# Open Archives



OPEN ARCHIVES

# Communities, Interoperability, and Services

# Contents

# Program

| | |
|---|---|
| **8:00 am – 8:30 am** | **Registration** |
| **8:30 am – 10:00 am** | **Session One – Introduction to the Open Archives Initiative** |

**Opening Remarks and Historical Overview**
*Edward Fox, Virginia Tech* (30 minutes)
Origins of the OAI and overview of the original technical agreements.

**Introductions from Workshop Participants**
chaired by: *Edward Fox, Virginia Tech* (30 minutes)
A short introduction from each participant to familiarize people with one another and contextualize further interaction during the workshop.

**Introduction to the OAI Metadata Harvesting Protocol**
*Hussein Suleman, Virginia Tech* (30 minutes)

| | |
|---|---|
| **10:00 am – 10:30 am** | **Break** |
| **10:30 am – 12:00 pm** | **Session Two – Technical Details** |

**Experiences from Implementers**
*Thorsten Schwander, LANL* and *Tim Cole, UIUC* (30 minutes)

**The OAI Metadata Harvesting Protocol – Communities and Services**
*Hussein Suleman, Virginia Tech* (30 minutes)

**Communities and Services - Practical Applications of OAI Standards**
Framing the afternoon's discussion and forming of breakout groups
*Edward Fox, Virginia Tech* (30 minutes)

| | |
|---|---|
| **12:00 pm – 1:30 pm** | **Lunch** |
| **1:30 pm – 3:30 pm** | **Session Three – Concurrent Group Discussions** |

| **Group 1: Building Communities** | **Group 2: Technical Services** |
|---|---|
| chaired by: *Edward Fox, Virginia Tech*<br>Applicability of OAI to community building, how the OAI standards can be used to support distributed communities and what community support is needed to leverage OAI standards.<br>*Details on next page.* | chaired by: *Hussein Suleman, Virginia Tech*<br>Evaluation of technical standards and current and future directions of standards and services related to the OAI protocol.<br>*Details on next page.* |

| | |
|---|---|
| **3:30 pm – 3:50 pm** | **Break** |
| **3:50 pm – 4:20 pm** | **Session Four – Presentations** |

**Reports from Breakout Groups**
(30 minutes)

| | |
|---|---|
| **4:20 pm – 5:00 pm** | **Session Five – Moving Forward** |

**Moving Forward**
chaired by: *Edward Fox, Virginia Tech* and *Hussein Suleman, Virginia Tech*
Plans for realization of higher-level services and community-building within the framework of OAI.

# Topics for Group Discussions

## Group 1: Community Building

Support for different types of communities:

* ETDs
* Courseware
* Libraries
* Southern History and Culture

Developments that contribute towards building communities:

* Metadata standards that are richer than DC
* Adaptation of workflow models to different communities
* Development of controlled vocabularies and thesauri
* Personal OAI (e.g., Kepler) vs. group-based OAI (e.g., E-Prints)
* Definition of community-specific selective harvesting mechanisms e.g., sets

Community building:

* Social aspects of developing OAI-based community projects

## Group 2: Technical Services

Protocol evaluation:

* Experiences from implementation
* Efficiency
* Support for internationalization
* Correctness of implementations: XML, character sets, etc.
* Changes that could be suggested to improve on the protocol: date granularity, sets for each identifier, etc.
* Complementary protocols to satisfy existing/future needs

Services enabled by the OAI protocol:

* What is the state of the art in service provision ?
* How can the OAI protocol be leveraged to provide more interesting services ?
* Support for full-text retrieval

Support for protocol adoption:

* What technical support is necessary to support protocol adoption ?
* Controlled vocabularies: rights management
* Replication for high-visibility archives

# List of Participants

1. Micah Beck — I2-DSI, University of Tennessee
2. Dennis Boone — Michigan State University
3. Tim Cole — University of Illinois at Urbana-Champaign
4. Tim Collins — OpenText Project
5. Bob Donahue — Department of Energy
6. Edward Fox — Virginia Tech
7. Michael Genuardi — NASA Center for AeroSpace Information
8. Christopher Gutteridge — E-Prints, University of Southampton
9. Thomas G. Habing — University of Illinois at Urbana-Champaign
10. Martin D. Halbert — Emory University
11. Christopher McGiffen — University of Aberdeen
12. Scott Pennington — Michigan State University
13. Faye Phillips — Louisiana State University
14. Thorsten Schwander — Los Alamos National Laboratories
15. Elaine Smyth — Louisiana State University
16. Heinrich Stamerjohanns — University of Oldenburg
17. Hussein Suleman — Virginia Tech
18. Ross Wilkinson — CMIS, Carlton

# Position Statements

## *Cole, Timothy W.*

Mathematics Librarian & Associate Professor of Library Administration
University of Illinois at Urbana-Champaign

**Past / Future Involvement in OAI:**

We began working with OAI metadata harvesting protocol in October of 2000 and participated in alpha testing of the protocol (as a metadata provider) through the end of that test phase earlier this year. Experience to date has demonstrated to us the applicability of the OAI metadata harvesting protocol to local DL projects encompassing a diverse range of disciplines and content types. Recently we have undertaken to implement an OAI metadata harvesting service that will harvest metadata describing digital cultural heritage collections. This work will be funded by a grant from the Andrew W. Mellon Foundation. We will be collaborating with the University of Michigan in developing generic OAI metadata harvesting and provider components (which will be made available under Open Source license). We also are interested (though less involved so far) in other aspects of OAI community activities — notably OAI discussions regarding other cross-repository federation and interoperability services (e.g., sharing of full-text) and standards and best practices for ensuring long-term availability of online scholarly information.

**Position Statement on Workshop Topics:**

Beyond dealing with OAI metadata harvesting technical implementation details that need to be further tested and refined (see, for example, issues raised in position paper submitted by Tom Habing), we need to identify and explore the nature of useful services that can be built on top of OAI metadata harvesting implementations.

- What kinds and ranges of search services can be implemented using OAI harvested metadata?

- To what extent do the harvesting system parameters and provider selection criteria used impact on the possible scope and quality of search services?

- What kinds of useful value-added features can be / should be implemented by metadata harvesting services?

- What changes at the metadata provider service level have the most leverage to improve quality of metadata harvesting services offered?

We also need to explore more focused issues as they relate to the above questions and to plans for future OAI services and standards. These will come up in the context of several of the specific Workshop topics planned.

For instance, in terms of cross-archive services, is there a need to define OAI brokering or proxy protocols to facilitate harvesting and/or archiving of metadata (and/or full-content) from small data providers? While the OAI metadata harvesting protocol is (and should remain) lightweight, there will be institutions with interesting data that are too small to implement and register an OAI provider service open to all comers. Such brokering or proxy layers could help include these smaller institutions in OAI. Or perhaps the OAI registration service could be improved to better describe provider services, including limitations of those services? Or will these concerns be obviated by the emergence of middle level aggregators (e.g., state-based metadata aggregators like the Colorado Digitization Project, or potential discipline-based metadata aggregators like CIMI)? What additional issues arise when harvesting metadata from brokers or aggregators (e.g., record de-duplication, metadata IP issues)?

In terms of community interoperability standards, how useful will the simple DC metadata format prove for building robust and useful search services to enable discovery of hard to find scholarly information resources? If a given provider is sufficiently consistent in use of DC, is it worth it (appropriate) for harvesting services to add structure to harvested records from that provider through customized processing? Alternatively, will there be enough metadata available from providers in more structured formats (e.g., Qualified DC in RDF, EAD, FGDC, etc.) to support OAI harvesting services that rely on these formats instead of simple DC? How can we integrate metadata that is not easily transformed to DC (e.g., data about scientific specimen objects that are not "document-like")?

In discussions of research issues and internal community standards and best practices, how sensitive is quality of OAI-based metadata search services to variations in consistency of community metadata practices? What kinds of value-added content & query refinements (e.g., vocabulary switching, automatic indexing, search term expansion, query or metadata translation) should be / can be added at harvest service level?

The list of questions raised above grows out of our OAI experiences so far, and my contributions to discussions of these questions and others will be based on results from our work here at Illinois up to the time of the Workshop. Our experience to date using OAI for a diverse range of content informs a broad perspective as an OAI metadata provider. By September that experience will be augmented with early experience as an OAI harvester.

**Relevant prior work:**

Illinois participation in the NSF-funded DLI I project (1994 – 1998) and follow-on Illinois participation in the CNRI-sponsored D-Lib Test Suite project (1998 – 2001) have allowed us to look at a wide-range of issues relating to federation of sci-tech literature, the use of XML and related technologies in a DL context, and the implementation of various metadata schemas and interoperability approaches.

As part of this work, a DL Testbed containing digital full content of published journal articles in physics and engineering was begun at Illinois in 1995 and continues to the present. Currently the Illinois Testbed contains 65,000 articles. Item-level metadata for all articles in the Testbed has been maintained throughout. These metadata records are extensive and well structured, containing links both to articles cited by and articles citing the item described. Testbed articles are maintained in XML, and both content and metadata (also in XML) is routinely manipulated using XSLT.

To explore cross-repository searching and federation issues, we established in 2000 a cross-repository metadata search system prototype using metadata harvested from other testbeds in the D-Lib Test Suite as well as our own. Though metadata currency has not been maintained in this prototype, this work also has given us a head start on many of the issues that are now coming to the fore at this stage in the development of the OAI.

## Donohue, Bob

Department of Energy
Office of Scientific and Technical Information

**Planned involvement in OAI**

The Department of Energy (DOE) Office of Scientific and Technical Information (OSTI) has established an OAI compliant repository, greatly facilitated by software developed by the Electronics and Computer Science Department at the University of Southampton, to facilitate collaboration in the National Science Foundation Science Mathematics Engineering and Technology Education Digital Library. This is the collection project 'SITE for Science' for the National Science, Mathematics, Engineering, and Technology Education Digital Library (NSDL) Program funded by the National Science Foundation (NSF). OSTI will provide future NSDL users with a major source of electronic access to scientific and technical information through PubSCIENCE (http://www.osti.gov/pubscience). As you may know PubSCIENCE is a metadata compendium, with links to

full-text, of some of the most prestigious peer-reviewed scientific journal publishers in the world; it includes over 1000 titles from 46 publishers located across the United States and Europe. As a World Wide Web application, it was designed and developed by OSTI to facilitate the accessibility of peer-reviewed journal literature in the physical sciences and areas of research of interest to the Department of Energy. Currently there are 1.4 million citations available in PubSCIENCE and it is growing weekly.

**Contribution to the meeting topics and background**

In depth knowledge of PubSCIENCE, content and how it works. Our experience with establishing our OAI compliant repository, the NSDL NSF project and other issues regarding Dublin Core and XML.

## Fox, Edward A.

I have worked with OAI since the founding meeting in 1999, serving on the Steering Committee, and helping run a number of workshops to promote discussion and dissemination (e.g., June and September 2000). I serve as PI on two NSF-funded projects related to OAI (see www.nudl.org), and as OAI consultant on the SOLINET project funded by the Mellon Foundation. Through four projects connected with NSDL (www.nsdl.nsf.gov), as well as with NDLTD (www.ndltd.org), I'm involved in OAI use to support education in a variety of communities (e.g., related to computing education, electronic theses and dissertations). In conjunction with ODU, UNC-CH, and UVA I'm involved in applying OAI to NCSTRL. All of these efforts involve the very capable group in Virginia Tech's Digital Library Research Laboratory (www.dlib.vt.edu), which has a number of researchers working on a variety of studies related to OAI (including its connection with Internet2 Distributed Storage Initiative).

## Genuardi, Michael

NASA Center for AeroSpace Information

I am affiliated with NASA through the Center for AeroSpace Information (CASI). CASI is currently providing support to an Open Archives project being conducted by the NASA Langley Research Center. We are responsible for data compiling and conversion, and metadata creation.

My objective in attending the Workshop is primarily 'fact gathering' and the opportunity to hear the experiences of those currently participating or planning to participate in the Open Archives Initiative.

## Habing, Thomas G.

Research Programmer
Grainger Engineering Library Information Center
University of Illinois at Urbana-Champaign

### Past / Future Involvement in OAI:

We began working with OAI metadata harvesting protocol in October of 2000 and participated in alpha testing of the protocol (as a metadata provider) through the end of that test phase earlier this year. Experience to date has demonstrated to us the applicability of the OAI metadata harvesting protocol to local DL projects encompassing a diverse range of disciplines and content types. Recently we have undertaken to implement an OAI metadata harvesting service that will harvest metadata describing digital cultural heritage collections. This work will be funded by a grant from the Andrew W. Mellon Foundation. We will be collaborating with the University of Michigan in developing generic OAI metadata harvesting and provider components (which will be made available under Open Source license). We also are interested (though less involved so far) in other aspects of OAI community activities — notably OAI discussions regarding other cross-repository federation and interoperability services (e.g., sharing of full-text) and standards and best practices for ensuring long-term availability of online scholarly information.

**Position Statement on Workshop Topics:**

My experience with the OAI has been mostly technical. That experience suggests that the time is ripe to consider and refine a number of technical issues having to do with the OAI metadata harvesting protocol:

Various technical architectures for both providers and harvesters are possible. Are there optimal configurations? For example, should data storage be based on the file system, a relational database, some combination, or other full-text or object-oriented database?

Strategies for optimizing harvesting throughput for both providers and harvesters need investigation. These include flow control and resumption tokens, plus harvesting strategies such as scheduling complete versus differential harvests, and dealing with latencies resulting from the one-day resolution of timestamps.

Developing turn-key provider implementations which can easily be adopted by smaller, less technically adept organizations should be a priority. It should be easy to insert an OAI provider layer into an existing system with minimal impact to the existing system. An example that should be explored is an OAI provider which sits on top of existing Z39.50 implementation. Another example would be an OAI provider which transforms metadata which is already available as HTML into standard OAI formats.

Support for metadata schemes beyond the simple Dublin Core, especially those based on multiple namespaces (application profiles) needs better support. We have done some work in developing XML Schemas and RDF Schemas which can be used for this purpose.

There are also issues related to brokering service providers, such as data synchronization, dealing with overlapping sets, and multiple metadata formats.

**Relevant prior work:**

Illinois participation in the NSF-funded DLI I project (1994 – 1998) and follow-on Illinois participation in the CNRI-sponsored D-Lib Test Suite project (1998 – 2001) have allowed us to look at a wide-range of issues relating to federation of sci-tech literature, the use of XML and related technologies in a DL context, and the implementation of various metadata schemas and interoperability approaches.

