Simple Line Access Protocol

Version 0.5

Draft Document 17 May 2006

Editor(s):
> **Pedro Osuna**
> **Jesus Salgado**

Author(s):
> Jesus Salgado
> Pedro Osuna
> Matteo Guainazzi
> Isa Barbarisi
> Marie-Lise Dubernet
> Doug Tody

Abstract

This specification defines a protocol for retrieving spectral lines from different spectral line databases through a uniform interface.

The interface is meant to be reasonably simple to implement by service providers. A basic query will be done in a wavelength range for the different services. The service returns a list of spectral lines formatted as a VOTable.

This type of query, based in a spectral range search, could be too simple to obtain useful results for some services. For instance, in the case of theoretical spectral line databases, the number of lines returned, even for small wavelength ranges, could be huge.

However, some services could implement some other extra search parameters to restrict the search or to score the line as per its probability within a certain physical conditions. How these extra search parameters can be handled by the protocol will be studied in this document.

Status of This Document

This is the first release 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/Documents/.

Acknowledgements

The authors acknowledge the comments from the DAL forum members and from the IVOA members in general

Contents

1 Overview

This Simple Line Access Protocol (SLAP henceforth) specification makes use of the work done in the Line Data Model definition, as the source of the abstract representation of a spectral line. All the information collected in this document will be used to homogenize the access to the different already existing spectral line databases.

The approach followed by this protocol is the same one as the successful SIAP protocol. This is why this document is intended to be a translation of the SIAP protocol specification for spectral lines access, diverging from it when the physics of the problem force to do it. At the same time, some ideas of the access to Theoretical spectra have been developed and used for Theoretical Spectral Line databases.

The SIAP and SSAP protocols are a two steps process. In a first step, the VO client application request for metadata to the server. These metadata include links to images or spectra respectively. In a second step, the VO client application requests the data from the server. In the particular case of Simple Line Access Services, this kind of services diverges from the SIAP and SSAP ones, as the SLAP service will be, in essence, a one step process, i.e., only the first request for metadata is needed.

Even when no link to astronomical products is expected because of the nature of the service, the SLAP metadata could contain access reference links to html pages, spectra files, spectral line profiles,… etc.

2 Requirements for Compliance

The spectral line query web method MUST be supported as in section 4 below. Through this web method, clients search for spectral lines that match certain client-specified criteria. The response is a VOTable that describes the output spectral lines.

2.1 Compliance

The keywords "MUST", "REQUIRED", "SHOULD", and "MAY" as used in this document are to be interpreted as described in RFC 2119 [34].

An implementation is compliant if it satisfies all the MUST or REQUIRED level requirements for the protocols it implements. An implementation that satisfies all the MUST or REQUIRED level and all the SHOULD level requirements for its protocols is said to be "unconditionally compliant"; one that satisfies all the MUST level requirements but not all the SHOULD level requirements for its protocols is said to be "conditionally compliant".

3   Spectral Line Service Types

It is assumed that compliant spectral line services fall into one of two categories.

1. Observational line databases. Lines observed and identified in real spectra collected by different instrument/projects.

2. Theoretical line databases. Servers containing theoretical spectral lines will be included in this group.

In both cases, the line description and the identification might be already present in a scientific publication, which could be used as a curation mechanism.

This document describes the most common query parameters that these services **must/should/may** support, classified in two main categories: compulsories and non-compulsories.

However, the theoretical line databases services could make use of extra parameters not cited in this document to filter out lines not expected to be identified in an observed spectrum or to score the output lines due to the application of physical models.

Examples:

For the Atomic spectral line database from CD-ROM 23 of R. L. Kurucz
The absorption oscillator strength, log(gf), can be selected

For the NIST Atomic Spectra Database Lines
As it is extracted from the Saha-LTE model, "*The level populations are calculated according to the Boltzmann distribution within each ion and Saha distribution between the ion stages. Thus, to calculate the spectrum from a single ion, e.g., C I, only* T_e *is required, while for the spectrum from several ions of the same element (e.g., C I-V),* N_e *must be defined as well.*"

At the same time, for observational spectral line databases, some project specific search parameters may be used.

Example:

ISO Astronomical Spectroscopy Database (IASD)
 The observation number parameter can be used to select only the lines observed during this ISO satellite observation.

As this kind of extra parameters are not easy to be taken in account in a simple implementation or compiled in the document, a general mechanism is described. As it will be explained later, to allow the discovery of these extra parameters by the VO client applications or by the registry, an implementation of the format=metadata input parameter will be needed. The exact expected response when this parameter appears in the query will be discussed in point 5.

