

Universal Atomic Database User's Guide

M. C. Witthoef¹, J. González², J. García^{1,3}, T. R. Kallman¹

¹ NASA Goddard Space Flight Center, Greenbelt, MD, USA

² Instituto Venezolano de Investigaciones Científicas (IVIC),
Caracas Venezuela

³ Catholic University of America, Washington D.C., USA

July 27, 2010

Contents

| | |
|---|-----------|
| Contents | 3 |
| 1 Introduction | 5 |
| 2 Database Description | 7 |
| 2.1 Data Characterization | 7 |
| 2.2 Database implementation | 8 |
| 3 Web page | 9 |
| 3.1 Browser page | 9 |
| 3.2 Grid page | 13 |
| 3.3 Definitions page | 13 |
| A Current data and parameter types | 15 |
| A.1 Data types | 15 |
| A.2 Parameter types | 15 |
| B Level Strings | 19 |
| B.1 Configuration strings | 19 |
| B.2 Term strings | 20 |
| B.3 Level strings | 20 |

Chapter 1

Introduction

The primary purpose of the Universal Atomic Database (uaDB) is to serve as a repository for any atomic data relevant to astrophysics, initially focusing on X-ray processes. It is designed to store data without alteration using the units and any possible fitting formulae used in the publication. A secondary goal is to be able to select and derive data sets for input into modeling codes. To accomplish this goal, while maintaining the integrity of inserted data, the database must be able to distinguish various data types as well as understand the physical units of the data quantities and any parameters. Furthermore, to assist comparisons between datasets, the database storage system is designed to satisfy general quantum mechanical constraints on atomic structure. No evaluation or rating of the data is performed and no data is excluded a priori. The xstarDB is designed to complement atomdb¹, which has many of the same goals and design principles.

All data can be accessed via the web through a search interface. Queries can be as general or specific as desired allowing the user to select data according to

- Data type, describing the physical process;
- Parameter type, such as energy or state;
- Data/parameter value, for filtering on single value or range; and
- Iso-electronic or iso-nuclear sequence.

uaDB is loosely based on the needs of xstar², a computer program for calculating the physical conditions and emission spectra of photoionized gases.

¹<http://cxc.harvard.edu/atomdb/>

²<http://heasarc.gsfc.nasa.gov/docs/software/xstar/xstar.html>

Chapter 2

Database Description

2.1 Data Characterization

In this section, we will briefly describe how data is characterized in the database. First of all, data cannot be entered into the database without a source. The source can have any number of references which contain a web hyperlink to the actual journal web page when applicable. Therefore, for each data set in the database, the source paper can be easily retrieved with just a couple clicks of the mouse.

The database supports a number of data types, each describing a fundamental physical process. Examples of data types include energy levels, partial photoionization cross sections, and total radiative rates. Data types are divided into a number of data sub-types. Each sub-type strictly defines a parameter set and units for a data set stored in the database. Take for example the data type of electron-impact ionization cross sections. One sub-type would be tabulated data with energies in eV and cross sections in Mb. Another sub-type might be a parameterized expression where the parameters are fitting coefficients. In the latter case, no data value is stored.

The database supports unit conversion. To assist in conversions, data sub-types are collected into data groups. Sub-types in the same group can be easily converted to each other; for example energies in eV and Ry. There may exist a transformation for converting data between categories of the same data type, but it is too complex to be handled by the database at this time. An example of this is electron-impact excitation cross sections versus rate coefficients.

Most parameters are simply numbers which are easy to store in the database, such as energy, temperature, or density. However, other parameters require special treatment. The complete set of elements and ions (currently up to $Z = 30$) is supported by the database. Users can refer to elements using the atomic symbol, nuclear charge, or even the full element name. Ions can be specified using chemical notation (e.g. Fe17+), astrophysical notation (e.g. Fe XVIII), or sequence notation (e.g. F-like Fe). Iso-nuclear sequences can be specified

using just the element symbol (e.g. Fe) and iso-electronic sequences can be referenced using the element symbol followed by “-like” (e.g. F-like). Levels are also considered a special parameter. The database stores a hierarchy of configurations, terms, and levels built using *LS* coupling. This allows for configuration-averaged, term-resolved, and fully level-resolved data to all be entered in the database self-consistently, but also consistently across data sets. For data generated using other coupling schemes (e.g. *jj*-coupling), a mapping (constant for each ion) is performed from the *LS*-coupling scheme to the appropriate level string. This mapping is done transparently to the user who will only see the level string provided by the authors of the data.