As part of this work, a DL Testbed containing digital full content of published journal articles in physics and engineering was begun at Illinois in 1995 and continues to the present. Currently the Illinois Testbed contains 65,000 articles. Item-level metadata for all articles in the Testbed has been maintained throughout. These metadata records are extensive and well structured, containing links both to articles cited by and articles citing the item described. Testbed articles are maintained in XML, and both content and metadata (also in XML) is routinely manipulated using XSLT.

To explore cross-repository searching and federation issues, we established in 2000 a cross-repository metadata search system prototype using metadata harvested from other testbeds in the D-Lib Test Suite as well as our own. Though metadata currency has not been maintained in this prototype, this work also has given us a head start on many of the issues that are now coming to the fore at this stage in the development of the OAI.

## *Halbert, Martin*

Director for Library Systems
Emory University

I'm the director for library systems here at Emory. We were just awarded a grant by the Mellon Foundation for a 2 year OAI metadata harvesting project. I am very interested in building up my knowledge and acquaintances in the emerging OAI community, and this workshop seems like the right way to go about that.

### *Phillips, Faye and Elaine Smyth*

**Planned involvement:**

The LSU Libraries has not yet become involved in the Open Archives Initiative but several on-going projects as well as upcoming projects are behind our motivation to participate. This year LSU will begin an electronic theses and dissertations program. In conjunction with maintaining the "archive" of those ETDs we are concerned with providing broader access through participation in the Open Archives Initiative. The LSU Libraries Special Collections is the division in which University Archives resides and University Archives is responsible for the permanent preservation of LSU theses and dissertations. Working in conjunction with LSU Computing Services we must develop a program to continue permanent preservation of these materials. In August the LSU Libraries will begin participation with SOLINET on a project entitled "AmericanSouth.org", which will use the Open Archives Initiative protocol to harvest metadata from seven partner libraries and to link their full text digital collections. This SOLINET project is funded through the Mellon Foundation. We also hope to utilize OAI for our current Institute for Museum and Libraries Services digital library grant and other digital library past projects, as well as digital projects in the LSU Libraries Special Collections. See: http://www.lib.lsu.edu/special and http://www.lsu.edu/diglib

**Position statement:**

The LSU Libraries Special Collections began the first digital library project on the LSU campus in 1991 with the creation of e-text files of rare books. Since 1991 we have worked with numerous other units on campus to create other e-text files, digital image libraries, and professors' desktop digital research files. We have created a rich resource of unique materials for which we wish to provide broader access and believe we can utilize OAI for that purpose. In 1995 the LSU Digital Library was established and built on the work that the LSU Libraries Special Collections had begun. Off-campus partners included the University of New Orleans, the Louisiana State Museum, Tulane University, and the New Orleans Public Library. On-campus partners include the University Archives, the Department of Art, the Department of Geography and Anthropology, the Coastal Studies Institute, the Department of History, and the Graduate School. We are involved with the ETD project at LSU and will be involved in a partnership with SOLINET this fall related to harvesting the metadata of digital libraries at seven partner sites.

Because of our experience we can contribute to the discussions at the meeting regarding creating successful partnerships, preparing appropriate metadata for digital collections, reviewing the quality and technical standards for digital libraries, building digital library projects, developing awareness of problems and economies of scale for digital projects and access to the content within those digital projects.

Elaine Smyth, Curator of Special Collections, LSU Libraries and Faye Phillips, Associate Dean of Libraries for Special Collections, LSU Libraries, will be the workshop participants. As Senior Curator of Special Collections, Elaine Smyth is responsible for the collection development and management of preservation and access to the collections. Her Assistant Curators and staffs provide access to collections ranging from the 14th century into the 21st century covering subjects such as the history and culture of the Louisiana and Lower Mississippi Valley region, LSU history, Louisiana literature, Louisiana state government documents, Louisiana politics, eighteenth-century British history and literature, new World exploration and travel, economic history, fine printing and book design, and world natural history. Formats of these collections include manuscripts, university archives, rare books, maps, photographs, art, periodicals, prints, pamphlets, electronic resources, and films and videos. Smyth makes decisions about standard library cataloging, digital library metadata, Web access, and application of new access protocols such as EAD and OAI. Smyth has over twenty years experience as a teacher, rare book cataloger, editor, grants writer and special collections administrator. She has worked at Cornell University, the LSU Press, and for private rare book dealers and collectors. She serves as the Project Manager for our current IMLS grant project.

Faye Phillips, Associate Dean of Libraries for Special Collections, was the initiator of the first LSU digital library project, has written successful grants for five digital library grants, and was one of the founders of the LSU Digital Library. As Associate Dean she holds administrative responsibility for all special collections activities, including facilitating access to the collections. Phillips has thirty years experience as an archivist,

librarian, and special collections administrator. She has served as a consultant for SOLINET digital library workshops, to the Library of Congress American Memory grants program, and to the National Endowment for the Humanities. Phillips has worked at the United States Senate, the National Archives, the University of North Carolina at Chapel Hill, and Georgia State University. She serves as the Project Director for our current IMLS grant project and is currently Chair of the LSU Council on Research.

**Others who might be applying to attend with whom community discussions will proceed:**

Tulane University, University of New Orleans, Auburn University, University of Tennessee, University of Georgia, Emory University, Georgia Department of Archives and History, University of Southern Mississippi and University of North Carolina at Chapel Hill.

## Schwander, Thorsten

arXiv Admin
Research Library
Los Alamos National Laboratories

I am one of the two active maintainers of the LANL arXiv e-print service. The other one is Simeon Warner. As recently announced the Los Alamos e-print arXiv is moving to Cornell this fall. Nevertheless arXiv will remain a cooperative effort between LANL and Cornell. I will remain at LANL, and will continue to provide technical and programming support for arXiv. However being freed of daily maintenance work I will finally have time to more actively participate in OAI implementation and technical discussion which I have been passively following since its inception. I participated in the pre-release workshop at Cornell last fall and have been working with various service providers harvesting arXiv on improving their algorithms.

The Library Without Walls project at Los Alamos provided the cradle for OAI while van de Sompel was visiting in '99 and the team here remains highly interested in development and deployment of interoperability protocols and infrastructure. While arXiv is moving to Cornell this fall I will remain at the Los Alamos research library (upstate NY can't lure me away from the New Mexico sun and Colorado mountains) and will shift my focus from daily arXiv maintenance to more genuine development work on arXiv and other digital library projects.

In particular I will be involved with the Technical Report Initiative spearheaded by LANL and ODU and will provide an OAI interface for the report collection in collaboration with ODU. I also intend to implement OAI interfaces for all other LANL library collections and resources which are not access restricted.

I look forward to discussion of problems with existing implementations, interpretation and clarification of the protocol specifications based on real life examples and in particular want to point out some shortcomings of current harvesting procedures as observed from the data provider perspective.

## Smyth, Elaine

See Faye Phillips

## Stamerjohanns, Heinrich

University of Oldenburg
Physics department

**Summary of prior work in OAI:**

I have presented a paper about "Distributed Archives and OAI" at the OAI Open Meeting in Berlin on February 26th and have also given talks on two meetings by the German DINI (the German initiative for networked information, www.dini.de) which were focused on implementing OAI at libraries.

I have implemented an OAI Data Provider for PhysDoc, a repository for physics documents. I am currently developing a subject-specific (physics) OAI Service Provider, which will not only collect data from other data providers but will search through other metadata collections which are not otherwise publicly available.

**Position statement:**

My contribution to the workshop can be my long experience with harvesting metadata from distributed archives, which we have been doing since 1995 with PhysDoc (http://physnet.uni-oldenburg.de/PhysNet).

I can explain how we have set up a Data Provider for this heterogeneous data and explain specific problems.

We are currently developing (in PHP) a Service Provider. I can contribute first experiences on becoming a service provider.

Besides our focus on the physics community we help libraries and other institutions to become OAI Data providers. University libraries will make especially dissertations available. A cross-archive service for dissertations will be set up at our Institute in the near future.

I have graduated in Physics in 1997 at the University of Oldenburg, Germany and am currently working on my Ph.D.

Besides my physics work, I have been involved in the development of the World Wide Web since 1993. I have done the initial work for PhysNet and PhysDoc in 1995 and have then (until end of 2000) worked in the German Digital Initiative "Global-Info".

Since 2001, I have been working in a joint DFG/NSF project on "Open Archives: Distributed services for physicists and graduate students (OAD)".

## Suleman, Hussein

Digital Library Research Laboratory
Virginia Tech

I am a PhD student at Virginia Tech, studying issues that arise from interoperability efforts in digital libraries. I currently work on the Computer Science Teaching Center (http://www.cstc.org) and, specifically, its federation mechanisms with other systems such as iLumina (http://www.ilumina-project.org). My research focuses on novel uses of OAI to support componentized digital libraries as a natural extension of and self-sustaining mechanism for interoperability.

I have worked closely with the Open Archives Initiative during their recent development effort that led to the Metadata Harvesting Protocol v1.0, and I serve as a member of the OAI Technical Committee. I actively

develop and support the OAI Repository Explorer (http://purl.org/net/oai_explorer/), a validation tool that assists developers in creating OAI-compliant archives.

## *Wilkinson, Ross*

CSIRO, Division of Mathematical and Information Science

My reasons for interest are 3-fold:

1. We have spent some time working with Government Archives in Australia, on developing an approach to capturing enduring electronic records using XML, PDF, and metadata standards derived from the archiving community (Bearman, et. al.)- ref: http://www.prov.vic.gov.au/vers/welcome.htm

2. At CSIRO, a government research organisation in Australia, we are seriously investigating using an OAI server to manage our publications.

3. I am interested in the research questions surrounding the use and query mechanisms associated with a rich data set that has both content and metadata available.

# The Open Archives Initiative Protocol for Metadata Harvesting

**Protocol Version 1.1 of 2001-07-02**
**Document Version 2001-06-20**
**http://www.openarchives.org/OAI/openarchivesprotocol.htm**

**Previous version: Protocol Version 1.0 of 2001-01-21
Instructions for migrating from Version 1.0 to 1.1**

**Editors**

Herbert Van de Sompel <herbertv@cs.cornell.edu> -- Cornell University - Computer Science
Carl Lagoze <lagoze@cs.cornell.edu> -- Cornell University - Computer Science

**Table of Contents**

# 1. Introduction

The goal of the Open Archives Initiative Protocol for Metadata Harvesting (referred to as the OAI protocol in the remainder of this document) is to supply and promote an application-independent interoperability framework that can be used by a variety of communities who are engaged in publishing content on the Web.   The OAI protocol described in this document permits *metadata harvesting.* The result is an interoperability framework with two classes of participants:

- *Data Providers* administer systems that support the OAI protocol as a means of exposing metadata about the content in their systems;

- *Service Providers* issue OAI protocol requests to the systems of data providers and use the returned metadata as a basis for building value-added services.

# 2. Definitions and Concepts

## *2.1 Repository*

**A *repository* is a network accessible server to which OAI protocol requests, embedded in HTTP, can be submitted.**  The OAI protocol provides access to metadata from OAI-compliant repositories.  This metadata is output in the form of a record.  A record is the result of a protocol request issued to the repository to disseminate metadata from an *item*. An *item* can be considered a constituent of the repository.  The  nature of an item -- for example, what type of metadata is actually stored in the item, what type is derived on-the-fly,  and whether the item includes the "full content" described by the metadata -- is outside the scope of the OAI protocol.  All uses of the term *repository* in the remainder of this document should be interpreted as an *OAI-compliant repository*.

## *2.2 Record*

**A *record* is an XML-encoded byte stream that is returned by a repository in response to an OAI protocol request for metadata from an item in that repository.**  OAI records -- referred to for the remainder of this document as *records* -- are organized into the following parts:

- *header* -- information that is common to all records and that is necessary for the harvesting process. This information is independent of the metadata format disseminated in the record. The header consists of two parts:

  o the unique identifier -- the key for extracting metadata from an item in a repository;

  o the datestamp -- the date of creation, deletion, or latest date of modification of an item, the effect of which is a change in the metadata of a record disseminated from that item.

- *metadata* -- a single manifestation of the metadata from an item.  The OAI protocol supports multiple manifestations (formats) of metadata for any single item.  At a minimum, repositories must be able to return records with metadata expressed in the Dublin Core format, without any qualification. Optionally, a repository may also be capable of disseminating other formats of metadata.  The specific manifestation that

should be returned in a record is specified by means of an argument -- the metadata prefix -- in the GetRecord or ListRecords protocol request that produces the record. The ListMetadataFormats protocol request returns the list of metadata formats available from a repository, or from a specific item (which can be specified as an argument to the ListMetadataFormats service request).

- *about* -- an optional container to hold data about the metadata part of the record. Typically, the *about* container could be used to hold rights information regarding the metadata part of the record, terms and conditions for usage of the metadata part of the record, etc. The internal structure of the about container is not defined by the protocol. It is left to individual communities to decide on its syntax and semantics.

The example shows a record and its components:

- the *header* part with:

  o a unique identifier equal to `oai:arXiv:quant-ph/9901001`;

  o a datestamp equal to `1999-01-01`;

- the *metadata* part expressed in unqualified Dublin Core. Note that the root element within the metadata part includes three attributes:

  o `xmlns` -- the value of which is the namespace URI (as defined by the XML namespace specification) of the metadata that follows in this part.

  o `xmlns:xsi` -- the value of which must always be the URI shown in the example, which is the namespace URI for XML schema.

  o `xsi:schemaLocation` -- the value of which is a pair, the first part is the namespace URI (as defined by the XML namespace specification) of the metadata that follows in this part, and the second part is the URL of the XML schema for validation of the metadata.

