VO ACCESS TO CDMS SPECTROSCOPIC DATABASE

N. Moreau[1], M.L. Dubernet[2], and H. Müller[3]

[1]LERMA, Paris observatory, France
[2]LERMA, Paris observatory, France
[3]I. Physikalisches Institut, Universität zu Köln, Germany

ABSTRACT

The CDMS database (http://www.ph1.uni-koeln.de) contains a catalogue of radio frequency and microwave to far-infrared spectral lines of atomic and molecular species that (may) occur in the interstellar or circumstellar medium or in planetary atmospheres. The current state of the database does not provide an easy access: line lists and documentations are currently stored in ASCII files and can be retrieved via a HTML interface, the identification of lines is not straightforward. Our team (LERMA and Paris VO Data Centre, Paris Observatory) has designed an automatic access to these data that allows query of the database via the SLAP protocol [1] following the Atomic & Molecular Line Data Model. We will describe this automatic access as well as its use in various applications.

Key words: Virtual Observatory.

1. INTRODUCTION

The CDMS database contains predictions and documentation for atoms and molecules of astrophysical and atmospheric interest. Data are stored in text files available on-line with one file per species. Due to this format, it is not possible to do some requests to get a subset of data. The automatic retrieval of any subset of data is done with a two steps procedure. Firstly, we imported all of the CDMS data into a relational database. Then we developed an HTTP access to the atomic and molecular lines using the Simple Line Access Protocol developed by the IVOA.

[1]`http://www.ivoa.net/twiki/bin/view/IVOA/SpectralLineListsDocs`

2. CDMS IMPORTATION

2.1. Import tool

The importation of the CDMS files is realised through a dedicated PHP program. PHP is a script language that we use because it provides a lot of convenient string manipulation and database connection functions.

The data are imported into a MySQL database. At the moment, we are importing approximately 360 files. Their total size in ASCII format is about 40 MB. Once they have been imported, they occupy approximately 170 MB.

Using a few modifications, it could be possible to use the same program to import the JPL database which uses the same storage format for its files. However, all the work needed to verify each file and to identify the quantum numbers would have to be done once again.

2.2. Importation procedure

During the importation procedure, we do not just copy data into the database. We identify precisely each quantum number for each transition. Consequently, the user does not have to take a look at the quantum numbers description code used by CDMS. Moreover, we separate the various states transitions that were initially melted inside a unique file. The data contained in each file have also been checked to be sure that they do not contain any mistake.

We noted that some of the informations contained in the CDMS files do not appear in the Line Data Model (the data model developed simultaneously with the SLAP). Indeed, in addition to quantum numbers, each transition may have a "symmetry species". Contrary to the quantum numbers, this information does not have a numerical value but a string. However, as the number of possible value is limited, it could be possible to replace each one of them by a corresponding numerical value to represent it.

2.3. Update procedure

The complete importation procedure is done only once. Then, each week, an update is performed.Each of the files is tested to be sure that it has been correctly downloaded.

The most recent version of a file is compared to the previous one. If something has been modified between the two versions, the newest is imported inside the database alongside the old one. To identify each set of date, we can use two mechanisms. The first one is the version number given by CDMS to each file. If everything is correct, each new downloaded file should be uniquely identified by combining its name and its version number. The second mechanisms is the date at which the data have been downloaded, this information is stored during the download process.

The comparison between the two versions of a data file is done by using a md5 checksum of each file. If the 2 results are different then the file has been modified. There can be a problem if a file underwent a small modification, like a single character. In that case, the results of the checksum will be different and the file will be imported. However, at the moment, this situation never occurred as files are not modified without a good reason.

3. DISTANT DATABASE ACCESS

The access to the database is done thanks to the simple line access protocol normalised by the IVOA. To perform a request, the user has to specify a minimum and a maximum wavelength value. Then he will get a VOTable containing all of the lines contained in this interval.

The output VOTable contains all of the mandatory fields specified by the SLAP 0.5 specification. In addition to them, we added the name of the source database from which data are coming (for now, it will always be CDMS). We also added the CDMS ID of the element concerned by the line in order to allow the user to easily retrieve the corresponding information directly on the CDMS website.

4. SOME DIFFICULTIES WE ENCOUNTERED

4.1. Units problems

The CDMS database contains frequencies for all the lines but does not provide any wavelength. As the wavelength is a SLAP mandatory field, we have to do the conversion on the fly. Still, sometimes, it is not possible to retrieve the original frequency value on the client size because the number of decimals returned isn't sufficient. Consequently, we are returning both the wavelength and the frequency to be sure that the client get the exact value.

Another similar problem occurs with the energy. According to the specification, it must be returned in Joules. However, CDMS provides it in cm^{-1}, which is the unit used by mm spectroscopists.

4.2. Getting quantum numbers

The simple lines access protocol does not give any information about how to get quantum numbers for each line that is returned as an the answer to a query. So, initially, we used a specific service that was able to return the quantum numbers of a given transition. However, it meant that for each lines returned, it was necessary to do another request to get the quantum numbers. As it is possible to get several thousands of lines for a single request, it was problematic to do a new request for each one of them. Finally, we return 2 strings concatenating the quantum numbers using the Line.initialLevel.name and Line.finalLevel.name utypes.