All sources, data types, sub-types, and parameters are supplied with a label and an identifier. The identifier is a short, unique string used internally to identify each type. These identifiers are largely kept from view in the web page. Instead, the more user-friendly label is provided to describe the types. However, it is not required or guaranteed that these labels be unique.

2.2 Database implementation

Here we explain the structure of the database. This information is not necessary to use the database, so feel free to skip this section.

The Universal Atomic Database is an SQL database consisting of 27 tables. Only six of these tables are used to store the actual data sets and the rest are used to define the sources, data types, parameter types, and units. In order to allow for an arbitrary number of parameters for a given data sub-type, the data values and parameter sets are stored in separate tables. Many tabulated data sub-types, such as cross section versus energy or transition rate versus density and temperature, have hundreds or thousands of data points while the rest of the parameters are constant (such as ion and levels). To store such data efficiently, four additional tables have been created.

There are 8 tables defining data types, parameter types, and the relationships between them. Three tables are needed to define sources and references. Three more tables store units and conversion formulae. The special parameters: elements, ions, configurations, terms, and levels all have their own table. Finally, tables exist keeping a log of changes to the database and another allows for notes to be attached to individual data sets.

The table structure is described in detail in the Developer’s Guide.

Chapter 3

Web page

The uaDB web page is the primary way of searching for and retrieving data sets. There are two methods for searching data: the Browser page and the Grid page. The Browser page allows you to do refined searches and view data while the Grid page is useful for viewing which data is available for each ion. The web page also contains definitions data types, parameter types, and sources. This chapter serves as a description of the web page and a tutorial for its use.

3.1 Browser page

The search form on the Browser page (see Figure 3.1) allows for three ways of searching data: an ion string, the data type, and parameter set. At least one of these must be set to get results, but none are required.

The ion string is a quick way of specifying a particular ion or an entire sequence. The web page understands three formats for ions: astrophysical notation (e.g. Fe XVIII), sequence notation (e.g. F-like Fe), and chemical notation (Fe9+). If just an element (e.g. Fe) is given, then the whole iso-nuclear sequence is taken. Leaving off the element in the sequence notation (e.g. F-like) will search for an iso-electronic sequence.

The data type describes a fundamental physical process, such as wavelength or total photoionization. The exact format of the data may vary (e.g. tabulated versus fitted), but any format can be stored and all are included in a search. On the Browser page, you can also limit the range of data values. This is useful, for example, when trying to find a wavelength in a certain range. If no unit is given, the default unit for each data type will be assumed. However, you can also specify a unit and the database will convert your value to the unit used by each data type. For the example of wavelength, you can specify cm, Angstroms, or eV. The list of available units and their acceptable input labels can be viewed in the Definitions page (see Section 3.3).

Finally, you can select up to three parameters to filter data by. If no parameters are selected then the results will simply be a list of data types and sources

uaDB Browser

http://heasarc.gsfc.nasa.gov/uadb/pages/browser.php

GODDARD SPACE FLIGHT CENTER
Smithsonian Astrophysical Observatory

Help/FAQ
What's New
Site Map
NASA Homepage

Search enter search terms Advanced Search
HEASARC Quick Links
---Quick Links---

HEASARC HOME OBSERVATORIES ARCHIVE CALIBRATION SOFTWARE TOOLS STUDENTS / TEACHERS / PUBLIC

universal atomic Database (uaDB)

HOME BROWSER GRID DEFINITIONS REFERENCES HELP

ION STRING:

DATA TYPE: (optional filter) any value/min: max:

PARAMETERS: (optional filter) -- -- --

Submit Reset [Need help?](#)

[HEASARC Home](#) | [Observatories](#) | [Archive](#) | [Calibration](#) | [Software](#) | [Tools](#) | [Students/Teachers/Public](#)

Last modified: 07/27/10 11:23

[NASA Astrophysics](#)

- [FAQ/Comments/Feedback](#)
- [Education Resources](#)
- [Find helper applications like Adobe Acrobat](#)

A service of the [Astrophysics Science Division](#) at [NASA/ GSFC](#) and the [High Energy Astrophysics Division](#) of the [Smithsonian Astrophysical Observatory \(SAO\)](#)