- the *about* part of the record which uses a -- fictional -- eprints-about.xsd schema as a means to express properties of the metadata part of the record. Note that the root element within the about part includes three attributes:

  o `xmlns` -- the value of which is the namespace URI (as defined by the XML namespace specification) of the XML elements that follow in this part.

  o `xmlns:xsi` -- the value of which must always be the URI shown in the example, which is the namespace URI for XML schema.

  o `xsi:schemaLocation` -- the value of which is a pair, the first part is the namespace URI (as defined by the XML namespace specification) of the elements that follow in this part, and the second part is the URL of the XML schema for validation of the about part.

```
<header>
 <identifier>oai:arXiv:quant-ph/9901001</identifier>
 <datestamp>1999-01-01</datestamp>
</header>
<metadata>
 <dc xmlns="http://purl.org/dc/elements/1.1/"
     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://purl.org/dc/elements/1.1
http://www.openarchives.org/OAI/1.1/dc.xsd">
   <title>Quantum slow motion</title>
   <creator>Hug, M.</creator>
```

```
      <creator>Milburn, G. J.</creator>
      <description>We simulate the center of mass motion of cold
atoms in a
        standing, amplitude modulated, laser field as an example of
a system
        that has a classical mixed phase-space.</description>
      <date>1999-01-01</date>
      <type>e-print</type>
      <identifier>http://arXiv.org/abs/quant-
ph/9901001</identifier>
    </dc>
  </metadata>
  <about>
   <ea xmlns="http://www.arXiv.org/eprints-about"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.arXiv.org/eprints-about
http://www.arXiv.org/eprints-about.xsd">
      <archive>The Los Alamos arXiv</archive>
      <usage>Metadata may be used without restrictions as long as
the OAI
        identifier remains attached to it.</usage>
    </ea>
  </about>
```

The protocol supports the notion of a *deleted record*, which may be used by repositories as follows. When returning a harvested record or identifier of a record, the `ListRecords` and `ListIdentifiers` service requests may indicate a status of "deleted". This status means that an item has been deleted and therefore no record can be disseminated from it. The length of time that a given repository keeps track of deleted items is not defined by the protocol. Therefore, the only *guaranteed* method in the protocol for determining whether a record can be returned by a repository (its corresponding item still exists) is through the `GetRecord` service request.

## 2.3 Unique Identifier

**A *unique identifier* is a key for extracting metadata from an item in a repository.** The OAI protocol specifies that each item may disseminate metadata in multiple formats. All of the members of this group of possible metadata formats disseminated from a single item share the same unique identifier. Each record disseminated by a `GetRecord` or `ListRecords` protocol request is identified by the combination of this unique identifier and a metadata prefix, which identifies the metadata format.

The format of unique identifiers corresponds to that of the URI (Uniform Resource Identifier) syntax. An individual community may develop a URI scheme for coordinated use across its repositories. For an example, see the format for identifiers proposed in Appendix 2.

These unique identifiers play two roles in the protocol:

1.  Identifiers are returned by both the `ListIdentifiers` and `ListRecords` requests; they can be used as keys to request metadata from items.

2.  An identifier, in combination with a metadata prefix, is used in the `GetRecord` protocol request as a means of requesting a record with a specific metadata format from an item. This means that the resulting record is identified by the combination of the identifier and the metadata prefix.

Note that the identifier described here is *not* that of the item. The nature of the item identifier is outside the scope of the OAI protocol.

Since many clients of the OAI protocol may want to access the content associated with harvested metadata, it is highly recommended that repositories use an element in metadata records to establish a linkage between the record (and its identifier) and the identifier (URL, URN, DOI, etc.) of the associated item. The mandatory Dublin Core format provides the `identifier` element that can be used for this purpose.

## 2.4 Datestamp

**A *datestamp* is the date of creation, deletion, or latest date of modification of an item, the effect of which is a change in the metadata of a record disseminated from that item.** Every record returned by the GetRecord and ListRecords service requests contains its respective datestamp in the header. Datestamps are used as values of optional arguments -- `until` or `from` -- to the ListRecords and ListIdentifiers requests to specify selective harvesting (by date) from the repository. The value of a datestamp must comply to the specifications for dates in the OAI protocol.

## 2.5 Set

**A *set* is an optional construct for grouping items in a repository for the purpose of selective harvesting of records.** Each repository may define a hierarchical organization of items; the hierarchy can have several top-level nodes. Each node in the hierarchy is a Set, which has:

- a `setTag` -- a non-space separated string of alphanumeric characters;

- a `setSpec` -- a colon [:] separated list of `setTags` of each node on the path leading from a root element to the actual node;

- a `setName` -- a string useful for display purposes.

The following is an example of a possible Set hierarchy in a repository:

- Institutions
  - o Oceanside University of Nebraska
  - o Valley View University of Florida
- Subjects
  - o Existential Kenesiology
  - o Quantum Psychology

The following table shows a representation of the above Set hierarchy by means of `setNames`, `setTags` and `setSpecs` :

| setName | setTag | setSpec |
|---|---|---|
| Institutions | institution | institution |

| | | |
|---|---|---|
| Oceanside University of Nebraska | `nebraska` | `institution:nebraska` |
| Valley View University of Florida | `florida` | `institution:florida` |
| Subjects | `subject` | `subject` |
| Existential Kenesiology | `kenesiology` | `subject:kenesiology` |
| Quantum Psychology | `quantum` | `subject:quantum` |

Each item in a repository may be organized in one Set, several Sets, or no Sets at all. A client should not assume that harvesting every Set in a repository will retrieve records for every item in the repository. Items may also be organized in interior nodes in the Set hierarchy. Thus, in the example above, it is conceivable that an individual item is organized in both `subject` and `florida`.

The actual meaning of a Set or of the arrangement of Sets in a repository is not defined in the protocol, nor is there any way to derive this meaning through the protocol. It is expected that individual communities who use the OAI protocol may formulate well-defined Set configurations with perhaps a controlled vocabulary for Set names, and may even develop mechanisms for exposing these to service providers. For example, a group of cooperating e-print archives in a specific discipline may agree on Sets that arrange items in their repositories based on a controlled subject classification. Note that there is no mechanism for "reverse lookup"; that is, determining the Set or Sets that an item -- from which a record was disseminated -- is organized in.

A repository's Set hierarchy is represented in the protocol via `setSpecs`. `ListSets` returns the configuration of Sets in a repository as a list of `setSpecs`. Any of these `setSpecs` can be used as optional arguments to `ListRecords` or `ListIdentifiers` to specify the target Set for selective harvesting from the repository. In the previous example of a Set hierarchy, the `setSpec institution:nebraska` can be used in a protocol request to return only those records that are disseminated from items organized in the Set represented by this `setSpec`. Three issues should be noted here:

- As described above an individual item may be organized in more than one Set; meaning that different `setSpec` arguments may return the same record(s).

- As described above an individual item may not be organized in any Set; meaning that an exhaustive repetition of `ListRecords` requests with all possible `setSpecs` will *not* return all records in the repository. The only guaranteed method of returning all records is a `ListRecords` or `ListIdentifiers` request with no `setSpec` argument.

- Using a `setSpec` as an argument in a protocol request will return:

  o all records from items in the Set specified by the `setSpec`,

  o all records from items in Sets that are descendent from the Set specified by the `setSpec`.

Using the previous example of a Set hierarchy, a `setSpec` of `institution` to the `ListRecords` protocol request will return all records from items organized within the `institution` Set and within the descendent Sets `florida` and `nebraska`.

# 3. Protocol Features

## 3.1 HTTP embedding of OAI requests

OAI protocol requests are expressed as HTTP requests. A typical implementation uses a standard Web server, such as Apache, that is configured to dispatch OAI requests to the software handling these requests. The remainder of this section describes the aspects of the protocol that are specific to the HTTP embedding.

### 3.1.1 HTTP Request Format

OAI protocol requests may be expressed using either the HTTP `GET` or `POST` methods. There is a single base URL, `BASE-URL`, for all requests. At a minimum, the `BASE-URL` specifies the Internet host and port of an HTTP server acting as a repository. Typically, the `BASE-URL` will also specify a *path* that is specified by the respective HTTP server as the handler for OAI protocol requests. Note that the composition of this path and its presence or absence is determined by the configuration of the repository's HTTP server.

In addition to the `BASE-URL`, all requests consist of a list of *keyword arguments*, which take the form `key=value` pairs. Arguments may appear in any order and multiple arguments should be separated by ampersands [`&`]. At a minimum, each OAI protocol request has one `key=value` pair that specifies the OAI protocol request issued by the client:

- `key` is the string `'verb'`;

- `value` is one of the defined OAI protocol requests.

The number and nature of additional `key=value` pairs is dependent on the arguments for the individual protocol request.

Libraries such as `libwww-perl` will decode `GET` and `POST` requests seamlessly; the application does not even need to know which method was used. The advantage of `POST` is that there are no limitations on the length of the arguments.

#### 3.1.1.1 Encoding an OAI protocol request in a URL for an HTTP GET

URLs for `GET` requests have keyword arguments appended to the `BASE-URL`, separated by a question mark [`?`]. For example, the URL of a `GetRecord` protocol request to a repository with `BASE-URL` that is `http://an.oa.org/OAI-script` might be:

```
http://an.oa.org/OAI-script?verb=GetRecord&identifier=oai:arXiv:hep-
th/9901001&metadataPrefix=oai_dc
```

However, since special characters in URIs must be encoded, the valid form of the above above `GET` request URL is:

```
http://an.oa.org/OAI-script?verb=GetRecord&identifier=oai%3AarXiv%3Ahep-
th%2F9901001&metadataPrefix=oai_dc
```

#### 3.1.1.2 Encoding an OAI protocol request in an HTTP POST

The keyword arguments for an OAI protocol request are carried in the message body of the HTTP POST. The `Content-Type` of the request should be `application/x-www-form-urlencoded`. For example,

submitting the same request as above using the `POST` method would use just `BASE-URL` as the URL, with the format of the `POST` being:

```
POST http://an.oa.org/OAI-script HTTP/1.0
Content-Length: 78
Content-Type: application/x-www-form-urlencoded
verb=GetRecord&identifier=oai%3AarXiv%3Ahep-
th%2F9901001&metadataPrefix=oai_dc
```

**3.1.1.3 Encoding of special characters in keyword arguments of OAI protocol requests**

The syntax rules for URIs restrict a few characters to special roles in certain contexts, and require that if these characters are used in any other way that they be written as an escape sequence, i.e. a percent sign followed by the character code in hexadecimal. The reserved characters include:

| *Character* | *URI Role* | *Escape Sequence* |
|:---:|:---:|:---:|
| / | Path Component Separator | `%2F` |
| ? | Query Component Separator | `%3F` |
| # | Fragment Identifier | `%23` |
| = | Name/Value Separator | `%3D` |
| & | Argument Separator in Query Component | `%26` |
| : | Host Port Separator | `%3A` |
| ; | Authority Namespace Separator | `%3B` |
| '' | Space Character | `%20` |
| % | Escape Indicator | `%25` |
| + | Escaped Space | `%2B` |

As a result, these characters must be represented by their respective escape sequence if their use does not correspond to their established URI role.  In case of the OAI protocol, this means that the reserved characters have to be encoded when they in the `value` part of the `key=value` pairs of which the list of keyword arguments of the OAI protocol request consists. This applies for both the `GET` and `POST` encoding of the OAI protocol requests.

## 3.1.2 HTTP Response Format

Responses to protocol requests are formatted as HTTP responses, with appropriate HTTP header fields. Every OAI protocol request returns a `Content-Type` of `text/xml`.  Similarly, the success or error status of each protocol request is returned via the HTTP `Status-Code`.

**3.1.2.1 Content-Type**

The `Content-Type` returned for all OAI protocol requests is `text/xml`.  Encoding of the XML is done using the UTF-8 representation of Unicode, whereby:

- The use of character references is preferred over the use of entity references.  Character references allow XML responses to be treated as stand-alone documents that can be manipulated without dependency on entity declarations external to the document.

- The use of a hexadecimal representation of character references is preferred over decimal representations, since Unicode character encodings are defined using hex numbers. For instance, `&#x00A9;` is preferred over `&#169;` for encoding © .

## The XML data for all responses has the following common markup:

1. The first tag output is an XML declaration where the `version` is always `1.0` and the `encoding` is always `UTF-8`, eg: `<?xml version="1.0" encoding="UTF-8" ?>`

2. The remaining content is enclosed in a root element that has the same name as the verb of the respective OAI protocol request. As shown, this element has three attributes:

   o `xmlns` -- the value of which is the namespace URI (as defined by the XML namespace specification) of the actual OAI protocol request (refer to the documentation of each protocol request for that URI).

   o `xmlns:xsi` -- the value of which must always be the URI shown the example, which is the namespace URI for XML schema.

   o `xsi:schemaLocation` -- is a pair, the first part is the namespace URI (as defined by the XML namespace specification) of the protocol request, and the second part is the URL of the XML schema for validation of the response.

3. For all responses, the first two children of the root element are:

   o `responseDate` -- a timestamp (compliant with the dates and times specifications) indicating the time and date that the response was sent. This *must* be expressed in the local time of the repository.

   o `requestURL` -- the URL for an HTTP `GET` request that could be used to re-submit the protocol request that generated this response. Note that this URL is encoded exactly as in protocol requests. In addition, in the encoded URL all ampersand characters (`&`) that are used to separate arguments must be encoded as `&amp;` in order to obtain valid XML.

For example, the reply to the `GetRecord` protocol request shown above will be of the form:

```
<?xml version="1.0" encoding="UTF-8" ?>
<GetRecord
    xmlns="http://www.openarchives.org/OAI/1.1/OAI_GetRecord"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_GetRecord
                http://www.openarchives.org/OAI/1.1/OAI_GetRecord.xsd">
  <responseDate>2001-06-01T19:20:30-04:00</responseDate>
  <requestURL>http://an.oa.org/OAI-script?verb=GetRecord
            &amp;identifier=oai%3AarXiv%3Ahep-th%2F9901001
            &amp;metadataPrefix=oai_dc</requestURL>
    list of records
</GetRecord>
```

### 3.1.2.2 Status-Code

The OAI protocol uses the HTTP `Status-Code` to indicate the success or nature of the failure of the request. A normal response to an OAI protocol request is signaled with the HTTP `200 Status-Code`. Error returns are signaled with the appropriate `4xx` code as specified in the HTTP protocol. The following HTTP `Status-Code` has a special meaning when returned in the context of responses to OAI protocol requests:

- `400` - a `Status-Code` of `400` is returned when the syntax of the protocol request is illegal, i.e.:

o illegal verbs are used in the request;

o illegal arguments are used in the request.

OAI compliant repositories may also employ other HTTP `Status-Codes` in manners as defined by the HTTP protocol. The following list includes `Status-Codes` that may be useful for load balancing of OAI repositories:

- `302` - Allows the repository to temporarily redirect an OAI protocol request to another repository. The URI of the temporary repository should be given by the `Location` field in the HTTP response.

- `503` - Service unavailable, a `Retry-After` period is specified.  Clients should wait this period before attempting another OAI protocol request (Clients that do not observe the `Retry-After` period may be denied access).

Repositories should use the HTTP `Reason-Phrase` to provide additional information about an error that is useful to a human reader.

