IVOA Registry Interfaces
Version 0.1

IVOA Working Draft
2004-01-27

> **Editor's Note:**
> This document contains two types of boxed comments like this one. Those marked "Editor's Note" represents comments intended for the standard editors and for reviewers; these comments would be removed when the issues they discuss are addressed. Those marked simply as "Note" are intended for those who will implement the standard, and are intended to provide tips and further explanation of how the standard is expected to be used. These notes are expected to remain embedded in the final version of the document

4.3 Harvesting

Harvesting is the mechanism by which a registry can collect resource descriptions from other registries. This mechanism is used by full searchable registries to aggregate resource descriptions from many publishing registries. It can also be used to synchronize two registries to ensure that they have the same contents. This section defines the **IVOA Harvesting Interface**. Client applications that make use of this interface are referred to as **harvesters.** Those registries that declare themselves as harvestable (section 4.3.4) must comply with the specification described in this section.

The harvesting interface builds on the Web Service version of the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) [ref.]. In particular, all IVOA Registries that support the Harvesting Interface must be compliant with the Web Service version of OAI-PMH. Compliance with this base standard allows IVOA registries to be accessed by applications from outside the IVOA community.

> **Editor's Note:**
> OAI does not currently support an official Web Services version of PMH. One of the purposes of the development of this standard is to drive the evolution of the OAI standard which has demonstrated to be a highly effective harvesting protocol across a broad continuum of communities.

In addition to OAI-PMH compliance, this specification defines an additional set of OAI-PMH compliant requirements and recommendations which are described in sections 4.3.2 through 4.3.6 below.

4.3.1 A Summary of the OAI Web Service Interface

The Web Service version of OAI-PMH is defined by:
- The OAI-PMH v2.0 specification (http://www.openarchives.org/OAI/openarchivesprotocol.html) which defines
  - the meaning and behavior of the the six harvesting operations, referred to as "verbs",
  - the meaning of the input arguments for each operation, and
  - the XML Schema used to encode response messages.
- The OAI-PMH Web Service Definition Language (WSDL) document (see Appendix H.1) which defines
  - the six "verbs" defined as Web Service operations
  - SOAP encoding of the operation input arguments and response messages, based on the OAI-PMH XML Schema.

In summary, the OAI-PMH standard defines six operations:

Identify: provides a description of the registry

ListIdentifiers: returns a list of identifiers for the resource records held by the registry.

ListRecords: returns a set of resource descriptions.

GetRecord: returns a single resource description matching a given identifier.

ListMetadataFormats: returns a list of supported formats that the registry can use to encode resource descriptions upon a harvester's request.

ListSets: return a list of category names supported by the registry that harvesters can request in order to get back a subset of the descriptions held by the registry.

The **ListRecords** and **GetRecord** operations return the actual resource description records held by the registry. These descriptions are encoded in XML and wrapped in a general-purpose envelope defined by the OAI-PMH XML Schema (namespace http://www.openarchives.org/OAI/2.0).

Through the operations' arguments, OAI-PMH provides a number of useful features:

- *Support for multiple return formats*. As suggested by the **ListMetadataFormats** operation, a harvester can request the format resource descriptions are encoded in.
- *Harvesting by date*. The **ListIdentifiers** and **ListRecords** operations both support "from" and "until" date arguments. The "from" argument can be used to retrieve records that have changed since the last harvest.
- *Harvesting by category*. The **ListIdentifiers** and **ListRecords** operations both support a "set" argument for retrieving resources that are grouped in a particular category. Resource records may belong to multiple groups.
- *Marking records as deleted*. Registries may mark records as deleted so that harvesters may remove access to them from their applications.
- *Support for resumption tokens*. If a request results in returning a very large number of records, the registry can choose to split the results over several calls; this is done by passing a resumption token back to the harvester. The harvester uses it to retrieve the next set of matching results.

> **Editor's Note:**
> The Web Service version of the OAI-PMH protocol has been designed to match the behavior and functionality of the original "HTTP GET"-based version as much as possible. One reason for this is to make it as straight-forward as possible to build bridges between implementations of both types and to build off the existing OAI software.