HEASARC Director: Dr. Alan P. Smale
 HEASARC Associate Director: Dr. Roger Brissenden
 Responsible NASA Official: [Phil Newman](#)
 Web Curator: [Meredith Gibb](#)

[Privacy Policy and Important Notices.](#)

Figure 3.1: Browser search form.

matching the criteria. If you want the results further sub-divided by, say, initial level, you can select that parameter in the drop-down menu without specifying a value below it. In this case, a separate column will appear in the results section with the initial level string. If you want only results from a particular initial level, you can input that level in the text box below the drop-down menu. For ion parameters, use the same rules as for the Ion String search box except that iso-electronic or iso-electronic sequences are not allowed. For levels (of any coupling scheme), you must follow the rules given in Appendix B.

The results listing on the Browser page consists of two parts: header information which is constant for all the results and the list of separate data sets matching the search criteria. See Figure 3.2 as an example for partial photoionization data for the oxygen iso-nuclear sequence. Since there can be a large number of search results, they have been paginated with 20 results per page. In the page navigation bar at the bottom of the results list, there is a total count, forward and backward navigation links, nearby page number links, and an ASCII button for downloading a text file containing the list of results. Each row in the listing is called a data set. It will have a sub-type, a source, a count of the number of data points in the set, and any parameter filters you have requested in the search form. There will also be a column with the ion if you entered in something for Ion String in the search form. To retrieve the data for any set, click on the blue GET button in the leftmost column.

Note that, as seen in Figure 3.2, you can hover the mouse over any link for the data sub-type, parameter type, or source to see the full description of each. Alternatively, you can click on the link to go to a separate page with more information.

Upon clicking on the GET button, the Browser page will be reloaded and both the search form and results section are updated. In the search form, instead of the data type, the specific data sub-type is given and, while you cannot change the sub-type, you can refine your search by specifying a range of data values. Any parameters specific to the data sub-type are shown and allowed to have filter specified. This allows for refinement of the data sets returned. The header information in the results section is expanded to include the specific data sub-type of the data set and its source. The results listing now shows the values of all parameters and the data value of the selected data set. Clicking on the ASCII button will store this data in an ASCII file. For data sub-types which are tabulated versus energy, density, or temperature, there is another GET button which will allow you to get the tabulated data. To grab all of the tabulated data for every result, you can click the ASCII button to get a text file. Clicking on the GET button, however, will update the Browser page again with the full set of data for the transition selected. The search form is absent since it serves no more purpose.

(optional filter)

PARAMETERS: (optional filter)

Submit Reset [Need help?](#)

Data type [partial PI](#)

Description Partial photoionization.

Ion filter O

| | ion | sub-type | source | count |
|---------------------|-------|---|--|-------|
| GET | O I | PI (ICIC), rel. threshold | Opacity Project Team (1995) | 38 |
| GET | O II | PI (ICIC), rel. threshold | Opacity Project Team (1995) | 97 |
| GET | O III | PI (ICIC), rel. threshold | Opacity Project Team (1995) | 59 |
| GET | O IV | PI (ICIC), rel. threshold | Opacity Project Team (1995) | 286 |
| GET | O V | PI (ICIC), rel. threshold | Opacity Project Team (1995) | 286 |
| GET | O V | PI (ICIC), rel. threshold | Opacity Project Team (1995) | 26 |
| GET | O VI | PI (ICIC), rel. threshold | Opacity Project Team (1995) | 26 |
| GET | O III | PI (ICIC), rel. threshold | Opacity Project Team (1995) | 120 |
| GET | O IV | PI (ICIC), rel. threshold | Opacity Project Team (1995) | 120 |
| GET | O VI | PI (ICLS), rel. ground | Garcia et al (2005) | 5 |
| GET | O VII | PI (ICLS), rel. ground | Garcia et al (2005) | 5 |
| GET | O I | PI (ICIC), rel. ground | Garcia et al (2005). Tabulated photoionization cross sections for oxygen ions. | |
| GET | O II | PI (ICIC), rel. ground | Garcia et al (2005) | |
| GET | O III | PI (ICIC), rel. ground | Garcia et al (2005) | |
| GET | O IV | PI (ICIC), rel. ground | Garcia et al (2005) | 226 |
| GET | O V | PI (ICIC), rel. ground | Garcia et al (2005) | 62 |
| GET | O VI | PI (ICIC), rel. ground | Garcia et al (2005) | 16 |
| GET | O VI | PI (ICLS), rel. threshold | Opacity Project Team (1995) | 24 |
| GET | O VII | PI (ICLS), rel. threshold | Opacity Project Team (1995) | 24 |
| GET | O VII | PI (LSLS), delta-function | Bautista & Kallman (2001) | 28 |

heasarc.gsfc.nasa.gov/uadb/.../source.ph...