## 3.2 Dates and Times

Dates and times are uniformly encoded throughout the protocol.  Their use is as follows:

- *Protocol Requests*: Values for the optional `from` and `until` arguments in [ListIdentifiers](#) and [ListRecords](#) protocol requests, as well as for [datestamps](#) in the headers of records are encoded using the "[Complete date" variant of ISO8601](#). That format is `YYYY-MM-DD` where `YYYY` is the year, `MM` is the month of the year between 01 (January) and 12 (December), and `DD` is the day of the month between 01 and 28 or 29 or 30 or 31, depending on length of month and whether it is a leap year. For example: 1957-03-20 is March 20th 1957.

- *Protocol Responses*:  Each protocol response includes a `responseDate` element, which *must* be the time and date of the response in the *local* time of the repository.  This is encoded using the "[Complete date plus hours, minutes and seconds" variant of ISO8601](#).  That format is `YYYY-MM-DDThh:mm:ssTZD,` with the following meanings:

  o `YYYY, MMM, DD` is the same as for [dates](#).

  o T is the character "T",

  o `hh` are the hours (between 00 and 23), `mm` are the minutes (between 00 and 59) and `ss` the seconds (between 00 and 59).

  o TZD is an indication of the time zone of the repository, expressed as an offset from UTC.  A time zone offset of "+hh:mm" indicates that the date/time uses a local time zone which is "hh" hours and "mm" minutes ahead of UTC. A time zone offset of "-hh:mm" indicates that the date/time uses a local time zone which is "hh" hours and "mm" minutes behind UTC.

  For instance, `1957-03-20T20:30:00+00:00` is UTC 8:30 PM on March 20th 1957.

## 3.3 Metadata Prefix and Metadata Schema

Each metadata format that is included in records disseminated by the OAI protocol is identified within the protocol requests to a repository by a *metadata prefix* and across multiple repositories by the URL of a *metadata schema*:

- The *metadata prefix* must be a non-space embedded string consisting of alphanumeric characters or an underscore [_] character. For the purpose of clarity in this document, all metadata prefixes will be preceded by the string `oai_` (for example, `oai_dc`).

- The *metadata schema* is an [XML schema](#) that may be used as a test of conformance of the metadata included in the record. It is highly recommended that the schema include comments that aid in human understandability.

The metadata formats that are available for a repository (or for a specific identifier) can be obtained through the `ListMetadataFormats` protocol request. For each metadata format that is supported, the response to the `ListMetadataFormats` protocol requests contains the metadata prefix of the format and the URL of the metadata schema for the metadata format. Optionally, this response may also contain the [XML namespace](#) URI associated with the format.

Metadata prefixes are used in the `ListRecords` and `GetRecord` protocol requests, to request records that include metadata in the format identified by the metadata prefix.

The metadata in each [record](#) returned by `ListRecords` and `GetRecord` complies to the conventions of the [XML namespace specification](#). This means that the root element of the metadata part must contain an `xmlns` attribute, the value of which is the namespace URI of the metadata format. The root element should also contain an `xsi:schemaLocation` attribute that has a value that includes the URL of the XML schema for validation of the metadata. This URL should match the URL of the *metadata schema* for the *metadata prefix* included as an argument to the `ListRecords` or `GetRecord` request (the mapping from *metadata prefix* to *metadata schema* is defined by the repository's response to the `ListMetadataFormats` protocol request).

The OAI protocol mandates one *required* metadata format for the purpose of interoperability. This format is [Dublin Core](#), without any [qualification](#). Therefore, the protocol reserves the metadata prefix '`oai_dc`', and the URL of a metadata schema for unqualified Dublin Core, which is [http://www.openarchives.org/OAI/1.1/dc.xsd](http://www.openarchives.org/OAI/1.1/dc.xsd).

Certain communities may adopt guidelines for sharing of metadata prefixes and/or metadata schema throughout the community. Such guidelines are outside of the scope of the OAI protocol.

[Appendix 1](#) provides some sample XML Schema and instance documents for common metadata formats such as Dublin Core, MARC and RFC1807

## *3.4 Flow Control*

A number of OAI protocol requests return a *list* of discrete entities; `ListRecords` returns a list of records, `ListIdentifiers` returns a list of identifiers, and `ListSets` returns a list of `setSpecs`. In some cases, these lists may be large and it may be practical to partition them among a series of protocol requests and responses. This partitioning is accomplished as follows:

- A repository replies to a protocol request with an *incomplete list* and a `resumptionToken;`

- In order to make the response a *complete list*, the client will need to issue one or more protocol requests with `resumptionTokens` as arguments. A complete response then consists of the combination of the *incomplete lists* from the sequence of protocol requests.

The details on flow control and the `resumptionToken` are as follows:

- The only defined use of `resumptionToken` is as follows:

  o a client receives a `resumptionToken` from a repository as part of a reply to a protocol request;

  o the client copies the `resumptionToken` and includes it in a subsequent protocol request.

  All other uses of `resumptionToken` by a client are illegal and hence will return a `Status-Code` of `400` from the repository.

- In all cases when a `resumptionToken` is issued, the incomplete list will always consist of complete entities; e.g., all individual records returned in an incomplete record list from a `ListRecords` request will be intact.

- The format of the `resumptionToken` is undefined by the OAI protocol and should be considered opaque by the client.

- The protocol does not define the semantics of incompleteness. Therefore, a client should not assume that the members in an *incomplete list* conform to some selection criteria (e.g., date ordering).

- The *time-to-live* of an individual `resumptionToken` is undefined by the OAI protocol. It is conceivable that there will be instances when a repository considers an individual `resumptionToken` extinct; i.e. the time span between the issuance of the `resumptionToken` and its use in a subsequent request is "too long". In that case, the request will return a `Status-Code` of `400` (illegal arguments) and the client will need to reinitiate the sequence of requests (and dispose of the existing *incomplete list*). Since the format of a `resumptionToken` is opaque, there is no defined way for a client to determine its *time-to-live*.

- Before including a `resumptionToken` in a subsequent request, a client should encode any [special characters](#) in it.

The use of the `resumptionToken` in the context of a series of [ListRecords](#) requests is as follows:

- The response to a `ListRecords` protocol request consists of a *record list* (a list of records). This *record list* is distinguished as *incomplete* -- in contrast to *complete* -- if the response to the `ListRecords` protocol request includes a `resumptionToken`.

- If a *record list* is incomplete, the client should include the `resumptionToken` as the only argument to a subsequent `ListRecords` request. The response to this request is a continuation of the *incomplete record list* that resulted from the `ListRecords` protocol request that issued the `resumptionToken`. The client may then merge the records from the continuation with the *incomplete record list*.

- This subsequent request may itself result in a reply that contains a `resumptionToken`. The client should then continue to issue `ListRecord` requests using the `resumptionToken` from the previous request, merging returned records with the *incomplete record list* until a reply does no longer contain a `resumptionToken`. The absence of a `resumptionToken` informs the client that all records that make up a *complete record list* have been gathered.

This flow control mechanism, in combination with the [503 Status-Code](#), provides some basic tools with which a repository can enforce an *acceptable use policy* for its harvesting interface. Communities implementing the OAI protocol may need more extensive tools to enforce acceptable use policies for either the harvesting interface of their repositories or for the metadata harvested from those repositories. The enforcement of such additional policies is outside of the scope of the OAI protocol.

# 4. Protocol Requests and Responses

This section lists the protocol requests -- `verbs` -- defined in the Open Archives Initiative protocol. The documentation for each protocol request is organized as follows:

- Each section header corresponds to the token used to specify the request as the required `verb` argument to an HTTP protocol request.

- A brief summary of the meaning of the verb and notes on its usage

- The list of additional arguments for the request. Arguments are of three types:

  - *required,* the argument must be included with the request (the `verb` argument is always *REQUIRED*).

  - *optional,* the argument may be included with the request.

  - *exclusive,* the argument may be included with request, but must be the only argument (except for the `verb` argument).

- The format of the response defined by means of an XML schema.

- Exception conditions for the protocol request. These are in addition to the HTTP `Status-Codes` that have a special meaning in the context of the OAI protocol, which are listed under Status-Codes.

- One or more example requests and corresponding responses, with explanatory notes if appropriate.

## *4.1. GetRecord*

### Summary and Usage Notes

This verb is used to retrieve an individual record (metadata) from an item in a repository. Required arguments specify the identifier, or key, of the requested record and the format of the metadata that should be included in the record

### Arguments

- **identifier** a *REQUIRED* argument that specifies the unique identifier that should be used as a key to extract the requested record from an item in the repository.

- **metadataPrefix** a *REQUIRED* argument that specifies the metadata prefix of the format that should be included in the metadata part of the returned record. The value of this argument must be a non-space embedded string consisting of alphanumeric and underscore [_] characters; [A-Za-z0-9_]. The metadata formats supported by a repository and for a particular record can be retrieved using the `ListMetadataFormats` request.

### Exception Conditions

- *Identifier does not exist* - The response will not contain a record container.

- *Metadata format can not be disseminated for the identifier* - The record in the response contains a header but no metadata container.

### Response Format

A XML schema for validating the response is as follows:

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:oai="http://www.openarchives.org/OAI/1.1/OAI_GetRecord"
targetNamespace="http://www.openarchives.org/OAI/1.1/OAI_GetRecord"
        elementFormDefault="qualified"
        attributeFormDefault="unqualified">

<annotation>
 <documentation>
    Schema to verify validity of responses to GetRecord OAI-protocol request.
    This Schema validated at http://www.w3.org/2001/03/webdata/xsv on 2001-05-
07
    with XSV XSV 1.189/1.95 of 2001/05/07 08:38:12
 </documentation>
</annotation>

<element name="GetRecord" type="oai:GetRecordType"/>

<!-- response to GetRecord-request -->

<complexType name="GetRecordType">
 <sequence>
   <element name="responseDate" minOccurs="1" maxOccurs="1" type="dateTime"/>
   <element name="requestURL" minOccurs="1" maxOccurs="1" type="anyURI"/>
   <element name="record" minOccurs="0" maxOccurs="1" type="oai:recordType"/>
 </sequence>
</complexType>

<!-- define recordType -->
<!-- a record has a header and a metadata part -->

<complexType name="recordType">
 <sequence>
   <element name="header" minOccurs="1" maxOccurs="1" type="oai:headerType"/>
   <element name="metadata" minOccurs="0" maxOccurs="1"
type="oai:metadataType"/>
   <element name="about" minOccurs="0" maxOccurs="1" type="oai:aboutType"/>
 </sequence>
     <attribute name="status" use="optional" type="oai:statusType"/>
</complexType>

<!-- define headerType -->
<!-- a header has a unique identifier and a datestamp -->

<complexType name="headerType">
 <sequence>
   <element name="identifier" minOccurs="1" maxOccurs="1" type="anyURI"/>
   <element name="datestamp" minOccurs="1" maxOccurs="1" type="date"/>
 </sequence>
</complexType>

<!-- define metadataType -->
<!-- metadata must be expressed in XML that is compliant with another XML  -->
<!-- Schema. metadata must be explicitly qualified in the response -->

<complexType name="metadataType">
 <sequence>
```

Page 26

```
  <any namespace="##other" processContents="lax"/>
 </sequence>
</complexType>


<!-- define aboutType -->
<!-- data "about" the record must be expressed in XML -->
<!-- that is compliant with an XML Schema defined by a community -->

<complexType name="aboutType">
 <sequence>
  <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="1"/>
 </sequence>
</complexType>


 <!-- define statusType -->
 <!-- a record can have a status of "deleted" . -->
<simpleType name="statusType">
  <restriction base="string">
   <enumeration value="deleted"/>
  </restriction>
</simpleType>

</schema>
```

This Schema is available at http://www.openarchives.org/OAI/1.1/OAI_GetRecord.xsd

## Examples

**Request**

Request a record in the Dublin Core metadata format [URL shown without encoding for better readability].

```
http://an.oa.org/OAI-script?verb=GetRecord&identifier=oai:arXiv:quant-
ph/9901001&metadataPrefix=oai_dc
```

**Response**

```
<?xml version="1.0" encoding="UTF-8"?>
 <GetRecord
    xmlns="http://www.openarchives.org/OAI/1.1/OAI_GetRecord"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_GetRecord
           http://www.openarchives.org/OAI/1.1/OAI_GetRecord.xsd">
  <responseDate>2001-06-10T14:09:57-07:00</responseDate>
  <requestURL>http://an.oa.org/OAI-script?verb=GetRecord
              &amp;identifier=oai%3AarXiv%3Aquant-ph%2F9901001
              &amp;metadataPrefix=oai_dc</requestURL>
  <record>
   <header>
    <identifier>oai:arXiv:quant-ph/9901001</identifier>
    <datestamp>1999-01-01</datestamp>
   </header>
   <metadata>
   <dc xmlns="http://purl.org/dc/elements/1.1/"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://purl.org/dc/elements/1.1/
                           http://www.openarchives.org/OAI/1.1/dc.xsd">
     <title>Quantum slow motion</title>
     <creator>Hug, M.</creator>
     <creator>Milburn, G. J.</creator>
```

```
      <description>We simulate the center of mass motion of cold atoms in a
standing,
       amplitude modulated, laser field as an example of a system that has a
classical
       mixed phase-space.</description>
      <date>1999-01-01</date>
      <type>e-print</type>
      <identifier>http://arXiv.org/abs/quant-ph/9901001</identifier>
     </dc>
    </metadata>
   </record>
  </GetRecord>
```

**Request**

Request a record in the Dublin Core metadata format. The requested record, however, can not be returned because the identifier does not exist. Therefore, the response does not contain a record container [URL shown without encoding for better readability].

```
http://an.oa.org/OAI-script?verb=GetRecord&identifier=oai:arXiv:quant-
ph/0201001&metadataPrefix=oai_dc
```

**Response**

```
<?xml version="1.0" encoding="UTF-8"?>
 <GetRecord
    xmlns="http://www.openarchives.org/OAI/1.1/OAI_GetRecord"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_GetRecord
              http://www.openarchives.org/OAI/1.1/OAI_GetRecord.xsd">
  <responseDate>2001-06-10T14:09:57-07:00</responseDate>
  <requestURL>http://an.oa.org/OAI-script?verb=GetRecord
              &amp;identifier=oai%3AarXiv%3Aquant-ph%2F0201001
              &amp;metadataPrefix=oai_dc</requestURL>
 </GetRecord>
```

**Request**

Request a record in the oai_marc metadata format. However, the requested metadata format can not be disseminated for this identifier. Therefore, the response record contains a header but no metadata container [URL shown without encoding for better readability].

