IVOA Standard Interfaces
Version 0.1

IVOA Working Draft
2004-01-07

This version:
0.1 http://www.ivoa.net/internal/IVOA/StandardInterfaces-0.1.pdf

Previous versions:

Editors:
William O'Mullane

Authors:
IVOA Web and Grid Services Working group

Please send comments to: mailto:grid@ivoa.net

Abstract

This document describes the minimum required interface to participate in the IVOA as a web service.

Status of this document

This is a Working Draft. There are no prior released versions of this document.

This is an IVOA Working Draft for review by IVOA members and other interested parties. It is a draft document and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use IVOA Working Drafts as reference materials or to cite them as other than "work in progress." A list of current IVOA Recommendations and other technical documents can be found at http://www.ivoa.net/docs/.

Acknowledgments

This work is based on discussions and actions from the 2003 IVOA meeting in Strasbourg and further discussions on registry functionality at JHU late in 2003.

Contents

Abstract 1
Status of this document 1
Acknowledgments 2
Contents 2
1 Introduction 2
1.1 About the document 2
2 Standard VO Interfaces 2
2.1 Meta Data 2
2.2 Heartbeat 3
2.3 Harvest Log 3
3 Changes from previous versions 3
4 References 4

1 Introduction

Web Services form an increasing part of the Virtual Observatory. It is felt there are some basic Interfaces which these services should provide. The word interface is used here in the SOAP sense, this has been referred to as a port in older SOAP documentation and perhaps still in the Grid community.
Hence this document attempt to define the standard interfaces which a service should provide.

1.1 About the document

In the normal requirements manner the words “should” and “shall” are used to convey the level of necessity of the interface. Each interface is clearly given a short description and a requirement number of the form SI-N where N is a running number in the document..

2 Standard VO Interfaces

2.1 Meta Data

For WebServices the service should be the single authoritative source for metadata. Currently services provide WSDL in a standard manner. WSDL alone is insufficient for the purposes of the VO. We need something more like the VOResource implementation of the RSM (Resource & Metadata) document. In a web service we would return the object type produced from the XSD for the RSM. The definition of VOResource must be included in the WSDL for the service. This does not preclude returning an extension of the VOResource.

SI-1 All VO WebServices shall implement the “MetaData” interface. This shall return a valid VOResource document describing the metadata of this service.

SI-2 All Vo WebServices shall implement the “MetaDataChangedOn” interface. This shall return the date the metadata last changed.

2.2 Heartbeat

The heartbeat interface is to tell us if the service is in operation. It should do a good check on the underlying service to see if it is still operational and not just be a simple return from a web server e.g. if it relies on a database it should check the database is still up.
Ultimately some portals may track these heartbeats and compile uptime statistics. With the location we could have 3D global maps of services and availability.

SI-3 All VO WebServices shall implement the “Heartbeat” interface. This shall return an xml document containing :

- Uptime - the up time of the service
- ValidTo - how long it believes this will be valid i.e. next scheduled downtime.
- ContactDetails – Name, Email and PhoneNumber of a person to contact if there is a problem
- Position – approximate GPS coordinates of the server.

2.3 Harvest Log

Many services perform some sort of logging – if only web logs. For a global Virtual Observatory it will be interesting to be able to correlate global logs. For example if we have multiple registries it would be nice to be able to have an idea of number of requests on each one and where they are coming from. This is of course a gory topic but we feel an initial effort should be made to have some rudimentary interface for grabbing logged information in some standard (preferably tabular) format.

SI-4 All VO WebServices should implement the “HarvestLog” interface. This interface should take two parameters “fromDate” and “toDate” and should return a set of LogEntry. A LogEnry would conatin the following information:

- IP - Address request came from
- Request - The actual request or some representation of it.
- Event - What that means internally i.e. QUERY,HARVEST,UPDATE etc.
- TimeIn - Time request was received.
- TimeOut - Time request was serviced.
- Volume - ~Size in bytes of the response
- Status - 0 for pending, 1 for done, 2 for failed
- Response - Error message if this failed, perhaps the actual response if it is a small response, otherwise some summary statement or blank. Mainly to have errors.

3 Changes from previous versions

- First draft

4 References