Figure 3.2: Browser search results for partial photoionization data for the iso-nuclear sequence of oxygen.

3.2 Grid page

The Grid page allows for you to view what data of a given data type is available for all ions supported by uaDB (currently up to Zn). The results can be categorized by coupling scheme (configuration-averaged, term-averaged, or level-resolved) or by source. See Figure 3.3 as an example for total dielectronic recombination data categorized by source. The colors of each circle are defined in the key above the grid. Only 10 sources are supported at a time; a color of black means that you need to click the *more* link in the key to see the next set of sources. If a circle is gray, then there is data from more than one source available for that ion. Clicking on any circle in the grid will take you to the Browser page showing the data sets for that ion.

3.3 Definitions page

The definitions pages describe all data types, parameter types, sources, and units supported by the uaDB. It is rather self-explanatory, but there are a couple things worth pointing out. Under the units sub-menu, if you click on a particular unit (say, 10^4 K for example), you will go to a page with detailed information about the unit. On this page you will see the list of aliases supported by the Browser search form. You will also see a list of other units which the selected unit can be converted to.

Going to the detail page for a source in the sources sub-menu you will find a button labeled DAT. Clicking on this button will take you to the Browser page showing all the data from that source. Furthermore, in the source detail page, there is a list of references belonging to the source. There are up to three buttons available for each reference: Journal will take you to the actual journal page for that reference, ADS Link will take you to the NASA/ADS page for the reference, and Data will take you to the Browser page and show all associated data. Therefore, the reference paper from any data set can be retrieved with just a couple clicks of the mouse.

A final note, the detail pages can be accessed by clicking on the data type, parameter type, unit, or source links appearing in the Browser page.