> **Note:**
> It is important to note that the OAI-PMH interface is not intended to be a general search interface. The filtering capabilities described above are just enough to support intelligent harvesting between registries. Most end-user applications will use the search interface described in sections 3 and 4 to retrieve resource descriptions.

The Web Service or SOAP version of OAI-PMH augments the original specification with a standard Web Service Definition Language (WSDL) document which is listed in Appendix H.1. Harvestable registries complying to the SOAP version of OAI-PMH must emit a copy of the WSDL document, with a service element appropriate for the local endpoint URL added in, in response to a call to the Web Service URL with the standard "?wsdl" argument. All six of the standard operations must be implemented. Additional, non-standard operations may be added; however, the definition of the six standard operations, along with the definition of their inputs and outputs, must not be altered. The

interface is recognized as the OAI-PMH standard when the default namespace for the WSDL matches "http://www.ivoa.net/wsdl/oai.wsdl" exactly.

> **Editor's Note:**
> The namespace for the WSDL would presumably be changed to something like http://www.openarchives.org/OAI-WS/1.0/ if and when it is accepted by the OAI community.

The subsequent sections below describe how the standard OAI-PMH features are used to support IVOA-specific functionality.

4.3.2 Metadata Formats for Resource Descriptions

All IVOA registries that support the Harvesting Interface must support two standard metadata formats: the OAI Dublin Core format (mandated by the base OAI-PMH standard) and the IVOA VOResource metadata format [ref].

The VOResource metadata format will have the metadata prefix name "ivo_vor" which can be used where ever an OAI-PMH metadata prefix name is supported (see OAI standard, section 3.4, "metadataPrefix and Metadata Schema"). The format uses the VOResource core XML Schema with the namespace http://www.ivoa.net/xml/VOResource/v0.9 (referred hereto with the namespace prefix "vr") along with any legal extension of this schema (including the IVOA standard extensions) to encode the resource descriptions within the OAI-PMH **metadata** tag from the OAI XML Schema (namespace http://www.openarchives.org/OAI/2.0, hereto referred by the namespace prefix "oai"). The format is specifically defined as a **vr:VOResource** element as the sole child of the **oai:metadata** element. In compliance with the VOResource schema, the child of the **vr:VOResource** element may be any legal extension of the **vr:Resource** element (i.e. that is, an element that is in the same substitution group as **vr:Resource**), except where otherwise restricted by this document.

> **Editor's Note:**
> If and when the VOResource schema evolves to a new version, this standard must be updated accordingly. Thus, this definition is locked to particular version of the VOResource, so saying that a registry is compliant with vX.X of this document implies a specific version of VOResource.

> **Note:**
> It is possible that the **vr:Resource** extension returned is unrecognized by the harvester. See section 4.3.4. for details about how a harvester may use sets—particularly the "ivo_standard" set—to guarantee the return of records that can guarantee support for.

> **Note:**
> Use of a **vr:Resource** extension where a IVOA standard resource extension exists is strongly discouraged for records in the "ivo_vor" format. Implementers should consider defining a custom metadata format name to encode using non-standard **vr:Resource** extensions.

> **Editor's Note:**
> A "standard resource extension" will be defined as an element in the **vr:Resource** substitution group in a schema that has been approved as an IVOA Recommendation. At this writing, no VOResource schemas have not reached this state, so for the purposes of prototyping, a "standard resource extension" will refer to any element in the **vr:Resource** substitution group from the following schemas:
> - VOResource: http://www.ivoa.net/xml/VOResource/v0.9
> - VOCommunity: http://www.ivoa.net/xml/VOCommunity/v0.2
> - VORegistry: http://www.ivoa.net/xml/VORegistry/v0.2
> - VODataService: http://www.ivoa.net/xml/VODataService/v0.4

The OAI Dublin Core format, with the metadata prefix of "oai_dc", is defined by the OAI-PMH base standard and must be supported by all OAI-PMH compliant registries. This document does not specify how a record in the VOResource format maps into the OAI Dublin Core format; however, the IVOA Registry Working Group may recommend such a mapping based on the IVOA Resource Metadata standard [ref].

Harvestable registries may support other metadata formats. The **ListMetadataFormats** must list all names for formats supported by the registry; this list must include "ivo_vor" and "oai_dc".