```
http://an.oa.org/OAI-
script?verb=GetRecord&identifier=oai:arXiv:quantph/9901001&metadataPrefix=oai
_marc
```

**Response**

```
<?xml version="1.0" encoding="UTF-8"?>
 <GetRecord
    xmlns="http://www.openarchives.org/OAI/1.1/OAI_GetRecord"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_GetRecord
            http://www.openarchives.org/OAI/1.1/OAI_GetRecord.xsd">
  <responseDate>2001-06-10T14:09:57-07:00</responseDate>
  <requestURL>http://an.oa.org/OAI-script?verb=GetRecord
              &amp;identifier=oai%3AarXiv%3Aquant-ph%2F9901001
              &amp;metadataPrefix=oai_marc</requestURL>
  <record>
   <header>
    <identifier>oai:arXiv:quant-ph/9901001</identifier>
```

```
      <datestamp>1999-01-01</datestamp>
    </header>
  </record>
 </GetRecord>
```

## *4.2. Identify*

### Summary and Usage Notes

This verb is used to retrieve information about a repository, including administrative, identity, and community-specific information.

### Arguments

None

### Exception Conditions

None

### Response Format

A XML schema for validating the response is as follows:

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:oai="http://www.openarchives.org/OAI/1.1/OAI_Identify"
targetNamespace="http://www.openarchives.org/OAI/1.1/OAI_Identify"
        elementFormDefault="qualified"
        attributeFormDefault="unqualified">

<annotation>
 <documentation>
   Schema to verify validity of responses to Identify OAI-protocol request.
   This Schema validated at http://www.w3.org/2001/03/webdata/xsv on 2001-05-07
   with XSV XSV 1.189/1.95 of 2001/05/07 08:38:12
 </documentation>
</annotation>

<element name="Identify" type="oai:IdentifyType"/>

<!-- response to Identify-request -->

<complexType name="IdentifyType">
 <sequence>
   <element name="responseDate" minOccurs="1" maxOccurs="1" type="dateTime"/>
   <element name="requestURL" minOccurs="1" maxOccurs="1" type="anyURI"/>
   <element name="repositoryName" minOccurs="1" maxOccurs="1" type="string"/>
   <element name="baseURL" minOccurs="1" maxOccurs="1" type="anyURI"/>
   <element name="protocolVersion" minOccurs="1" maxOccurs="1" type="string"/>
   <element name="adminEmail" minOccurs="1" maxOccurs="1" type="anyURI"/>
   <element name="description" minOccurs="0" maxOccurs="unbounded"
           type="oai:descriptionType"/>
 </sequence>
</complexType>

<complexType name="descriptionType">
 <sequence>
```

```
  <any namespace="##other" processContents="lax"/>
 </sequence>
</complexType>

</schema>
```

This Schema is available at http://www.openarchives.org/OAI/1.1/OAI_Identify.xsd

## Examples

**Request**

```
http://an.oa.org/OAI-script?verb=Identify
```

**Response**

The response to this protocol request includes the following elements that must be provided by every OAI compliant repository:

- `repositoryName`: a human readable name for the repository, in the example "The University of Spa E-print System";

- `baseURL`: the BASE-URL of the repository;

- `protocolVersion`: the version of the OAI protocol supported by the repository;

- `adminEmail`: the e-mail address of the administrator of the repository.

In addition, the response may contain a list of `description` containers, which provide an extensible mechanism for communities to describe their repositories. The `description` container, could -- for instance -- be used to include collection-level metadata in the response to the Identify request. Each `description` container must be accompanied by the URL of an XML schema, which provides the semantics of the descriptive container. The below example of a response to the `Identify` request contains two description containers:

- The `oai-identifier` container complies to an XML Schema, which is available at http://www.openarchives.org/OAI/1.1/oai-identifier.xsd. This schema, shown in Appendix 2, is used by repositories that choose to comply with a specific format of unique identifiers for records. The format of that identifier is explained by means of comments in the oai-identifier.xsd XML Schema.

- The `eprints` container complies to an XML Schema, which is available at http://www.openarchives.org/OAI/1.1/eprints.xsd. This schema, shown in Appendix 2, has been agreed upon by the OAI e-print community, and contains information specific to repositories in that community.

```
<?xml version="1.0" encoding="UTF-8"?>
  <Identify
    xmlns="http://www.openarchives.org/OAI/1.1/OAI_Identify"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_Identify
            http://www.openarchives.org/OAI/1.1/OAI_Identify.xsd">
    <responseDate>2001-06-01T19:20:30-04:00</responseDate>
    <requestURL>http://an.oa.org/OAI-script?verb=Identify</requestURL>
    <repositoryName>The University of Spa E-print System</repositoryName>
    <baseURL>http://an.oa.org/OAI-script</baseURL>
    <protocolVersion>1.1</protocolVersion>
    <adminEmail>mailto:adm&#x0040;spa.ac.be</adminEmail>
    <description>
```

```
      <oai-identifier xmlns="http://www.openarchives.org/OAI/1.1/oai-
identifier"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/oai-
identifier
                         http://www.openarchives.org/OAI/1.1/oai-
identifier.xsd">
        <scheme>oai</scheme>
        <repositoryIdentifier>bespa</repositoryIdentifier>
        <delimiter>:</delimiter>
        <sampleIdentifier>oai:bespa:medi99-123</sampleIdentifier>
      </oai-identifier>
    </description>
    <description>
      <eprints xmlns="http://www.openarchives.org/OAI/1.1/eprints"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/eprints
http://www.openarchives.org/OAI/1.1/eprints.xsd">
        <content>
          <URL>http://an.oa.org/info/content.htm</URL>
        </content>
        <metadataPolicy>
          <text>Metadata can be used by commercial and non-commercial service
           providers</text>
          <URL>http://an.oa.org/info/metadata_use.htm</URL>
        </metadataPolicy>
        <dataPolicy>
          <text>Full content, i.e. preprints may not be harvested by
robots</text>
        </dataPolicy>
        <submissionPolicy>
          <URL>http://an.oa.org/info/submission.htm</URL>
        </submissionPolicy>
      </eprints>
    </description>
  </Identify>
```

## *4.3. ListIdentifiers*

### Summary and Usage Notes

This verb is used to retrieve the identifiers of records that can be harvested from a repository. Optional arguments permit selectivity of the identifiers - based on their membership in a specific Set in the repository or based on their modification, creation, or deletion within a specific date range.

### Arguments

- **until** an *OPTIONAL* argument with a date value, which specifies that only the unique identifiers of records with a datestamp older than or equal to the specified date should be returned.

- **from** an *OPTIONAL* argument with a date value, which specifies that only the unique identifiers of records with a datestamp that is more recent than or equal to the specified date should be returned.

- **set** an *OPTIONAL* argument with a setSpec value, which specifies that only the unique identifiers of records from the specified Set should be returned.

- **resumptionToken** an *EXCLUSIVE* argument with a value that is the <u>flow control</u> token returned by a previous `ListIdentifiers` request that issued a partial response.

## Exception Conditions

- *No records match the request* - The response will not contain any identifier elements.

## Response Format

A XML schema for validating the response is as follows:

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:oai="http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers"
targetNamespace="http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers"
        elementFormDefault="qualified"
        attributeFormDefault="unqualified">


<annotation>
 <documentation>
   Schema to verify validity of responses to ListIdentifiers OAI-protocol
request.
   This Schema validated at http://www.w3.org/2001/03/webdata/xsv on 2001-05-
07
   with XSV XSV 1.189/1.95 of 2001/05/07 08:38:12
 </documentation>
</annotation>

<element name="ListIdentifiers" type="oai:ListIdentifiersType"/>

<!-- response to ListIdentifiers-request -->
<!-- records have an optional "deleted" status -->
<!-- this response may contain an optional resumptionToken -->

<complexType name="ListIdentifiersType">
 <sequence>
   <element name="responseDate" minOccurs="1" maxOccurs="1" type="dateTime"/>
   <element name="requestURL" minOccurs="1" maxOccurs="1" type="anyURI"/>
   <element ref="oai:identifier" minOccurs="0" maxOccurs="unbounded"/>
   <element name="resumptionToken" minOccurs="0" maxOccurs="1" type="string"/>
 </sequence>
</complexType>

<element name="identifier">
 <complexType>
  <simpleContent>
   <extension base="anyURI">
    <attribute name="status" use="optional" type="oai:statusType"/>
   </extension>
  </simpleContent>
 </complexType>
</element>

 <!-- define statusType -->
 <!-- a record can have a status of "deleted" . -->
<simpleType name="statusType">
  <restriction base="string">
   <enumeration value="deleted"/>
```

```
    </restriction>
</simpleType>

</schema>
```

This Schema is available at [http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers.xsd](http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers.xsd)

## Examples

**Request**

List the unique identifiers of records added or modified since January 15, 1998 in the "hep" Set of the "physics" Set [URL shown without encoding for better readability].

```
http://an.oa.org/OAI-script?verb=ListIdentifiers&from=1998-01-
15&set=physics:hep
```

**Response**

A list of four identifiers is returned, one of which has a "deleted" status. In addition, a resumption token has been returned, indicating the list of identifiers is *incomplete* and one or more subsequent requests will need to be issued to complete the list.

```
<?xml version="1.0" encoding="UTF-8"?>
  <ListIdentifiers
      xmlns="http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers
http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers.xsd">
    <responseDate>2001-06-01T19:20:30-04:00</responseDate>
    <requestURL>http://an.oa.org/OAI-script?verb=ListIdentifiers
                &amp;from=1998-01-15&amp;setSpec=physics%3Ahep</requestURL>
     <identifier>oai:arXiv:hep-th/9801001</identifier>
     <identifier>oai:arXiv:hep-th/9801002</identifier>
     <identifier>oai:arXiv:hep-th/9801005</identifier>
     <identifier status="deleted">oai:arXiv:hep-th/9801010</identifier>
    <resumptionToken>xxx45abttyz</resumptionToken>
  </ListIdentifiers>
```

**Request**

Issue a subsequent request to the one issued above, with a single resumptionToken argument whose value is that returned in the previous response.

```
http://an.oa.org/OAI-script?verb=ListIdentifiers&resumptionToken=xxx45abttyz
```

**Response**

```
<?xml version="1.0" encoding="UTF-8"?>
  <ListIdentifiers
      xmlns="http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers
http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers.xsd">
    <responseDate>2001-06-01T19:20:30-04:00</responseDate>
    <requestURL>http://an.oa.org/OAI- script?verb=ListIdentifiers
                &amp;resumptionToken=xxx45abttyz</requestURL>
     <identifier>oai:arXiv:hep-th/9801020</identifier>
     <identifier>oai:arXiv:hep-th/9801060</identifier>
  </ListIdentifiers>
```

List the unique identifiers of records added or modified on January 1, 2001 in the "hep" Set of the "physics" Set. There are no matches for this request, hence, the response does not contain any identifier tags [URL shown without encoding for better readability].

```
http://an.oa.org/OAI-script?verb=ListIdentifiers&from=2001-01-01&until=2001-
01-01&set=physics:hep
```

**Response**

```
<?xml version="1.0" encoding="UTF-8"?>
  <ListIdentifiers
      xmlns="http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers
http://www.openarchives.org/OAI/1.1/OAI_ListIdentifiers.xsd">
    <responseDate>2001-06-01T19:20:30-04:00</responseDate>
    <requestURL>http://an.oa.org/OAI-script?verb=ListIdentifiers
              &amp;from=2001-01-01&amp;until=2001-01-01
              &amp;setSpec=physics%3Ahep</requestURL>
  </ListIdentifiers>
```

## *4.4. ListMetadataFormats*

### Summary and Usage Notes

This verb is used to retrieve the metadata formats available from a repository.  An optional argument restricts the request to the formats available for a specific record.

### Arguments

- **identifier** an *OPTIONAL* argument that specifies the unique identifier for which available metadata formats is being requested.  If this argument is omitted, then the response includes all metadata formats supported by this repository.  Note that the fact that a metadata format is supported by a repository does *not* mean that it can be disseminated by all items in that repository.

### Exception Conditions

- *The identifier does not exist* - The response will not contain a metadataFormat container.

### Response Format

A XML schema for validating the response is as follows:

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:oai="http://www.openarchives.org/OAI/1.1/OAI_ListMetadataFormats"
targetNamespace="http://www.openarchives.org/OAI/1.1/OAI_ListMetadataFormats"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

<annotation>
 <documentation>
   Schema to verify validity of responses to ListMetadataFormats OAI-protocol
   request.
   This Schema validated at http://www.w3.org/2001/03/webdata/xsv on 2001-05-
07
   with XSV XSV 1.189/1.95 of 2001/05/07 08:38:12
 </documentation>
</annotation>
```

```
<element name="ListMetadataFormats" type="oai:ListMetadataType"/>

<!-- response to ListMetadataFormats-request -->

<complexType name="ListMetadataType">
 <sequence>
    <element name="responseDate" minOccurs="1" maxOccurs="1" type="dateTime"/>
    <element name="requestURL" minOccurs="1" maxOccurs="1" type="anyURI"/>
    <element name="metadataFormat" minOccurs="0" maxOccurs="unbounded"
            type="oai:metadataFormatType"/>
 </sequence>
</complexType>

<complexType name="metadataFormatType">
 <sequence>
   <element name="metadataPrefix" minOccurs="1" maxOccurs="1"
            type="oai:metadataPrefixType"/>
   <element name="schema" minOccurs="1" maxOccurs="1" type="anyURI"/>
   <element name="metadataNamespace" minOccurs="0" maxOccurs="1"
            type="anyURI"/>
 </sequence>
</complexType>

<simpleType name="metadataPrefixType">
 <restriction base="string">
   <pattern value="[a-zA-Z0-9_]+"/>
 </restriction>
</simpleType>

</schema>
```

This Schema is available at [http://www.openarchives.org/OAI/1.1/OAI_ListMetadataFormats.xsd](http://www.openarchives.org/OAI/1.1/OAI_ListMetadataFormats.xsd)

## Examples

**Request**

List the metadata formats that can be disseminated from the record with unique identifier oai:arXiv:hep-th/9901001 [URL shown without encoding for better readability].

```
http://an.oa.org/OAI-
script?verb=ListMetadataFormats&identifier=oai:arXiv:hep-th/9901001
```