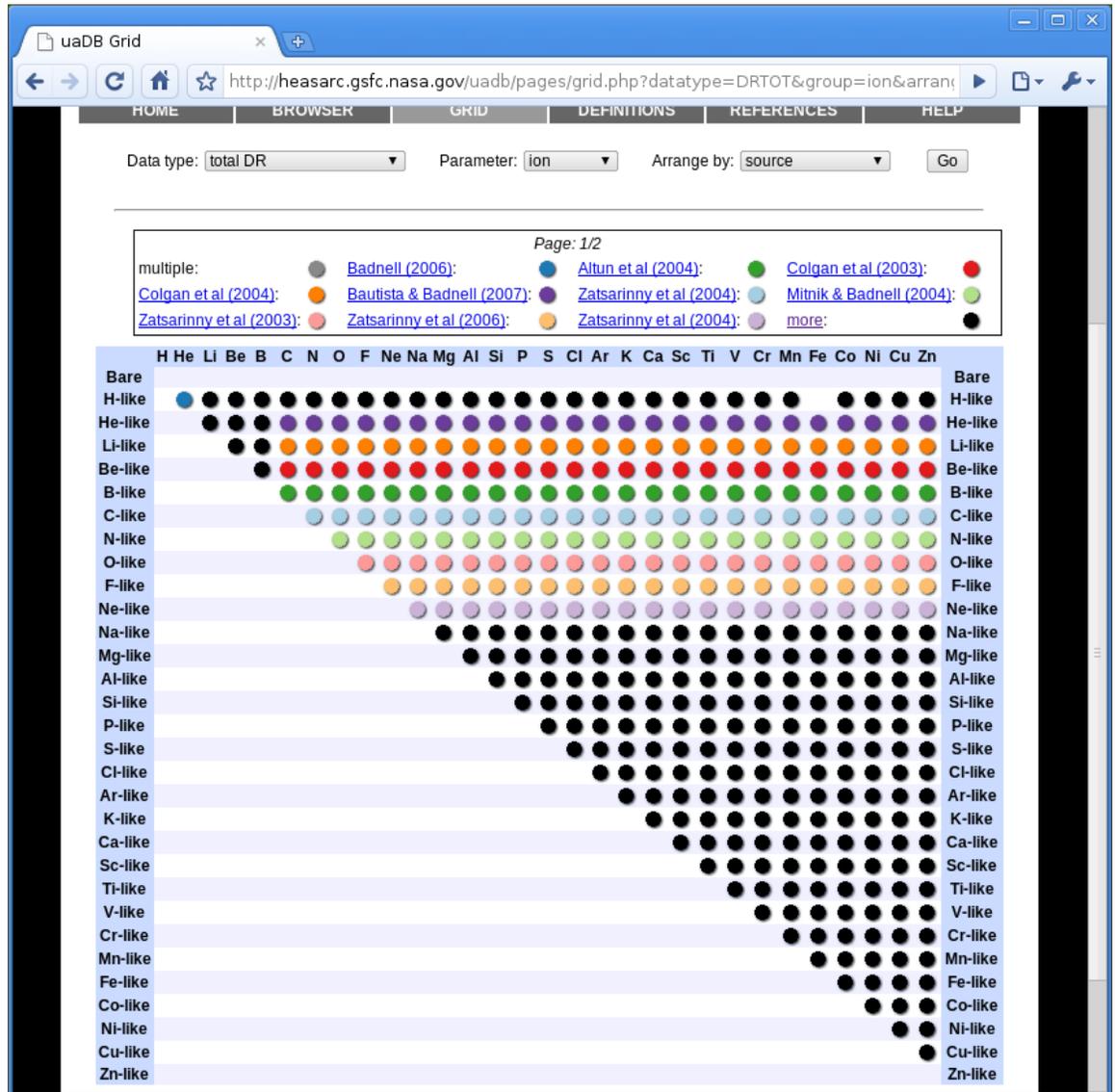


Figure 3.3: Grid results for total dielectronic recombination data by source.

Appendix A

Current data and parameter types

In this appendix we show the list of data types and parameter types currently supported by uaDB.

A.1 Data types

The list of data types is given in Table [A.1](#). These are rather self-explanatory, so we will move on.

A.2 Parameter types

The list of parameter types is given in Table [A.2](#) along with the default unit. Other units are permitted, but if you enter a value in the web search form without a unit, the default unit will be assumed. Most of the parameters are self-explanatory, but a note needs to be made about those which take special values. Values for the ion parameters (`ION`, `ION_INIT`, `ION_FINAL`, and `ION_PROJ`) can take three forms: astrophysical notation (e.g. Fe XVIII), sequence notation (e.g. F-like Fe), or chemical notation (e.g. Fe17+). Note that underscores can be used instead of spaces in the above forms.

Values for the state parameters have up to three parts: the configuration string, term identifier, and J -value. The rules for each is detailed in Appendix [B](#). If a full level string is given, the base term and configuration are also searched for with term- and configuration-averaged data, respectively. Conversely, if just a configuration string is given, terms and levels having that configuration are also searched for.

There are also some parameters for superlevels. Superlevels act much like the level parameters described above except that they represent an entire Rydberg series of levels. Users refer to superlevels in a similar way as with regular levels.

| identifier | label |
|------------|-------------------------|
| ABUND | elemental abundance |
| AUPAR | partial autoionization |
| AUTOT | total autoionization |
| CXPAR | partial charge exchange |
| CXTOT | total charge exchange |
| DRPAR | partial DR |
| DRTOT | total DR |
| EIEPAR | partial EIE, |
| EIPAR | partial EII |
| EITOT | total EII |
| IONPOT | ionization threshold |
| LEVEL | level energies |
| MASS | elemental mass |
| PEPAR | partial PE |
| PIPAR | partial PI |
| PITOT | total PI |
| RADPAR | partial radiative rate |
| RADTOT | total radiative rate |
| RECPAR | partial recombination |
| RECTOT | total recombination |
| RRPAR | partial RR |
| RRTOT | total RR |
| SUPER | super levels |
| TRANSTYPE | transition type |
| TWOPHOT | two-photon decay |
| WAVELEN | wavelength |