4 Spectral Line Query

The purpose of the spectral line query is to allow users/clients to search in a wavelength range for spectral lines. The most basic query parameters will be the minimum and maximum value for the wavelength range. Additional parameters may be used to refine the search or to model physical scenarios.

4.1 Input format

A URL with two parts transmits the spectral Line query input as an HTTP GET request represented:

 * A base URL of the form:
 http://<server-address>/<path>?[<extra GET arg>&[...]]

  Example:
 http://esavo02.esac.esa.int:8080/slap/slap.jsp?

Note that when it contains extra GET arguments, the base URL ends in an ampersand, &; if there are no extra arguments, then it ends in a question mark, ?.

Every query to a given spectral line query service uses the same base URL.
Constraints expressed as a list of ampersand-delimited GET arguments, each of the form:
 <name>=<value>

Examples
 WAVELENGTH=5.1E-6/5.6E-6

The constraints represent the query parameters that can vary for each successive query.

As it is showed in the example, ranges can be defined following a simple syntax described in appendix A.

This range syntax will be used for the wavelength range and for other input parameters.

For example:
        INITIAL_LEVEL_ENERGY =8.01E-18/1.6E-17
is equivalent to
INITIAL_LEVEL_ENERGY>=8.01E-18 AND INITIAL_LEVEL_ENERY<=1.6E-17

4.2 Compulsory input

Wavelength spectral range: The service **MUST** support the following two parameters that are used to specify the spectral wavelength range to be used in the search

  o **WAVELENGTH**
  A service **MUST** implement a parameter to specify the wavelength spectral range. To be specified in meters. This wavelength range will be interpreted as the wavelength in the vacuum of the transition originating the line (ucd="em.wl" ;utype="**ldm:Line.wavelength**"). For range definition, please refer to appendix A.

As the units in the spectral line database could be different than meters, the service will need to translate from the selected units (meter) to the internal ones. The election of one type of units (in this case the SI unit meters) will help to unify access to different spectral line databases, even when in some cases, the units selected (meter) could not be the best one for the range on interest.

 Example:
http://esavo02.esac.esa.es:8080/slap/jsp/slap.jsp?WAVELENGTH=5.1E-6/5.6E-6
to query for spectral lines in the wavelength range between 5.1 and 5.6 micrometers.

4.3 Non-compulsory Parameters

The next list of non-compulsory parameters may be implemented at server side:

- ○ **CHEMICAL_ELEMENT**
  A service **MAY** have a search parameter called CHEMICAL_ELEMENT. This parameter would constraint the search to the chemical element selected (ucd="phys.atmol.element";utype="**ldm: Line.initialElement.name"**)

Energy Level range: A service **MAY** accept constraints in the energy for the starting and final levels of the transition. For these parameters, only a range constraint compulsory with appendix A is allowed.

- ○ **INITIAL_LEVEL_ENERGY**
  A service **MAY** implement a parameter to specify the minimum and maximum energy for the INITIAL level of the transition. To be expressed in Joules (ucd=" phys.energy; phys.atmol.initial;phys.atmol.level";
  utype="**ldm: Line.initialLevel.energy"**)

- ○ **FINAL_LEVEL_ENERGY**
  A service **MAY** implement a parameter to specify the minimum and maximum energy for the FINAL level of the transition. To be expressed in Joules (ucd=" phys.energy; phys.atmol.final;phys.atmol.level";
  utype="**ldm: Line.finalLevel.energy"**)

The implementation of these search parameters may be extremely useful to model in the client physical situations.

- ○ **TEMPERATURE**
  A service **MAY** implement a parameter to specifythe expected temperature of the object. To be specified in Kelvin. This parameter would be used (in particular for theoretical spectral line databases) to score the lines in the output using physical models.
  (ucd="phys.temperature"; utype="**ldm:Line.environment.temperature**")

Emission probability: A service **MAY** accept constraints in the transition probability through the Einstein A coefficient, defined as the probability per unit time s^{-1} for spontaneous emission in a bound-bound transition. For these parameter, only the operators ">" and "<" should be implemented.

- ○ **EINSTEIN_A**
  A service **MAY** implement a parameter to specify the minimum and maximum Einstein A coefficient as described before. To be specified in s^{-1}.
  (ucd=" phys.at.transProb"; utype="**ldm:Line.einsteinA**")

4.4 Free query parameters

As we saw in point 3, there is a need to give a general mechanism for the use of some free query parameters to filter out or to score the table result.

Both for the non-compulsory parameters and/or for the free ones, the way to notify to a VO client the implementation or not of these parameters will be through the FORMAT=METADATA response. The exact format of this response will be studied in the next section.

Using this information, a VO client would be able to construct on the fly a form, where this information could be entered.

