-PMH identifier is formatted as a Dienst handle and appended into a Dienst record request. The record is returned in

RFC-1807 as plain text, which DOG parses, returning XML formatted RFC-1807 or DC record (Table 2). Error handling issues are discussed in section 4.0.

3.5 ListIdentifiers & ListRecords

ListIdentifiers and ListRecords utilize the same Dienst calls and only differ in the amount of metadata returned for each record. DOG issues an `Index/2.0/List-Contents` request that gives it an rfc-1807 metadata dump of all the records. Initially the parameter “file-after” was utilized to implement an OAI-PMH “from” argument, but since many Dienst archives had not implemented this argument, its use threatened to corrupt result sets. Once the metadata dump is received, DOG parses it, extracting any records that meet any given “from” or “until” arguments. Interestingly, by its ability to handle the “until” parameter, DOG adds utility that did not previously exist within the Dienst protocol. DOG does not cache the results of the metadata dump, which permits it to provide realtime harvesting.

4. ISSUES & ERROR HANDLING

OAI-PMH 2.0 adds many requirements for handling OAI-PMH errors, which are distinct from http errors. DOG is fully compliant with the OAI-PMH and goes to lengths to provide the user with useful error message content when an OAI-PMH error occurs.

DOG also is careful to distinguish http errors and errors to be handled by the OAI-PMH. For example, an http 404 (File Not Found) error could be the result of the archive being down, a bad repository name in the request or the file requested by an identifier not existing. Invalid syntax issues like a malformed repository name are easily resolved by pattern matching to OAI-PMH regular expressions provided in its schema (fig. 2). Should the syntax be deemed valid, then it must be determined if the problem is a communications fault or a non-existent file. To resolve these uncertainties, DOG issues a `Repository/2.0/List-Contents/verb`. If a Dienst connection is made a 404 is not returned and it is concluded that the identifier being requested does not exist. Subsequently, an OAI-PMH error “idDoesNotExist” is returned instead of an http 404 error. Http errors are handled separately and display only the error code and message.

A different development issue was the hiding of the Dienst archive URL into the request URL to DOG. Traditionally this might be accomplished through the use of standard escape character “%2F” for the forward slashes, but a security feature in Tomcat 4.0.4 which does not handle these escape characters before the method call, necessitated the use of the custom “SLASH” chars to escape instances of “/” in the Dienst URL. In anticipation of the resolution of this issue, DOG handles escape characters “%2F”, “%2F” and “SLASH”.

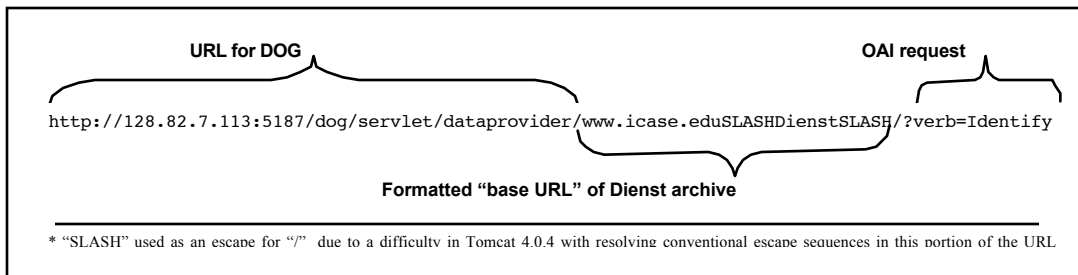


Figure 3. Sample OAI request to Dienst archive, via DOG

Another issue faced was in the metadata inconsistency in the use of the RFC-1807 fields “ID::” and “HANDLE:: “. Since handles are unique permanent identifiers of the form `HANDLE:: <repository>/<identifier>` they seem the most logical choice for <identifier> content. However, because they are optional, they may be missing altogether from an archive (ie: `dienst.iei.pi.cnr.it`). In this case, DOG uses data from the mandatory ID:: field, which is typically the same data. This also works when searching for a record. For example, when a GetRecord request is made and DOG is comparing Dienst record metadata for a match, should there not be a HANDLE:: field, then the ID:: field is used instead:

Dienst call which displays record without HANDLE:: field :

```
http://dienst.iei.pi.cnr.it/Dienst/Index/2.0/Bibliography/ercim.cnr.cnuce/
1996-ARCA-T13-API
```

DOG call for record which does not contain a HANDLE:: field :

```
http://128.82.7.113:5187/dog/servlet/dataprovider/dienst.iei.pi.cnr.itSLASHDie
nstSLASH/?verb=GetRecord&identifier=oai:ercim.cnr.cnuce:ercim.cnr.cnuce/1996-
ARCA-T13-API&metadataPrefix=oai_dc
```

Due to RFC-1807 syntax, the ID:: field data will contain two forward slashes (ID:: <publisher-id>//<free-text>) instead of one as in the HANDLE:: field :

```
(Dienst)      ID:: ncstrl.umassa_cs//UM-CS-1984-021
(Dienst)      HANDLE:: hdl:ncstrl.umassa_cs/UM-CS-1984-021
(OAI-PMH)     <identifier>ncstrl.umassaUNDERSCOREcs/UM-CS-1984-021</identifier>
```

To avoid confusion the DOG response represents the <identifier> using only one slash. DOG can resolve this, should it need to match the ID:: field later. Additionally, DOG will handle requests that mistakenly use two slashes in the identifier.

5. FUTUREWORK

The new OAI-PMH has introduced some interesting avenues of exploration. Through the use of the <friends> element, we might announce the existence of other (cached) Dienst archive URLs that may also be harvested. We will need to be sure to avoid appending non-existent repositories to the cache (such as “mmit” instead of “mit”) and consider how to deal with defunct repositories. These may need to be manually removed from the cache.

6. CONCLUSIONS

The Dienst Open Archive Gateway (DOG) is a gateway between the OAI-PMH and the Dienst (version 4.1) protocol. DOG allows OAI-PMH harvesters to extract metadata records (in RFC-1807 or Dublin Core) from Dienst servers. DOG has overcome the many issues faced in the protocol conversion process of building a gateway. DOG hides the Dienst server name in the URL and makes sure the names are OAI-PMH schema compliant, transforming them if they are not. Dienst does not handle deleted records and subsequently DOG does not either. Lacking a real Dienst equivalent implementation to the concept of “sets”, DOG issues a “noSetHierarchy” response for ListSets. The earliestDatestamp for each repository is cached to reduce overhead, while the sampleIdentifier is not, as its retrieval confirms that the Dienst server is operational. Dienst RFC-1807 dates are converted to OAI-PMH standard ISO8061 (YYY-MM-DD). Adding functionality to Dienst servers, DOG handles both “from” and “until” date parameters, regardless of whether Dienst “file-after” date processing is implemented. Finally, errors are carefully processed and separated into distinct http and OAI-PMH errors, each handled in a unique manner.

While the previous NCSTRL transition project provided a specific strategy for OAI-PMH conversion, DOG provides on-demand, general OAI-PMH compatibility for any Dienst archive. DOG has been exhaustively tested with the Repository Explorer and has been used to harvest ICASE Dienst repositories. Sadly, ICASE no longer exists at NASA and its Dienst repositories have been shut down. This underscores the importance of providing protocol gateways with an eye toward long-term preservation.