Ex: <TD>N=4 lambda=1 J=4.5 F=5 <\TD>
<TD>N=4 lambda=-1 J=3.5 F=4 <\TD>

With this system, it is up to the client to split the data by himself.

4.3. Statistical weight

As it is currently described in the Lines Data Model [2], the statistical weight is not applicable to CDMS data. Indeed, the data model uses the following physical description : - TotalStatistical weight
- Nuclear Statistical weight

But CDMS does not provide any of those because common factors in NuclearStatisticalWeight are often divided off for a given species. As it is cumbersome to calculate those values, another solution would be to modify the data model. The question is how? A possible solution would be to create a new attribute called PseudoStatisticalWeight that would contain a value and a comment on a how to use it. It would only be valid in equation with ratio of statistical weight.

The following equations illustrates what we call pseudostatisticalweight :

Initial equation = exp1 * g1_total/(exp1 *g1_total+ exp2 * g2_total + exp3 * g3_total)

Here the common factor comes from NuclearStatisticalWeight and the total statistical weights are given by :

g1_total = g1 * common_factor g2_total = g2 * common_factor g3_total = g3 * common_factor

[2] `http://www.ivoa.net/twiki/bin/view/IVOA/SpectralLineListsDocs`

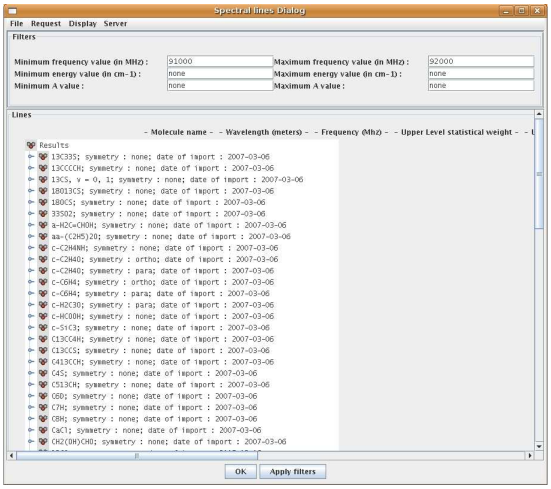

Figure 1. Lines ordered by elements

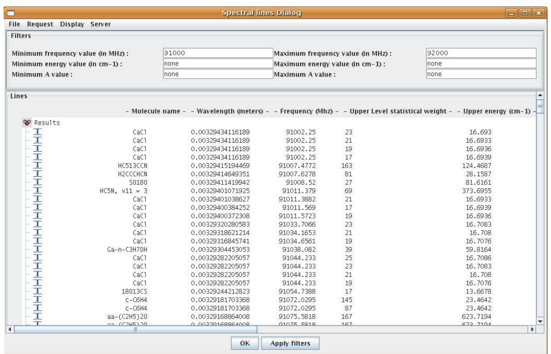

Figure 2. Lines ordered by frequencies

Then the reduced equation can be written :

Reduced_Equation = exp1 * g1/(exp1 * g1 + exp2 * g2 + exp3 * g3)

In this equation, g1, g2, g3 are the quantities given by CDMS that we call PseudoStatisticalWeight.

5. JAVA CLIENT USING THE SIMPLE LINES ACCESS PROTOCOL

In order to test our SLAP service, a simple Java client has been developed. It is able to connect to the three available slap services (ESA, NIST and our own) and to display a fixed number of fields. Moreover, it can save save selected lines into an ASCII file or a VOTable.

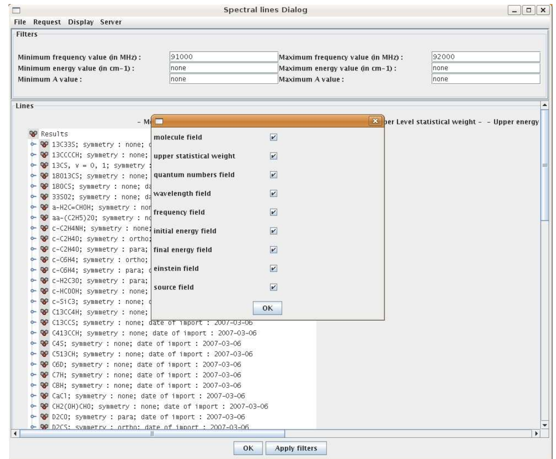

Figure 3. Choice of the fields to display

6. INCOMING EVOLUTIONS

The database importation process has not been completed yet. Some complex elements are not imported because it is necessary to verify the files and to identify the quantum numbers. Concerning the SLAP Java client, there are some modifications that have to be done. First of all, the list of displayed fields has to adapt itself to the service that it is interrogating. For now, this list is fixed because it was initially developed specifically for our service. Moreover, some filters have to be added, as the possibility to choose which molecules will be displayed.

This page is left intentionally blank.