**Response**

```
<?xml version="1.0" encoding="UTF-8"?>
  <ListMetadataFormats
      xmlns="http://www.openarchives.org/OAI/1.1/OAI_ListMetadataFormats"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_ListMetadataForma
ts
http://www.openarchives.org/OAI/1.1/OAI_ListMetadataFormats.xsd">
    <responseDate>2001-06-01T19:20:30-04:00</responseDate>
    <requestURL>http://an.oa.org/OAI-script?verb=ListMetadataFormats
                &amp;identifier=oai%3AarXiv%3Ahep-th%2F9901001</requestURL>
    <metadataFormat>
      <metadataPrefix>oai_rfc1807</metadataPrefix>
      <schema>http://www.openarchives.org/OAI/1.1/rfc1807.xsd</schema>
```

```
    </metadataFormat>
    <metadataFormat>
      <metadataPrefix>oai_dc</metadataPrefix>
      <schema>http://www.openarchives.org/OAI/1.1/dc.xsd</schema>
      <metadataNamespace>http://purl.org/dc/elements/1.1/</metadataNamespace>
    </metadataFormat>
</ListMetadataFormats>
```

**Request**

List the metadata formats that can be disseminated from the repository http://an.oa.org/OAI-script.

```
http://an.oa.org/OAI-script?verb=ListMetadataFormats
```

**Response**

```
<?xml version="1.0" encoding="UTF-8"?>
  <ListMetadataFormats
      xmlns="http://www.openarchives.org/OAI/1.1/OAI_ListMetadataFormats"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_ListMetadataForma
ts
http://www.openarchives.org/OAI/1.1/OAI_ListMetadataFormats.xsd">
    <responseDate>2001-06-01T19:20:30-04:00</responseDate>
    <requestURL>http://an.oa.org/OAI-
script?verb=ListMetadataFormats</requestURL>
    <metadataFormat>
      <metadataPrefix>oai_rfc1807</metadataPrefix>
      <schema>http://www.openarchives.org/OAI/1.1/rfc1807.xsd</schema>
    </metadataFormat>
    <metadataFormat>
      <metadataPrefix>oai_dc</metadataPrefix>
      <schema>http://www.openarchives.org/OAI/1.1/dc.xsd</schema>
      <metadataNamespace>http://purl.org/dc/elements/1.1/</metadataNamespace>
    </metadataFormat>
    <metadataFormat>
      <metadataPrefix>oai_marc</metadataPrefix>
      <schema>http://www.openarchives.org/OAI/1.1/oai_marc.xsd</schema>
    </metadataFormat>
</ListMetadataFormats>
```

**Request**

List the metadata formats that can be disseminated from the record with unique identifier oai:arXiv:hep-th/0101001. The identifier, however, does not exist and therefore, the response does not contain a metadataFormat container. [URL shown without encoding for better readability].

```
http://an.oa.org/OAI-
script?verb=ListMetadataFormats&identifier=oai:arXiv:hep-th/0101001
```

**Response**

```
<?xml version="1.0" encoding="UTF-8"?>
  <ListMetadataFormats
      xmlns="http://www.openarchives.org/OAI/1.1/OAI_ListMetadataFormats"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_ListMetadataForma
ts
http://www.openarchives.org/OAI/1.1/OAI_ListMetadataFormats.xsd">
    <responseDate>2001-06-01T19:20:30-04:00</responseDate>
    <requestURL>http://an.oa.org/OAI-script?verb=ListMetadataFormats
              &amp;identifier=oai%3AarXiv%3Ahep-th%2F0101001</requestURL>
</ListMetadataFormats>
```

## *4.5. ListRecords*

## Summary and Usage Notes

This verb is used to harvest records from a repository.  Optional arguments permit selectivity of the harvesting - based on the membership of records in a specific Set in the repository or based on their modification, creation, or deletion within a specific date range.

## Arguments

- **until** an *OPTIONAL* argument with a <u>date value</u>, which specifies that only records with a datestamp older than or equal to the specified date should be returned.

- **from** an *OPTIONAL* argument with a <u>date value</u>, which specifies that only records with a datestamp that is more recent than or equal to the specified date should be returned.

- **set** an *OPTIONAL* argument with a <u>setSpec value</u>, which specifies that only records from the specified <u>Set</u> should be returned.

- **resumptionToken** an *EXCLUSIVE* argument with a value that is the <u>flow control</u> token returned by a previous ListRecords request that issued a partial response.

- **metadataPrefix** a *REQUIRED* argument that specifies the <u>metadata prefix</u> of the format that should be included in the <u>metadata part of the returned record</u>s.  The value of this argument must be a non-space embedded string consisting of alphanumeric and underscore [_] characters; [A-Za-z0-9_].  The metadata formats supported by a repository and for a particular record can be retrieved using the <u>ListMetadataFormats</u> request.

## Exception Conditions

- *No records match the request* - The response will not contain any record containers.

- *Requested metadata format can not be disseminated for a matching record* - The response record contains a header but no metadata container.

## Response Format

A XML schema for validating the response is as follows:

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:oai="http://www.openarchives.org/OAI/1.1/OAI_ListRecords"
targetNamespace="http://www.openarchives.org/OAI/1.1/OAI_ListRecords"
        elementFormDefault="qualified"
        attributeFormDefault="unqualified">

 <annotation>
  <documentation>
    Schema to verify validity of responses to ListRecords OAI-protocol
```

```
request.
    This Schema validated at http://www.w3.org/2001/03/webdata/xsv on 2001-05-
07
    with XSV XSV 1.189/1.95 of 2001/05/07 08:38:12
  </documentation>
 </annotation>

 <element name="ListRecords" type="oai:ListRecordsType"/>

 <!-- response to ListRecords-request -->
 <!-- this response may contain an optional resumptionToken -->

 <complexType name="ListRecordsType">
  <sequence>
    <element name="responseDate" minOccurs="1" maxOccurs="1" type="dateTime"/>
    <element name="requestURL" minOccurs="1" maxOccurs="1" type="anyURI"/>
    <element name="record" minOccurs="0" maxOccurs="unbounded"
            type="oai:recordType"/>
    <element name="resumptionToken" minOccurs="0" maxOccurs="1"
type="string"/>
  </sequence>
  </complexType>

 <!-- define recordType -->
 <!-- a record has a header and a metadata part -->

 <complexType name="recordType">
  <sequence>
    <element name="header" minOccurs="1" maxOccurs="1" type="oai:headerType"/>
    <element name="metadata" minOccurs="0" maxOccurs="1"
type="oai:metadataType"/>
    <element name="about" minOccurs="0" maxOccurs="1" type="oai:aboutType"/>
  </sequence>
  <attribute name="status" use="optional" type="oai:statusType"/>
 </complexType>

 <!-- define headerType -->
 <!-- a header has a unique identifier and a datestamp -->

 <complexType name="headerType">
  <sequence>
    <element name="identifier" minOccurs="1" maxOccurs="1" type="anyURI"/>
    <element name="datestamp" minOccurs="1" maxOccurs="1" type="date"/>
  </sequence>
 </complexType>

 <!-- define metadataType -->
 <!-- metadata must be expressed in XML that complies with another XML Schema
-->
 <!-- metadata must be explicitly qualified in the response -->

 <complexType name="metadataType">
  <sequence>
   <any namespace="##other" processContents="lax"/>
  </sequence>
 </complexType>
```

Page 38

```
<!-- define aboutType -->
<!-- data "about" the record must be expressed in XML -->
<!-- that is compliant with an XML Schema defined by a community -->

<complexType name="aboutType">
 <sequence>
  <any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="1"/>
 </sequence>
</complexType>

 <!-- define statusType -->
 <!-- a record can have a status of "deleted" . -->
<simpleType name="statusType">
  <restriction base="string">
   <enumeration value="deleted"/>
  </restriction>
</simpleType>

</schema>
```
This Schema is available at [http://www.openarchives.org/OAI/1.1/OAI_ListRecords.xsd](http://www.openarchives.org/OAI/1.1/OAI_ListRecords.xsd)

## Examples

**Request**

List the records in rfc1807 format added or modified since January 15, 1998 in the "hep" set of the "physics" set [URL shown without [encoding](encoding) for better readability].

```
http://an.oa.org/OAI-script?verb=ListRecords&from=1998-01-
15&set=physics:hep&metadataPrefix=oai_rfc1807
```

**Response**

Three records are returned:

- The first includes the rfc1807 metadata. This record also has an "about" part.

- The second only has a header without the rfc1807 metadata container because the metadata for that record can not be expressed in rfc1807 format.

- The third has a status="deleted" attribute (and therefore no metadata part).

```
<?xml version="1.0" encoding="UTF-8"?>
  <ListRecords
      xmlns="http://www.openarchives.org/OAI/1.1/OAI_ListRecords"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_ListRecords
             http://www.openarchives.org/OAI/1.1/OAI_ListRecords.xsd">
    <responseDate>2001-06-01T19:20:30-04:00</responseDate>
    <requestURL>http://an.oa.org/OAI-script?verb=ListRecords
               &amp;from=1998-01-15&amp;setSpec=physics%3Ahep
               &amp;metadataPrefix=oai_rfc1807</requestURL>
    <record>
      <header>
        <identifier>oai:arXiv:hep-th/9901001</identifier>
        <datestamp>1999-12-25</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>v2</bib-version>
       <id>hep-th/9901001</id>
       <entry>January 1, 1999</entry>
       <title>Investigations of Radioactivity</title>
       <author>Ernest Rutherford</author>
       <date>March 30, 1999</date>
      </rfc1807>
    </metadata>
    <about>
     <dc xmlns="http://purl.org/dc/elements/1.1/"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://purl.org/dc/elements/1.1/
                          http://www.openarchives.org/OAI/1.1/dc.xsd">
     <publisher>Los Alamos arXiv</publisher>
     <rights>Metadata may be used without restrictions as long as the oai
      identifier remains attached to it.</rights>
     </dc>
    </about>
   </record>
   <record>
     <header>
       <identifier>oai:arXiv:hep-th/9901004</identifier>
       <datestamp>1999-12-26</datestamp>
     </header>
   </record>
   <record status="deleted">
     <header>
       <identifier>oai:arXiv:hep-th/9901007</identifier>
       <datestamp>1999-12-21</datestamp>
     </header>
   </record>
 </ListRecords>
```

## *4.6. ListSets*

### Summary and Usage Notes

This verb is used to retrieve the set structure in a repository.

### Arguments

- **resumptionToken** an *EXCLUSIVE* argument with a value that is the <u>flow control</u> token returned by a previous ListSets request that issued a partial response.

### Exception Conditions

- *Repository contains no set hierarchy* - The response will not contain any set containers.

## Response Format

A XML schema for validating the response is as follows:

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:oai="http://www.openarchives.org/OAI/1.1/OAI_ListSets"
targetNamespace="http://www.openarchives.org/OAI/1.1/OAI_ListSets"
         elementFormDefault="qualified"
         attributeFormDefault="unqualified">

  <annotation>
   <documentation>
    Schema to verify validity of responses to ListSets OAI-protocol request.
    This Schema validated at http://www.w3.org/2001/03/webdata/xsv on 2001-05-
07
    with XSV XSV 1.189/1.95 of 2001/05/07 08:38:12
   </documentation>
  </annotation>

 <element name="ListSets" type="oai:ListSetsType"/>

 <!-- this response may contain an optional resumptionToken -->

 <complexType name="ListSetsType">
  <sequence>
    <element name="responseDate" minOccurs="1" maxOccurs="1" type="dateTime"/>
    <element name="requestURL" minOccurs="1" maxOccurs="1" type="anyURI"/>
    <element name="set" minOccurs="0" maxOccurs="unbounded"
type="oai:setType"/>
    <element name="resumptionToken" minOccurs="0" maxOccurs="1"
type="string"/>
  </sequence>
 </complexType>

 <!-- each set in the list consists of a setSpec and a pretty name -->

 <complexType name="setType">
  <sequence>
    <element name="setSpec" minOccurs="1" maxOccurs="1"
type="oai:setSpecType"/>
    <element name="setName" minOccurs="1" maxOccurs="1" type="string"/>
  </sequence>
 </complexType>

 <simpleType name="setSpecType">
    <restriction base="string">
    <pattern value="([A-Za-z0-9])+(:[A-Za-z0-9]+)*"/>
    </restriction>
 </simpleType>

</schema>
```

This Schema is available at http://www.openarchives.org/OAI/1.1/OAI_ListSets.xsd

## Examples

**Request**

```
http://an.oa.org/OAI-script?verb=ListSets
```

**Response**

The following response indicates a set hierarchy with two top level sets - `Oceanside` and
`ValleyView` - each with two sets within them.

```
<?xml version="1.0" encoding="UTF-8"?>

  <ListSets
      xmlns="http://www.openarchives.org/OAI/1.1/OAI_ListSets"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_ListSets
              http://www.openarchives.org/OAI/1.1/OAI_ListSets.xsd">
    <responseDate>2001-06-01T19:20:30-04:00</responseDate>
    <requestURL>http://an.oa.org/OAI-script?verb=ListSets</requestURL>
    <set>
      <setSpec>Oceanside</setSpec>
      <setName>Oceanside University of Nebraska</setName>
    </set>
    <set>
      <setSpec>Oceanside:CompEnt</setSpec>
      <setName>Department of Computational Entomology</setName>
    </set>
    <set>
      <setSpec>Oceanside:MetPhen</setSpec>
      <setName>Department of Metaphysical Phenomenology</setName>
    </set>
    <set>
      <setSpec>ValleyView</setSpec>
      <setName>Valley View University of Florida</setName>
    </set>
    <set>
      <setSpec>ValleyView:Fren</setSpec>
      <setName>Department of Frenetics</setName>
    </set>
    <set>
      <setSpec>ValleyView:Hist</setSpec>
      <setName>Department of Histrionics</setName>
    </set>
</ListSets>
```

**Request**
`http://another.oa.org/OAI-script?verb=ListSets`
**Response**

The response shows that the repository does not have a set hierarchy.

```
<?xml version="1.0" encoding="UTF-8"?>

  <ListSets
      xmlns="http://www.openarchives.org/OAI/1.1/OAI_ListSets"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_ListSets
              http://www.openarchives.org/OAI/1.1/OAI_ListSets.xsd">
    <responseDate>2001-06-01T19:20:30-04:00</responseDate>
    <requestURL>http://another.oa.org/OAI-script?verb=ListSets</requestURL>
  </ListSets>
```

# Appendices

## *Appendix 1: Sample XML schema for metadata formats*

Each metadata format that is included in records disseminated by the OAI protocol is identified within the repository by a *metadata prefix* and across multiple repositories by the URL of a *metadata schema*. The *metadata schema* is an [XML schema](#) that may be used as a test of conformance of the metadata included in the record. XML Schemas for three metadata formats are given below:

- An XML Schema for the mandatory unqualified Dublin Core metadata format;

- An XML Schema for the rfc1807 metadata format;

- An XML Schema to represent MARC21 records in an XML format.