7. REFERENCES

- [1] Davis, J., and Lagoze, C. NCSTRL: design and deployment of a globally distributed digital library. *Journal of the American Society for Information Science*, 51(3), 2000, 273-280.
- [2] Lagoze, C., Van de Sompel, H., Nelson, M. and Warner, S. The Open Archives Initiative Protocol for Metadata Harvesting Version 2.0; available from <http://www.openarchives.org/OAI/openarchivesprotocol.html>.
- [3] Anan, H., Liu, X., Maly, K., Nelson, M., Zubair, M., French, J., Fox, E., and Shivakumar, P. Preservation and transition of NCSTRL using an OAI-based architecture. in *Proceedings of JCDL 2002 (Portland OR, July 2002)*, 181-182.
- [4] Dienst Repositories, "Participating Publishers"; available from <http://ncstrl.mit.edu/Dienst/UI/2.0/ListPublishers>.
- [5] University of California, San Diego, Computer Science and Engineering Dienst Records; available from <http://www.cs.ucsd.edu:80/Dienst/Index/2.0/List-Contents>.
- [6] Nelson, M., Rucker, J., and Harrison, T. OAI and NASA scientific and technical information. *Library Hi-Tech*, 21(2), 2003.
- [7] Castelli, D., and Pagano, P. OpenDLib: A digital library service system. in *Proceedings of ECDL 2002 (Rome, Italy, September 2002)*, 292-308.
- [8] Lasher, R. & Cohen, D. (1995). A format for bibliographic records. Internet RFC-1807; available from <ftp://ftp.isi.edu/in-notes/rfc1807.txt>
- [9] Dublin Core Metadata for Resource Discovery, IETF RFC 2413, July 1998; available at <ftp://ftp.isi.edu/in-notes/rfc2413.txt>.
- [10] Liu, X., Maly, K., Zubair, M. and Nelson, M. L. Arc: An OAI Service Provider for Cross Archive Searching, in *Proceedings of JCDL 2001 (Roanoke VA, June 2001)*, 65-66.
- [11] OAI-PMH Identifier Schema; available from <http://www.openarchives.org/OAI/2.0/oai-identifier.xsd>.

Appendix 1. Dienst 4.1 Verbs

Verb	Required args	Optional args
Repository/2.0/List-Contents	none	none
Repository/2.0/Body/	handle,format	none
Repository/2.0/Page/	handle, format, page number	none
Repository/2.0/Npages/	handle,format	none
Repository/2.0/Formats/	handle	none
Index/2.0/List-Contents	none	file-after
Index/2.0/Bibliography/	handle	none
Index/2.0/SearchBoolean	none	title, author, keyword, abstract, boolean, authority, name
Meta/2.0/Publishers	none	none
Meta/2.0/Indices	none	none
Meta/2.0/Repositories	none	none
Meta/2.0/Lite	none	none
UI/2.0/Search	none	(search keywords)
UI/2.0/QueryNF	none	(search keywords)
UI/2.0/Describe/	handle	allowed but not defined
UI/2.0/BrowseYears	none	none
UI/2.0/ListYears	(span of years)	none
UI/2.0/BrowseAuthors	none	none
UI/2.0/ListAuthors	(span of letters)	none
UI/1.0/RegistryAdd	none	name,email,password,verifypwd,institution,address,phone
UI/1.0/RegistryDescribe	none	userid, password
UI/1.0/RegistryModify	none	userid, name, email, password, verifypwd, newpwd, institution, address, phone
LibMgt/2.0/Submit	none	(varies)
Registry/1.0/Add	password, name, email	userid, institution, address, phone
Registry/1.0/Authenticate/	userid, password	none
Registry/1.0/List/	userid, password	none
Registry/1.0/Modify/	userid, password	name, email, institution, address, phone, newpwd
Registry/1.0/Delete	userid, password	none
Info/2.0/Version	none	none
Info/2.0/Log	none	class, start, end
Info/2.0/Log_Summary	none	report, start, end
Info/2.0/List-Services	service	none
Info/2.0/Describe-Verb	service, verb	none

Appendix 2. OAI-PMH 2.0 Verbs

	Verb	Required args	Optional args
1	Identify	none	none
2	List Identifiers	metadataPrefix	from, until, set, (resumptionToken)
3	List Sets	none	identifier
4	ListMetadataFormats	none	identifier
5	GetRecord	identifier, metadataPrefix	none
6	ListRecords	metadataPrefix	from, until, set, (resumptionToken)