4.3.3 Identifiers in OAI Messages

In accordance with the OAI-PMH standard, an OAI-PMH XML envelope that contains a resource description must include a globally unique URI that identifies that resource record. It is *not* intended to be the same as the URI that identifies the resource itself (which is encoded in the resource description itself). The implementer is free to assign these URIs as long as they are compliant with the OAI-PMH standard. These record identifiers are the ones that are returned by the **ListIdentifiers** operation.

4.3.4 Required Records

This section describes the records that a harvestable IVOA Registry must include among those it emits via the OAI-PMH operations.

The harvestable registry must return one record that describes the registry itself as a whole, and the "ivo_vor" format must be supported for this record. This record is included in the **Identify** operation response (see section 4.3.4). When encoded using the "ivo_vor" format, the sole child of the **vr:VOResource** element must be a **Registry** element from the VORegistry schema (namespace http://www.ivoa.net/xml/VORegistry/v0.2; hereto referred by the "vg" namespace prefix). The record must include a **vg:ManagedAuthority** for every Authority Identifier [ref IVOA Identifiers] that originated at that registry. The registry may contain other registry records for other registries it knows about; use of **vr:Resource** elements other than **vg:Registry** to describe these other registries is strongly discouraged.

> **Editor's Note:**
> The registry description record will also need to support additional metadata which is not currently defined in the VORegisry schema. An explanation of the required metadata should go here.
>
> Among the needed metadata is the base URL to use as the service access point. Another is an indication of whether the harvestable registry supports the Web Service or original HTTP GET versions of OAI-PMH.

The harvestable registry must return exactly one record in "ivo_vor" format for each Authority Identifer listed as a **vg:ManagedAuthority** in the **vg:Registry** record that describes that registry. When encoded in the "ivo_vor" format, the sole child of the **vr:VOResource** element must be an **vg:Authority** element.

4.3.5 The Identify Operation

The **Identify** operation describes the harvestable registry as a whole. The response from this operation must include all information required by the OAI-PMH standard. In particular, it must include a **oai:baseURL** element which must refer to the base URL to the Web Service endpoint (i.e. the URL used to retrieve the WSDL document via the standard URL suffix, "?wsdl").

> **Note:**
> A traditional "HTTP GET" implementation of OAI-PMH that serves as a bridge to Web Service implementation must transform the value of the **oai:baseURL** element to refer to itself rather than the delegate Web Service.

The **Identify** response must include a **oai:description** element containing a single **vg:Registry** element. This element must contain the proper namespace definitions for the record. The content of **vg:Registry** element must be the registry description of the harvestable registry itself. Other **oai:description** elements are allowed; however, there may only be one containing the **vg:Registry** element.

4.3.6 IVOA Supported Sets

Sets, as defined in the OAI-PMH standard, "is an optional construct for grouping items for the purpose of selective harvesting" (see the OAI-PMH standard, section 2.6). Harvestable IVOA registries are free to define any number of custom sets for categorizing records. The OAI-PMH standard allows a record to be a member of multiple sets. This document defines a set of reserved set names with special meanings. Their names all start with the characters "ivo_"; implementers must not define their own set names that begin with this string.   Support for two of the reserved sets, "ivo_standard" and "ivo_managed," are required by this specification; thus, when applied to IVOA-compliant harvestable registries, support for sets is not optional.

This specification implicitly defines a set for each of the IVOA standard extensions to the vr:Resource element as well as the vr:Resource element itself. The set name is formed by prepending "ivo_" to the local element name for the resource extension. (For example, a set defined for **vg:Registry** is named "ivo_Registry".) A request for records in such a set will return records whose "ivo_vor" rendering features the associated resource extension. (For example, requesting the "ivo_Registry" set will return all records whose "ivo_vor" form has a **vg:Registry** element as the child of the **vr:VOResource** element.) Requests for the "ivo_Resource" set (if supported) should return records whose "ivo_vor" form has a **vr:Resource** element as the child of the **vr:VOResource** element. Harvesting registries should support all sets associated with IVOA standard **Resource** extensions. Requests for these sets that are not supported should return an error (in accordance with the OAI-PMH standard), even if such records exist.