In the future, the SLAP services will be registered and the registry could be able to give the exact description of the SLAP service to the clients, so this process of discovery of special parameters could be done at this point. The exact mechanism for this discovery process will be defined in the registry evolution.

5 Service Metadata

Compliant simple spectral line services describe themselves in two ways. They advertise their availability by registering with a registry service. If possible, depending of the registry service type and the registry definition evolution, all the metadata needed to characterize the service should be included, in particular the **non-compulsory** and **free parameters** implemented by the service. The way this information will be supplied to VO clients will be defining in the registry definition documents.

On the other hand, the services advertise their capabilities through support of the special metadata query signaled through the FORMAT=METADATA parameter. The service metadata will allow the registry to discover the service. At the same time, this service will allow the VO clients to make use of the service capabilities, either through a direct call to the service or through the registry in the future.

5.1 Metadata Query

A compliant simple spectral line service MUST support spectral line queries with FORMAT=METADATA, used to query the service metadata; only metadata is returned by the service in this case. The response to this query advertises of the input parameters the service supports.

The structure of the FORMAT=METADATA result will be a VOTable with the parameters supported by the service, using the VOTable <PARAM> tag. Every input parameter supported by the service should be listed as a PARAM element of the RESOURCE that normally contains the spectral line list table. Each param should have a name attribute of the form "INPUT:param_name", where

param_name is the parameter name as it should appear in the query URL. For example, name="INPUT:WAVELENGTH" refers to the input parameter "WAVELENGTH".

All input parameters meant to be available to clients of the service must be listed as PARAM elements, including required parameter WAVELENGTH), non-compulsory parameters and free parameters specific to the service. The PARAM may contain a value attribute that should contain the default value that will be assumed it the parameter is not set in the query input URL. Implementors are encouraged to include, as children of the PARAMs, DESCRIPTION elements to describe the parameter and (where appropriate) VALUES elements to given allowed ranges or values.

At the same time, implementers are encouraged to include UCD elements in the PARAM tags to describe the physical meaning of one parameter, not always obvious in particular for physical models.

Example: The input parameter listing below from a Observed Spectral Lines database shows that in addition to supporting the required parameter (WAVELENGTH_MIN, WAVELENGTH_MIN), it also supports the free parameter OBSNO.

```
<RESOURCE type="results">
   <DESCRIPTION>IASD Simple Line Access Service</DESCRIPTION>
   <INFO name="QUERY_STATUS" value="OK"/>
   <PARAM name="INPUT:WAVELENGTH"
                              ucd="em.wl" utype=" ldm:Line.wavelength"  value="30">
     <DESCRIPTION> Specify the wavelength spectral range. To be specified in meters. This
wavelength will be interpreted as the wavelength in the vacuum of the transition originating the
line
.      </DESCRIPTION>
   </PARAM>


   <PARAM name="INPUT:OBSNO" ucd="obs.id">
     <DESCRIPTION> Specify the ISO observation number where this line has been identified
.      </DESCRIPTION>
   </PARAM>

…………………………………..
```

5.2 Registering a Compliant Service

To be updated in coordination with the Registry Working Group

6 Successful Output

The output returned by a SLAP service is a VOTable, an XML table format, returned with a MIME-type of text/xml. The table lists all the Spectral lines found in the server database that match the query constraints. The following requirements are placed on the contents of the table when the query successfully returns a list of spectral lines:

1. The VOTable MUST contain a RESOURCE element identified with the tag type="results" containing a single TABLE element which contains the results of the query. The VOTable is permitted to contain addition RESOURCE elements, but the usage of any such elements is not defined here. If multiple resources are present it is recommend that the query results be returned in the first resource element.
2. The RESOURCE element SHOULD contain an INFO with name="QUERY_STATUS". Its value attribute should set to "OK" if the query executed successfully, regardless of whether any matching spectral lines were found.

Examples:
```
<INFO name="QUERY_STATUS" value "OK">
<INFO name="QUERY_STATUS" value="OK"> Successful Search </INFO>
```

3. Each table row represents a different spectral line available to the client.
4. Each row of the output VOTable MUST contain FIELDs where the following UCDs have been set.
5. Every FIELD SHOULD contain a utype reference to the Line Data Model whenever possible.

6.1 Standard output fields

o Exactly one field **MUST** have utype="**ldm:Line.wavelength**", with datatype="double" and ucd="em.wl", containing the wavelength in the vacuum of the transition originating the line in meters.

o Exactly one field **MUST** have utype="**ldm:Line.title**", with datatype="char" arraysize="*" and ucd="em.line", containing a small description identifying the line.