Table A.1: Data types

| identifier | default unit | label |
|--------------|-----------------|----------------------|
| DENSITY | $1.\text{cm}^3$ | density |
| ENERGY | eV | energy |
| ION | – | ion |
| ION_INIT | – | initial state ion |
| ION_FINAL | – | final state ion |
| ION_PROJ | – | projectile ion |
| MULTIPLICITY | – | multiplicity (2S+1) |
| PARAMETER | – | fitting parameter |
| STATE | – | state |
| STATE_INIT | – | initial state |
| STATE_FINAL | – | final state |
| STATE_CA | – | configuration |
| STATE_LS | – | term-resolved state |
| STATE_IC | – | level-resolved state |
| SUBSHELL | – | sub-shell |
| SUPER | – | super state |
| SUPER_INIT | – | initial super state |
| SUPER_FINAL | – | final super state |
| TEMPERATURE | K | temperature |
| WAVELENGTH | eV | wavelength |
| WEIGHT | – | weight (2J+1) |

Table A.2: Parameter types

When using full level notation, the base level will follow the same rules as given above for regular levels, but followed by `n1` to represent the Rydberg electron and optionally `S#` where `#` is replaced by the total spin of the system. If the superlevel covers all spin states then this last term is absent. For example, a superlevel building off of the example level above and having a total spin of $2S + 1 = 1$ is specified as: `2p1 2P_3/2 n1 S1`. If we want instead a superlevel built off the configuration only (which then includes all levels of that configuration) and all total spin states, we write: `2p1 n1`.

When storing parameterized formulae in the database, we use parameters of the type, `PARAMETER`. These parameters need not have any inherent physical significance; the data type description will define the expression using these parameters.

Appendix B

Level Strings

While level strings from any coupling scheme can be stored and retrieved from uaDB, currently it only supports searching for *LS*-coupled level strings. In order to guarantee uniqueness, level strings entered into the database must conform to the rules outlined in this appendix.

All states must have a configuration. Term-averaged or level-resolved states must also include a term string and level-resolved states must specify *J*. The rules for each part follow.

B.1 Configuration strings

Configurations are stored in the database using an unambiguous notation which should be familiar to most users. A configuration consists of a space-delimited list of sub-shells in standard order each having the form, *nlm*, where *nl* is the sub-shell (standard order: 1s, 2s, 2p, 3s, ...) and *m* is the occupation number. Note that the shorthand notation of omitting *m* when unity is not used, e.g. 2s1 not 2s. Configuration strings obey the rules:

- all closed sub-shells starting with 1s and ending just prior to the first open (or last) sub-shell are not part of the configuration string,
- the first open sub-shell is always displayed even if it is empty ($m = 0$), and
- all empty sub-shells beyond the first open sub-shell are not displayed.

Some examples:

- $1s^2 2s^2 2p^3$ becomes 2p3,
- $1s^2 2s^1 2p^4$ becomes 2s1 2p4,
- $1s^2 2s^0 2p^5$ becomes 2s0 2p5, and
- $1s^1 2s^2 2p^4$ becomes 1s1 2s2 2p4.

Using a list of occupation numbers as the configuration label was considered and ultimately rejected due to the impracticality of storing Rydberg levels. Consider the configuration, 1s 200p; whereas only 13 characters are needed to store this configuration in the form described above, nearly 40 000 characters are required if using a list of occupation numbers.

To get the number of electrons of a configuration takes two steps; first you need to calculate the number of electrons in the core and then add up the occupation numbers of the visible sub-shells. To get the number of electrons in the core, n_{core} , take the principal quantum number, n , and the orbital angular momentum, l of the first **open** sub-shell and apply the following expression:

$$n_{core} = \frac{1}{3}n(n-1)(2n-1) + 2l^2. \quad (\text{B.1})$$

For a configuration of 4p5 5s2 5p1 we have $n = 4$ and $l = 1$. The above expression yields $n_{core} = 30$ and the total occupation of the visible sub-shells is 8 so this configuration has 38 electrons.

B.2 Term strings

The format for the term should be familiar to most users. It starts with an integer representing $2S + 1$ followed by the spectroscopic letter representing the total orbital angular momentum, L . An example is 2P where $S = 1/2$ and $L = 1$.

B.3 Level strings

To specify the total angular momentum, J , of a level-resolved state, you append the term string defined above with an underscore and the J value. If J is a half-integer then you must use fractional notation. Examples of the term and level strings include: 2P_{1/2} and 1S₀.