The "ivo_standard" set refers to all of the IVOA reserved sets that correspond to IVOA standard **Resource** extensions that are supported by the registry. Harvesters may request this set to guarantee getting back records it can fully parse. Harvesting registries must support this set.

The "ivo_managed" set refers to all records that originate from the queried registry. That is, those records that were harvested from other registries are excluded. The IVOA Resource identifiers given in the records must have an Authority Identifier that matches on of the **vg:ManagedAuthority** values in the **vg:Registry** record for that registry. Full searchable registries may use this set to avoid getting duplicate records when harvesting from many registries.

All sets that are supported by the harvestable registry, including the two required sets, must be listed in the response to the **ListSets** operation in compliance with the OAI-PMH standard. Appendix H.2 lists the recommended set descriptions which can be returned by the **ListSets** operation for the IVOA reserved set names.

4.3.7 Comments on Changes from Version 0.4 of this Document

> **Editor's Note:**
> This section would not appear in the final Registry Interface document. It is provided in this version to illustrate to reviewers how the original harvesting features are enabled in the OAI-based version presented in this document.

"RI-5 A harvesting query should only return resources for which it manages the authorityid."

  This is handled by requesting of the "ivo_managed" set (section 4.3.6).

"RI-5 Registry shall implement a "HarvestFrom" interface taking in a timestamp parameter. "from" gets results based on the ">=" (Greater-Than-Equal) Operation rule for dates."

  This is handled by the standard, optional OAI arguments, "from" and "until."

"RI-5 Registry shall implement a "Harvest" interface with no parameters that returns everything in the registry."

  This is handled by the **ListRecords** operation when called with no optional, filtering arguments.

The old "Harvest Resource" section is described as a new section 4.4 below. The old section 5 on "Paging" can be dropped as it is discussed for harvesting in section 4.3. Paging in the context of search queries should be described in the earlier sections regarding searches.

Section 4.5

4.4 Harvesters

A registry that collects resource descriptions from other registries through the Harvesting Interface defined above in section 4.3 are referred to as a **harvester registry.** A registry that operates in this mode should implement the Harvester Interface which provides a way for harvestable registries to request to be harvested from (e.g. because updates have recently occurred). A registry that conforms to this interface should indicate so within its registry description using the metadata provided in the VORegistry schema (namespace, http://www.ivoa.net/xml/VORegistry/v0.X). A harvester registry that does not support this interface is understood has supporting some other mechanism for deciding which registries to harvest from and when to harvest.

The Harvester Interface is defined by the single operation WSDL listed in Appendix H.3. The operation called "harvest" is called to request that the harvestable registry referred to in the inputs be harvested from at the next earliest convenience of the harvester. The

harvester, upon receipt of this request, has several options regarding when the harvesting will begin; it may choose:

1. to harvest immediately,
2. to postpone harvesting to a later time (e.g. to synchronize with its own update cycle), or
3. to not harvest at all (e.g. because the inputs do not meet its criteria for harvesting).

The operation's input arguments have the following meaning:

`ivo-id`: the IVOA Identifier for the harvestable registry requesting a harvest.

`harvestingType:` This indicates whether the harvesting registry supports the Web Serivce or the traditional HTTP Get version of the OAI protocol.

`baseURL`: the base URL for the service. Whether this is to interpreted as a Web Service endpoint or the base URL in the sense of the traditional "HTTP GET" version of OAI-PMH depends on the value of `harvestingType`.

`lastUpdate:` the date of the last update made to any of the records held by the caller (optional).

> **Note:**
> It is recommended that a harvester registry limit how frequently it re-harvests from a harvestable registry; that is, if the harvester has harvested from a registry before, it should choose option (2) above. This will prevent multiple calls to the harvest operation in quick succession from triggering multiple, unnecessary harvesting processes. Instead, it should queue the request and ignore subsequent requests until the initial harvesting is complete.

Appendix H.1 Web Services Definition Language Document for OAI-PHM

Appendix H.2 Recommended Descriptions for IVOA Reserved Sets

Appendix H.3 Web Services Definition Language Document for the Harvesters Interface