Note that this line title is only a short string representation to be used in the clients for display. Syntax is free. Examples:
H I
N III 992.873 A

- Exactly one field **SHOULD** have utype="**ldm:Line.species.name**" with datatype="char", arraysize="*" and ucd="phys.atmol.element", containing a name of the chemical element source of this line.

- Exactly one field **SHOULD** have utype="**ldm:Line.initialLevel.name**", with datatype="char", arraysize="*" and ucd="phys.atmol.initial;phys.atmol.level", containing a full description of the initial level of the transition originating the line.

- Exactly one field **SHOULD** have utype="**ldm:Line.finalLevel.name**", with datatype="char", arraysize="*" and ucd="phys.atmol.final;phys.atmol.level", containing a full description of the final level of the transition originating the line.

- Exactly one field **MAY** have utype="**ldm:Line.observedWavelength**", with datatype="double" and ucd="em.wl", containing the observed wavelength in the vacuum of the transition originating the line in meters, as it was observed. This may be used by observational spectral line databases.

- Exactly one field **MAY** have utype="**Query.Score**", with datatype="double". A line with a higher score more closely matches the query parameters. The query response table should normally be returned sorter in order of decreasing values of score, with the top-scoring items at the top of the list. The details of the heuristic used to compute score are left to the service. This is particularly useful for the theoretical spectral line databases examples explained in section 3.

- Exactly one field **MAY** have utype="**ldm:Line.initialLevel.energy**", with datatype="double" and ucd="phys.energy;phys.atmol.initial;phys.atmol.level", describing the energy for the starting level of the transition which originates this line. To unify results, the value must appear in Joules

- Exactly one field **MAY** have utype="**ldm:Line.finalLevel.energy**", with datatype="double" and ucd="phys.energy;phys.atmol.final;phys.atmol.level", describing the energy for the final level of the transition which originates this line. To unify results, the value must appear in Joules

- One or more fields **MAY** have ucd="meta.bib", with datatype="char" and arraysize="*", to specify a http links that contains a scientific publication related to the spectral line. Since the URL will often contain metacharacters the URL is normally enclosed in an XML CDATA section (<![CDATA[...]]>) or otherwise encoded to escape any embedded metacharacters.

- One or more fields **MAY** have ucd="meta.ref.url" with datatype="char" and arraysize="*" and free names, to specify URLs that contains extra information related to the spectral line. Same criteria than before about the CDATA section.

6.2 Non-Standard output fields

In many occasions, extra scientifically interesting parameters may be added to the output. Implementors are encouraged to add descriptions and UCDs to the return fields to clarify the meaning of this information and utypes to the Line data Model or other existing IVOA Data Model, whenever possible.

Appendix A: Range definition for input parameters

The Simple Image Access Protocol (SIAP), Simple Spectrum Access Protocol (SSAP) and the Simple Line Access Protocol (SLAP) are URL based protocols, i.e., the constraints in the queries for the input parameters need to be done in a valid URL.

However, the URL standard is not very specific defining special constraints as, e.g., ranges. We describe here some examples of accepted range syntax for URL DAL protocols:

param = x	Equality	Equivalent to param=x
param = x/y	Closed range	Equivalent to param>=X AND param<=y
param = x/	Open range	Equivalent to param>=x
param = x,y	List	Equivalent to param=x OR param=y
param = x/a,b/y	Range list	Equivalent to [param<x AND param<a] OR [param>b and param<y]

References

[1] M.L. Dubernet, P. Osuna, M. Guainazzi, J. Salgado, Atomic and Molecular Line Data Model
[2] D. Tody, R. Plante, Simple Image Access Specification, http://www.ivoa.net/Documents/WD/SIA/sia-20040524.html

[3] R. Hanisch, M. Dolensky, M. Leoni, *Document Standards Management: Guidelines and Procedure* , http://www.ivoa.net/Documents/latest/DocStdProc.html

[4] Peter L. Smith (a), Claas Heise (b), Jim R. Esmond (a), Robert L. Kurucz (a), Atomic spectral line database from CD-ROM 23 of R. L. Kurucz., http://cfa-www.harvard.edu/amdata/ampdata/kurucz23/sekur.html

[5] Various references, NIST National Institute of Standards and Technology, NIST Atomic Spectra Database Lines, http://physics.nist.gov/PhysRefData/ASD/lines_form.html

[6] Ochsenbein et al, VOTable Format Definition V. 1.094, http://www.ivoa.net/internal/IVOA/IvoaVOTable/VOTable-1.094.pdf

[7] McDowell et al, IVOA SED Data Model V.093, http://hea-www.harvard.edu/~jcm/vo/docs/spec.html