* resumption token is an exclusive argument

Appendix 3. Active Dienst Repositories

	Archive	BaseURL	HANDLE:: field	Current Records
1	CNR - Istituto per le Applicazioni della Matematica e dell'Informatica (Milano)	http://exlibris.ian.pv.cnr.it:80/Dienst/	no	267
2	CNR - Istituto per la Matematica Applicata (Genova)	http://exlibris.ian.pv.cnr.it:80/Dienst/	no	(see above)
3	CNR - Istituto di Analisi Numerica (Pavia)	http://exlibris.ian.pv.cnr.it:80/Dienst/	no	(see above)
4	CNR - Istituto CNUCE (Pisa)	http://dienst.iei.pi.cnr.it:80/Dienst/	no	178
5	CNR - Istituto di Elaborazione della Informazione (Pisa)	http://dienst.iei.pi.cnr.it:80/Dienst/	no	(see above)
6	CNR - Istituto di Linguistica Computazionale (Pisa)	http://dienst.iei.pi.cnr.it:80/Dienst/	no	(see above)
7	CNR - Istituto di Matematica Computazionale (Pisa)	http://dienst.iei.pi.cnr.it:80/Dienst/	no	(see above)
8	CNR - Istituto di Studi sulla Ricerca e sulla Documentazione Scientifica (Roma)	http://dienst.iei.pi.cnr.it:80/Dienst/	no	(see above)
9	CNR - Istituto per le Applicazioni Telematiche (Pisa)	http://dienst.iei.pi.cnr.it:80/Dienst/	no	(see above)
10	Foundation for Research and Technology - Hellas. Institute of Computer Science	http://dienst.csi.forth.gr:80/Dienst/	no	240
11	M.I.T., Artificial Intelligence Lab	http://ncstrl.mit.edu:80/Dienst/	yes	1487
12	M.I.T., Laboratory for Computer Science	http://ncstrl.mit.edu:80/Dienst/	yes	(see above)
13	Masaryk University - Faculty of Informatics	http://dienst.muni.cz:80/Dienst/	no	65
14	National and Capodistrian University of Athens. Department of Informatics.	http://alexandra.di.uoa.gr:80/Dienst/	no	87
15	New York University	http://csdocs.cs.nyu.edu:80/Dienst/	no	267
16	Rice University	http://cs-tr.cs.rice.edu:80/Dienst/	no	138
17	SICS - Swedish Institute of Computer Science	http://dienst.sics.se:80/Dienst/	yes	94
18	Universitaet Leipzig. Institute fuer Informatik	http://dol.uni-leipzig.de:80/Dienst/	yes	461
19	Universitaet Stuttgart, Germany. Fakultae Informatik	http://ncstrl.informatik.uni-stuttgart.de:80/Dienst/	yes	886
20	University of California, San Diego, Computer Science and Engineering	http://www.cs.ucsd.edu:80/Dienst/	no	108
21	University of Hamburg, Germany	http://medoc.informatik.uni-hamburg.de:80/	no	57
22	University of Illinois at Urbana-Champaign	http://www.cs.uiuc.edu:80/Dienst/	no	517
23	University of Massachusetts, Amherst	http://www.cs.umass.edu:80/Dienst/	yes	1150

Appendix 4. Dienst record (RFC-1807 .txt file)

```
BIB-VERSION:: CS-TR-v2.1E
ID:: ercim.cnr.cnuce//1996-ARCA-T13-API
PAGES:: 30
ACM:: C-2-2 Network Protocols
TYPE:: EC-deliverable
TITLE:: ARCA API Specification Document
ENTRY:: May 4, 1999
AUTHOR:: Catoni, Alberto
AUTHOR:: Loffredo, Mario
AUTHOR:: Romano, Giuseppe
AUTHOR:: Signore, Oreste
LANGUAGE:: English
ABSTRACT:: This report presents the API functions
prototypes and the necessary datatypes to
implement the communication between the ARCA
Target and the OPAC.
EMAIL:: O.Signore@cnuce.cnr.it
KEYWORD:: SR
KEYWORD:: Z39.50
KEYWORD:: YAZ
KEYWORD:: C++
DATE:: 1996-01-01
END:: ercim.cnr.cnuce//1996-ARCA-T13-API
```