---

### *Dublin Core*

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
        xmlns:dc="http://purl.org/dc/elements/1.1/"
        targetNamespace="http://purl.org/dc/elements/1.1/"
        elementFormDefault="qualified"
        attributeFormDefault="unqualified">

  <annotation>
   <documentation>
    Schema for Dublin Core metadata format.
    the Open Archives Initiative. 2000.
    Schema validated at http://www.w3.org/2001/03/webdata/xsv on 05-09-2001
    Dublin Core semantics available at
          http://purl.org/DC/documents/rec-dces-19990702.htm
   </documentation>
  </annotation>

<element name="dc" type="dc:dublincoreType"/>

<complexType name="dublincoreType">
 <choice minOccurs="0" maxOccurs="unbounded">
  <element name="title" minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="creator"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="subject"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="description"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="contributor"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="publisher"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="date"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="type"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="format"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="identifier"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
```

```
  <element name="source"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="language"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="relation"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="coverage"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="rights"  minOccurs="0" maxOccurs="unbounded" type="string"/>
 </choice>
</complexType>

 </schema>
```

This Schema is available at http://www.openarchives.org/OAI/1.1/dc.xsd

## Examples

```
<dc xmlns="http://purl.org/dc/elements/1.1/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://purl.org/dc/elements/1.1/
                        http://www.openarchives.org/OAI/1.1/dc.xsd">
<title>The Cornell Law Quarterly</title>
<date>1915-1916</date>
<identifier>http://heinonline.org/HeinOnline/show.pl?
           handle=hein.journals/clqv1%26id=1%26size=4</identifier>
<rights>Available by Subscription. See http://www.wshein.com</rights>
</dc>

<dc:dc xmlns:dc="http://purl.org/dc/elements/1.1/"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://purl.org/dc/elements/1.1/
                           http://www.openarchives.org/OAI/1.1/dc.xsd">
<dc:title>Grassmann's space analysis</dc:title>
<dc:creator>Hyde, E. W. (Edward Wyllys)</dc:creator>
<dc:subject>LCSH:Ausdehnungslehre; LCCN QA205.H99</dc:subject>
<dc:publisher>J. Wiley &amp; Sons</dc:publisher>
<dc:date>Created: 1906; Available: 1991</dc:date>
<dc:type>text</dc:type>
<dc:identifier>http://resolver.library.cornell.edu/math/1796949</dc:identifie
r>
<dc:language>eng</dc:language>
<dc:rights>Public Domain</dc:rights>
</dc:dc>
```

## *rfc1807*

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
   xmlns:rfc1807="http://info.internet.isi.edu:80/in-
notes/rfc/files/rfc1807.txt"
   targetNamespace="http://info.internet.isi.edu:80/in-
notes/rfc/files/rfc1807.txt"
   elementFormDefault="qualified"
   attributeFormDefault="unqualified">

  <annotation>
   <documentation>
```

```
    Schema for rfc1807 metadata format.
    the Open Archives Initiative. 2000.
    rfc1807 semantics available at
            http://info.internet.isi.edu:80/in-notes/rfc/files/rfc1807.txt
    XML transportation format on which this schema is inspired
        available at http://www.openarchives.org/sfc/sfc_rfc1807.htm
    This Schema was validated with XSV 1.196/1.98 of 2001/06/16 11:56:53
   </documentation>
  </annotation>

<element name="rfc1807" type="rfc1807:rfc1807Type"/>

<complexType name="rfc1807Type">
 <sequence>
  <element name="bib-version" minOccurs="1" maxOccurs="1" type="string"/>
  <element name="id"  minOccurs="1" maxOccurs="1" type="string"/>
  <element name="entry"  minOccurs="1" maxOccurs="1" type="string"/>
  <element name="organization"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="title"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="type"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="revision"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="withdraw"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="author"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="corp-author"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="contact"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="date"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="pages"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="copyright"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="handle"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="other_access"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="retrieval"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="keyword"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="cr-category"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="period"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="series"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="monitoring"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="funding"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="contract"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="grant"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="language"  minOccurs="0" maxOccurs="unbounded"
type="string"/>
  <element name="notes"  minOccurs="0" maxOccurs="unbounded" type="string"/>
  <element name="abstract"  minOccurs="0" maxOccurs="unbounded"
```

```
type="string"/>
 </sequence>
</complexType>

</schema>
```

This Schema is available at http://www.openarchives.org/OAI/1.1/rfc1807.xsd

## *oai_marc*

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.openarchives.org/OAI/1.1/oai_marc"
xmlns:oai_marc="http://www.openarchives.org/OAI/1.1/oai_marc"
   elementFormDefault="qualified"
   attributeFormDefault="unqualified">

   <annotation>
     <documentation>
      Schema for MARC metadata format.
      MARC semantics available at http://www.loc.gov/marc/
      .....
      This Schema has been successfully applied for MARC21 records.
      It is likely to also work for older versions of USMARC and CANMARC.
      Application of this Schema for other MARC formats has not been
      tested and may require some adjustments.
      .....
      the Open Archives Initiative. 2000.
      Herbert Van de Sompel
      MARC XML transportation format on which this schema is inspired
          available at
http://www.dlib.vt.edu/projects/OAi/marcxml/marcxml.html
      This Schema validated at http://www.w3.org/2001/03/webdata/xsv on 05-09-
2001.
     </documentation>
   </annotation>

  <element name="oai_marc">
   <complexType>
     <sequence>
       <element ref="oai_marc:fixfield" minOccurs="1" maxOccurs="unbounded"/>
       <element ref="oai_marc:varfield" minOccurs="0" maxOccurs="unbounded"/>
     </sequence>
    <attribute name="status" type="string" use="optional"/>
    <attribute name="type" type="string" use="required"/>
    <attribute name="level" type="string" use="required"/>
    <attribute name="ctlType" type="string" use="optional"/>
    <attribute name="charEnc" type="string" use="optional"/>
    <attribute name="encLvl" type="string" use="optional"/>
    <attribute name="catForm" type="string" use="optional"/>
    <attribute name="lrRqrd" type="string" use="optional"/>
   </complexType>
  </element>

  <element name="fixfield">
```

Page 46

```xml
  <complexType>
    <simpleContent>
       <extension base="oai_marc:fixfieldType">
       <attribute name="id" type="oai_marc:idType" use="required"/>
       </extension>
    </simpleContent>
 </complexType>
</element>

<simpleType name="fixfieldType">
  <restriction base="string">
  <!-- fixfield must be enclosed between quotes because spaces
       are meaningfull -->
   <pattern value='[\n\r\t\s]*"[^"]*"[\n\r\t\s]*'/>
  </restriction>
</simpleType>

<element name="varfield">
 <complexType>
   <sequence>
     <element ref="oai_marc:subfield" minOccurs="1" maxOccurs="unbounded"/>
   </sequence>
  <attribute name="id" type="oai_marc:idType" use="required"/>
  <attribute name="i1" type="oai_marc:iType" use="required"/>
  <attribute name="i2" type="oai_marc:iType" use="required"/>
 </complexType>
</element>

<element name="subfield">
 <complexType>
   <simpleContent>
      <extension base="string">
      <attribute name="label" type="oai_marc:subfieldType" use="required"/>
      </extension>
   </simpleContent>
 </complexType>
</element>

<simpleType name="subfieldType">
  <restriction base="string">
  <!-- MARC subfield (the leading $ i not used)
       may be any lowercase alphabetic or numeric character  -->
   <pattern value="[0-9a-z]"/>
  </restriction>
</simpleType>

<simpleType name="idType">
  <restriction base="string">
  <!-- MARC tags are 1 to 3 digits -->
   <pattern value="[0-9]{1,3}"/>
  </restriction>
</simpleType>

<simpleType name="iType">
  <restriction base="string">
  <!-- MARC indicator may be any lowercase alphabetic or numeric character
```

```
        or a blank  -->
      <pattern value="[0-9a-z\s]?"/>
    </restriction>
  </simpleType>

</schema>
```

## Examples

```
<oai_marc
    status="n" type="a" level="m" encLvl="7"
    catForm="a" xmlns="http://www.openarchives.org/OAI/1.1/oai_marc"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/oai_marc
                        http://www.openarchives.org/OAI/1.1/oai_marc.xsd">

<fixfield id="1">"   90178038 "</fixfield>
<fixfield id="3">"DLC"</fixfield>
<fixfield id="5">"19900814092959.1"</fixfield>
<fixfield id="8">"900724s1974    po af          000 0 fre  "</fixfield>
<varfield id="10" i1=" " i2=" ">
 <subfield label="a">90178038</subfield>
</varfield>
<varfield id="40" i1=" " i2=" ">
 <subfield label="a">DLC</subfield>
 <subfield label="c">DLC</subfield>
</varfield>
<varfield id="50" i1="0" i2="0">
 <subfield label="a">MLCL 90/00723 (Q)</subfield>
</varfield>
 <varfield id="100" i1="1" i2="0">
 <subfield label="a">Berthou, P. Y.</subfield>
 <subfield label="q">(Pierre Yves)</subfield>
</varfield>
 <varfield id="245" i1="1" i2="3">
 <subfield label="a">Le C&#x00E9;nomanien de l"Estr&#x00E9;madure
                     portugaise /</subfield>
 <subfield label="c">par Pierre-Yves Berthou.</subfield>
</varfield>
<varfield id="260" i1="0" i2=" ">
 <subfield label="a">Lisboa :</subfield>
 <subfield label="b">Minist&#x00E9;rio da Economia, Secretaria de Estado
          da Ind&#x00FA;stria, Direc&#x00E7;&#x00E3;o-Geral de Minas e
          Servi&#x00E7;os Geol&#x00F3;gicos, Servi&#x00E7;os
          Geol&#x00F3;gicos de Portugal,</subfield>
 <subfield label="c">1973 [i.e. 1974]</subfield>
</varfield>
<varfield id="300" i1=" " i2=" ">
 <subfield label="a">168 p., 67 leaves of plates :</subfield>
 <subfield label="b">ill. ;</subfield>
 <subfield label="c">33 cm.</subfield>
</varfield>
<varfield id="490" i1="1" i2=" ">
 <subfield label="a">Mem&#x00F3;ria ;</subfield>
 <subfield label="v">no. 23 (Nova s&#x00E9;r.)</subfield>
```

Page 48

```
</varfield>
<varfield id="653" i1="0" i2=" ">
 <subfield label="a">Stratigraphic geology;</subfield>
 <subfield label="a">Cenomanian deposits;</subfield>
 <subfield label="a">Portugal</subfield>
</varfield>
<varfield id="830" i1=" " i2="0">
 <subfield label="a">Mem&#x00F3;ria (Servi&#x00E7;os Geol&#x00F3;gicos
          de Portugal) ;</subfield>
 <subfield label="v">nova s&#x00E9;r. no. 23.</subfield>
</varfield>
</oai_marc>
```

An oai_marc formated record embedded in in a reply to a GetRecord protocol request:

```
<?xml version="1.0" encoding="UTF-8"?>
  <GetRecord
     xmlns="http://www.openarchives.org/OAI/1.1/OAI_GetRecord"
     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/OAI_GetRecord
                  http://www.openarchives.org/OAI/1.1/OAI_GetRecord.xsd">
  <responseDate>2001-01-20T20:32:09-05:00</responseDate>
  <requestURL>http://memory.loc.gov/cgi-bin/oai1_0?verb=GetRecord
      &amp;metadataPrefix=oai_marc
      &amp;identifier=oai%3Alcoa1%3Aloc.gmd%2Fg3711p.rr004620</requestURL>
    <record>
      <header>
        <identifier>oai:lcoa1:loc.gmd/g3711p.rr004620</identifier>
        <datestamp>2000-01-04</datestamp>
      </header>
      <metadata>
      <oai_marc status="c" type="e" level="m" encLvl="1" catForm=" "
               xmlns="http://www.openarchives.org/OAI/oai_marc"
               xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xsi:schemaLocation="http://www.openarchives.org/OAI/oai_marc
               http://www.openarchives.org/OAI/1.1/oai_marc.xsd">
      <fixfield id="001">"   98688713 "</fixfield>
      <fixfield id="003">"DLC"</fixfield>
      <fixfield id="005">"20000104155648.0"</fixfield>
      <fixfield id="007">"cr||||"</fixfield>
      <fixfield id="007">"aj|aanzn"</fixfield>
      <fixfield id="008">"980520s1850    nyu        a    0    eng
"</fixfield>
      <varfield id="035" i1=" " i2=" ">
        <subfield label="9">(DLC)   98688713</subfield>
      </varfield>
      <varfield id="035" i1=" " i2=" ">
        <subfield label="a">(DLC)5572293</subfield>
      </varfield>
      <varfield id="010" i1=" " i2=" ">
        <subfield label="a">   98688713 </subfield>
      </varfield>
      <varfield id="040" i1=" " i2=" ">
        <subfield label="a">DLC</subfield>
        <subfield label="c">DLC</subfield>
        <subfield label="d">DLC</subfield>
```

```xml
  </varfield>
  <varfield id="050" i1="0" i2="0">
    <subfield label="a">G3711.P3 1850</subfield>
    <subfield label="b">.J4</subfield>
    <subfield label="u">RR 462</subfield>
  </varfield>
  <varfield id="052" i1=" " i2=" ">
    <subfield label="a">3711</subfield>
  </varfield>
  <varfield id="072" i1=" " i2="7">
    <subfield label="a">P3</subfield>
    <subfield label="2">lcg</subfield>
  </varfield>
  <varfield id="100" i1="1" i2=" ">
    <subfield label="a">Jervis, John B.</subfield>
    <subfield label="q">(John Bloomfield),</subfield>
    <subfield label="d">1795-1885.</subfield>
  </varfield>
  <varfield id="245" i1="1" i2="0">
    <subfield label="a">Skeleton map, showing the position and
                connections of the Michigan Southern Rail Road
                (from Toledo to Chicago) with the several great
                rail road routes to the Atlantic seaboard and
                New York City via the south shore of Lake Erie.
                </subfield>
  </varfield>
  <varfield id="260" i1=" " i2=" ">
    <subfield label="a">N[ew] Y[ork],</subfield>
    <subfield label="c">1850.</subfield>
  </varfield>
  <varfield id="300" i1=" " i2=" ">
    <subfield label="a">map</subfield>
    <subfield label="c">22 x 51 cm.</subfield>
  </varfield>
  <varfield id="507" i1=" " i2=" ">
    <subfield label="a">Scale ca. 1:3,000,000.</subfield>
  </varfield>
  <varfield id="510" i1="4" i2=" ">
    <subfield label="a">LC Railroad maps,</subfield>
    <subfield label="c">462</subfield>
  </varfield>
  <varfield id="500" i1=" " i2=" ">
    <subfield label="a">Description derived from published
              bibliography.</subfield>
  </varfield>
  <varfield id="520" i1=" " i2=" ">
    <subfield label="a">Outline map of the northeastern
        United States showing railroads in operation,
        under construction, and under proposal. Chartered
        in 1846 and consolidated April 1855 with the Northern
        Indiana Railroad under the name Michigan Southern and
        Northern Indiana Railroad.</subfield>
  </varfield>
  <varfield id="530" i1=" " i2=" ">
    <subfield label="a">Available also through the Library
        of Congress Web site as a raster image.</subfield>
  </varfield>
```

```
        <varfield id="610" i1="2" i2="0">
          <subfield label="a">Michigan Southern Railroad</subfield>
          <subfield label="v">Maps.</subfield>
        </varfield>
        <varfield id="650" i1=" " i2="0">
          <subfield label="a">Railroads</subfield>
          <subfield label="z">Northeastern States</subfield>
          <subfield label="v">Maps.</subfield>
        </varfield>
        <varfield id="752" i1=" " i2=" ">
          <subfield label="a">United States</subfield>
          <subfield label="c">Northeastern States.</subfield>
        </varfield>
        <varfield id="710" i1="2" i2=" ">
          <subfield label="a">Michigan Southern Railroad.</subfield>
        </varfield>
        <varfield id="852" i1="0" i2=" ">
          <subfield label="a">Library of Congress</subfield>
          <subfield label="b">Geography and Map Division</subfield>
          <subfield label="e">Washington, D.C. 20540-4650 USA</subfield>
          <subfield label="n">dcu</subfield>
        </varfield>
        <varfield id="856" i1="7" i2=" ">
          <subfield label="d">g3711p</subfield>
          <subfield label="f">rr004620</subfield>
          <subfield label="g">urn:hdl:loc.gmd/g3711p.rr004620</subfield>
          <subfield
label="u">http://hdl.loc.gov/loc.gmd/g3711p.rr004620</subfield>
          <subfield label="2">http</subfield>
        </varfield>
      </oai_marc>
    </metadata>
  </record>

</GetRecord>
```

## *Appendix 2: Sample XML Schemas for the `description` part of a reply to Identify request*

The response to an Identify request may contain a list of `description` containers, which provide an extensible mechanism for communities to describe their repositories. Each `description` container must be accompanied by the URL of an XML schema, which provides the semantics of the container.  XML Schemas for two examples of `description` containers are given below:

- The `oai-identifier` which is used by repositories that choose to comply with a specific format of unique identifiers for records. The format of that identifier is explained by means of comments in the oai-identifier.xsd XML Schema.

- The `eprints` container which has been agreed upon by the OAI e-print community. The container is used to describe content and policy properties of repositories in the e-print community.

### *description for repositories that share the OAI format for unique identifiers of records*

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
          xmlns:oai-identifier="http://www.openarchives.org/OAI/1.1/oai-
identifier"
          targetNamespace="http://www.openarchives.org/OAI/1.1/oai-
identifier"
          elementFormDefault="qualified"
          attributeFormDefault="unqualified">

   <annotation>
    <documentation>
     Schema for description section of Identify reply, for repositories
        that comply with the oai format for unique identifiers for records.
     the Open Archives Initiative. 2000.
     Validated at http://www.w3.org/2001/03/webdata/xsv on 05-09-2001.
    </documentation>
   </annotation>

  <element name="oai-identifier" type="oai-identifier:oai-identifierType"/>

  <complexType name="oai-identifierType">
   <sequence>
    <!--     ....    -->
    <!-- Repositories that comply to the OAI format of unique record -->
    <!-- identifiers, use unique identifiers that are the concatenation of: --
>
    <!--     * A "scheme" that must be equal to oai -->
    <!--     * A "repositoryIdentifier" that is a unique identifier for a -->
    <!-- repository.  Its uniqueness will be tested as part of a -->
    <!-- registration process.  The repositoryIdentifier is case -->
    <!-- sensitive and may contain the characters [a-z]|[A-Z]|[0-9] -->
    <!-- e.g. arXiv, VTETD.
    <!--     * A local identifier that is the unique identifier of a record -->
    <!-- within a repository -->
    <!--     * These three parts must be concatenated using a "delimiter" -->
    <!-- which must be a colon.    -->
    <!--     ....    -->
    <element name="scheme" minOccurs="1" maxOccurs="1" fixed="oai"/>
    <element name="repositoryIdentifier" minOccurs="1" maxOccurs="1"
            type="oai-identifier:repositoryIdentifierType"/>
    <element name="delimiter" minOccurs="1" maxOccurs="1" fixed=":"/>
    <!--     ....    -->
    <!-- "sampleIdentifier" : A sample identifier from the repository, -->
    <!-- e.g. oai:arXiv:hep-th/9901001 , oai:VTETD:etd-3345131939761081 -->
    <!--     ....    -->
    <element name="sampleIdentifier" minOccurs="1" maxOccurs="1"
            type="oai-identifier:IdentifierType"/>
   </sequence>
  </complexType>

  <simpleType name="repositoryIdentifierType">
   <restriction base="string">
     <pattern value="[a-zA-Z0-9]+"/>
```

Page 52

```
     </restriction>
   </simpleType>

   <simpleType name="IdentifierType">
    <restriction base="string">
      <pattern value="oai:[a-zA-Z0-9]+:.*"/>
    </restriction>
   </simpleType>

    </schema>
```

This Schema is available at http://www.openarchives.org/OAI/1.1/oai-identifier.xsd

## Examples

```
<oai-identifier xmlns="http://www.openarchives.org/OAI/1.1/oai-identifier"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/oai-identifier
              http://www.openarchives.org/OAI/1.1/oai-identifier.xsd">
<scheme>oai</scheme>
<repositoryIdentifier>bespa</repositoryIdentifier>
<delimiter>:</delimiter>
<sampleIdentifier>oai:bespa:medi99-123</sampleIdentifier>
</oai-identifier>
```

### *description used to describe content and policy properties of repositories in the e-print community*

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:eprints="http://www.openarchives.org/OAI/1.1/eprints"
targetNamespace="http://www.openarchives.org/OAI/1.1/eprints"
            elementFormDefault="qualified"
            attributeFormDefault="unqualified">

    <annotation>
     <documentation>
      Schema for description section of Identify reply, defined by
      the e-print community.
      the Open Archives Initiative. 2000.
      Validated at http://www.w3.org/2001/webdata/xsv on 05-09-2001.
     </documentation>
    </annotation>

  <element name="eprints" type="eprints:eprintsDescriptionType"/>

  <complexType name="eprintsDescriptionType">
   <sequence>
    <!--     ....     -->
    <!-- "content" : Text and/or a URL linking to text describing the -->
    <!-- content of the repository.      -->
    <!-- It would be appropriate to indicate the language(s) -->
    <!-- of the metadata/data in the repository. -->
    <!--     ....     -->
    <element name="content"  minOccurs="0" maxOccurs="1"
            type="eprints:TextURLType"/>
    <!--     ....     -->
```

```
    <!-- "metadataPolicy" : Text and/or a URL linking to text describing -->
    <!-- policies relating to the use of metadata  -->
    <!-- harvested through the OAI interface.  -->
    <!--     ....    -->
    <element name="metadataPolicy"  minOccurs="1" maxOccurs="1"
            type="eprints:TextURLType"/>
    <!--     ....    -->
    <!-- "dataPolicy" : Text and/or a URL linking to text describing -->
    <!-- policies relating to the data  held in the -->
    <!-- repository.  This may also describe policies -->
    <!-- regarding downloading data (full-content). -->
    <!--     ....    -->
    <element name="dataPolicy"  minOccurs="1" maxOccurs="1"
            type="eprints:TextURLType"/>
    <!--     ....    -->
    <!-- "submissionPolicy" : -->
    <!-- Text and/or a URL linking to text describing -->
    <!-- policies relating to the submission of content -->
    <!-- to the repository (or other accession mechanisms).  -->
    <!--     ....    -->
    <element name="submissionPolicy"  minOccurs="0" maxOccurs="1"
            type="eprints:TextURLType"/>
    <!--     ....    -->
    <!-- "comment" : -->
    <!-- Text and/or a URL linking to text describing anything -->
    <!-- else that is not covered by the fields above. -->
    <!-- It would be appropriate to include additional -->
    <!-- contact details (additional to the adminEmail -->
    <!-- that is part of the response to the Identify request). -->
    <!--     ....    -->
    <element name="comment"  minOccurs="0" maxOccurs="unbounded"
            type="string"/>
  </sequence>
 </complexType>
 <complexType name = "TextURLType" >
  <choice minOccurs="0" maxOccurs="unbounded">
    <element name="URL" minOccurs="0" maxOccurs="unbounded" type="anyURI"/>
    <element name="text" minOccurs="0" maxOccurs="unbounded" type="string"/>
  </choice>
  </complexType>

   </schema>
```

This Schema is available at http://www.openarchives.org/OAI/1.1/eprints.xsd

## Examples

```
<eprints xmlns="http://www.openarchives.org/OAI/1.1/eprints"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.openarchives.org/OAI/1.1/eprints
    http://www.openarchives.org/OAI/1.1/eprints.xsd">
  <content>
     <URL>http://arXiv.org/arXiv_content.htm</URL>
  </content>
  <metadataPolicy>
     <text>Metadata can be used by commercial and non-commercial
           service providers</text>
     <URL>http://arXiv.org/arXiv_metadata_use.htm</URL>
```

```
   </metadataPolicy>
   <dataPolicy>
      <text>Full content, i.e. preprints may not be harvested by robots</text>
   </dataPolicy>
   <submissionPolicy>
      <URL>http://arXiv.org/arXiv_submission.htm</URL>
   </submissionPolicy>
</eprints>
```

## *Acknowledgements*

## *Document History*

**2001-08-07**: Modified "oai_marc.xsd", "eprints.xsd", and "oai-identifier.xsd" to fix some whitespace problems and to make some attribute improvements. The changes are are complements of Hussein Suleman (and any typos are Donna Bergmark's).

**2001-06-20**: Released protocol version 1.1. Changes were made to every .xsd file as well as to all the examples. This version is compliant with the May, 2001 W3C XML Schema recommendation.

**2001-04-24**: Made changes to GetRecord.xsd , ListRecords.xsd and ListIdentifiers.xsd regarding the specification of the "status" attribute that can be used with a record. "status" was specified with use="fixed" and value="deleted", which caused parsers to use "deleted" as a default. The

correction specifies "status" as "optional" and restricts it to be of statusType, which can only have one value being "deleted".
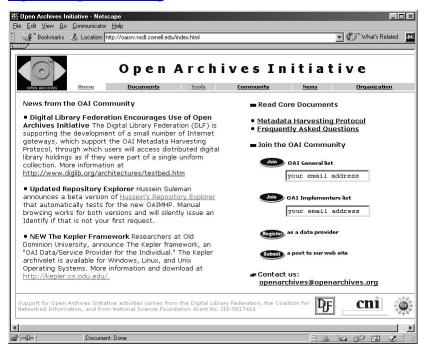
**2001-04-06**: Added occurency specification -- minOccurs="0" maxOccurs="1" -- to the aboutType in GetRecord.xsd and ListRecords.xsd .

**2001-02-23**: Corrected an error in ListMetadataFormats.xsd. The root element was listed as ListMetdata instead of ListMetadataFormats.

# Links

**Open Archives Initiative**

- http://www.openarchives.org



**Virginia Tech DLRL OAI Projects**

- http://www.dlib.vt.edu/projects/OAI/

# ✳Repository Explorer

- [http://purl.org/net/oai_explorer](http://purl.org/net/oai_explorer)



# ✳NDLTD
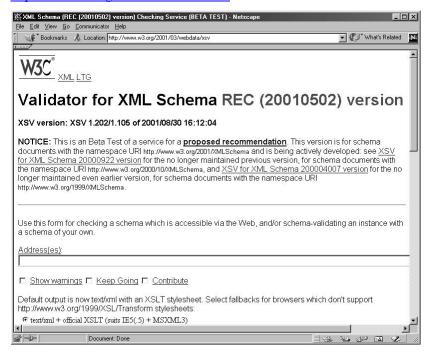
- [http://www.ndltd.org](http://www.ndltd.org)

# ✳ ARC Cross-Archive Search Service

- ◆ http://arc.cs.odu.edu/



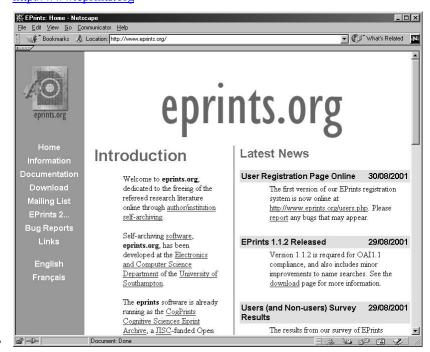# ✳ XML Schema Validator

- ◆ http://www.w3.org/2001/03/webdata/xsv

# ✳Dublin Core Metadata Initiative
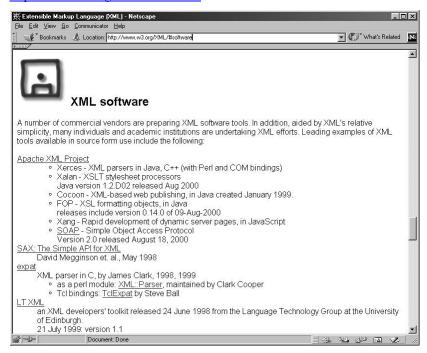
- http://www.dublincore.org



# ✳E-Prints DL-in-a-box

- http://www.eprints.org

# ✹XML Tools at W3C

- http://www.w3.org/XML/#software



# ✹OAI Metadata Harvesting Protocol

- http://www.openarchives.org/OAI/openarchivesprotocol.htm