Appendix 5. OAI-PMH record (RFC-1807 .xml file)

```
<?xml version="1.0" encoding="UTF-8" ?>
- <OAI-PMH xmlns="http://www.openarchives.org/OAI/2.0/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/ http://www.openarchives.org/OAI/2.0/OAI-PMH.xsd">
  <responseDate>2003-02-19T05:30:56-05:00</responseDate>
  <request identifier="oai:ercim.cnr.cnuce:ercim.cnr.cnuce/1996-ARCA-T13-API" metadataPrefix="oai_rfc1807"
verb="GetRecord">http://128.82.7.113:5187/dog/servlet/dataprovider/dienst.iei.pi.cnr.itSLASHDienstSLASH/</request>
- <GetRecord>
- <record>
- <header>
  <identifier>oai:ercim.cnr.cnuce:ercim.cnr.cnuce//1996-ARCA-T13-API</identifier>
  <datestamp>1999-05-04</datestamp>
</header>

- <metadata>
- <rfc1807 xmlns="http://info.internet.isi.edu:80/in-notes/rfc/files/rfc1807.txt" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://info.internet.isi.edu:80/in-notes/rfc/files/rfc1807.txt
http://www.openarchives.org/OAI/1.1/rfc1807.xsd">
  <bib-version>CS-TR-v2.1E</bib-version>
  <id>ercim.cnr.cnuce//1996-ARCA-T13-API</id>
  <entry>May 4, 1999</entry>
  <title>ARCA API Specification Document</title>
  <type>EC-deliverable</type>
  <author>Catoni, Alberto</author>
  <author>Loffredo, Mario</author>
  <author>Romano, Giuseppe</author>
  <author>Signore, Oreste</author>
  <date>1996-01-01</date>
  <pages>30</pages>
  <keyword>SR</keyword>
  <keyword>Z39.50</keyword>
  <keyword>YAZ</keyword>
  <keyword>C++</keyword>
  <language>English</language>
  <abstract>This report presents the API functions prototypes and the necessary datatypes to implement the communication
between the ARCA Target and the OPAC.</abstract>
</rfc1807>
</metadata>
</record>
</GetRecord>
</OAI-PMH>
```

Appendix 6. OAI-PMH record (Dublin Core .xml file)

```
<?xml version="1.0" encoding="UTF-8" ?>
- <OAI-PMH xmlns="http://www.openarchives.org/OAI/2.0/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/ http://www.openarchives.org/OAI/2.0/OAI-PMH.xsd">
  <responseDate>2003-02-19T05:53:40-05:00</responseDate>
  <request identifier="oai:ercim.cnr.cnuce:ercim.cnr.cnuce/1996-ARCA-T13-API" metadataPrefix="oai_dc"
verb="GetRecord">http://128.82.7.113:5187/dog/servlet/dataprovider/dienst.iei.pi.cnr.itSLASHDienstSLASH/</request>
- <GetRecord>
- <record>
- <header>
  <identifier>oai:ercim.cnr.cnuce:ercim.cnr.cnuce//1996-ARCA-T13-API</identifier>
  <datestamp>1999-05-04</datestamp>
</header>
- <metadata>
- <oai_dc:dc xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:oai_dc="http://www.openarchives.org/OAI/2.0/oai_dc/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.openarchives.org/OAI/2.0/oai_dc/ http://www.openarchives.org/OAI/2.0/oai_dc.xsd">
  <dc:title>ARCA API Specification Document</dc:title>
  <dc:creator>Catoni, Alberto</dc:creator>
  <dc:creator>Loffredo, Mario</dc:creator>
  <dc:creator>Romano, Giuseppe</dc:creator>
  <dc:creator>Signore, Oreste</dc:creator>
  <dc:subject>SR</dc:subject>
  <dc:subject>Z39.50</dc:subject>
  <dc:subject>YAZ</dc:subject>
  <dc:subject>C++</dc:subject>
  <dc:description>This report presents the API functions prototypes and the necessary datatypes to implement the
communication between the ARCA Target and the OPAC.</dc:description>
</oai_dc:dc>
</metadata>
</record>
</GetRecord>
</OAI-PMH>
```