International

Virtual

Observatory

Alliance

IVOA Vocabularies
Version 0.11

IVOA Working Draft 2007 September 3

This version:
http://www.ivoa.net/internal/IVOA/IvoaSemantics/Vocabularies-20070903.html
Latest version:
http://www.ivoa.net/internal/IVOA/IvoaSemantics/Vocabularies-20070903.html
Previous version(s):

Editor(s):
Andrea Preite Martinez, andrea.preitemartinez@iasf-roma.inaf.it
Frederic V. Hessman, hessman@astro.physik.uni-goettingen.de

Author(s):
Frederic Hessman, Georg-August-Universität Göttingen, Germany
Andrea Preite Martinez, IASF Roma, Italy
Sebastian Derriere, CDS Strasbourg, France
Soizick Lesteven, CDS Strasbourg, France

Abstract

IVOA *VOcabularies* are named dictionaries consisting of a set of ASCII string tokens representing astrophysical concepts, data, objects, structures, devices, and processes. The tokens of a dictionary can be used to help identify, label, classify, and/or automatically process astrophysical information within Virtual Observatory (VO) or external contexts. The dictionaries are stored in a simple XML document based on a formal schema. It is possible to use XML-style namespaces to access different dictionaries in a syntactically controlled fashion, enabling different groups to define and maintain their own specialized *VOcabularies* while letting the rest of the astronomical community access and use them. Several examples of *VOcabularies* are presented, including a dictionary for the IVOA *Unified Content Descriptors* (UCD).

We also present a proposed *Standard Vocabulary* (SV), consisting of a large number of commonly encountered astrophysical concepts that go beyond the simple data labels of UCD. The purpose of the SV is to provide an immediate and broad common vocabular basis for the VO so that other contexts need only refine or extend the existent vocabulary with tokens representing specialized concepts unique or particularly relevant to those contexts. The SV includes a small number of grammatical tokens that can be used to construct labels expressing more complex entities and relationships. By including the UCD plus SV equivalents of each token in external *VOcabularies*, it is possible to translate semi- or fully-automatically between them.

Status of This Document

This is a Working Draft. The first release of this document was 2007 September 1.

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

This document is based on the W3C documentation standards as adapted for the IVOA.

Contents

1 Introduction

Astronomical information of relevance to the Virtual Observatory (hereafter "VO") is not confined to quantities easily expressed in a catalogue or a table. Fairly simple things like position on the sky, brightness in some units, times measured in some frame, redshifts, classifications or other similar quantities are easily manipulated and stored in VOTables and can now be identified using IVOA *Unified Content Descriptors* (hereafter "UCD"). However, astrophysical concepts and quantities consist of a wide variety of names, identifications, classifications, and associations, most of which cannot be described or labeled via UCD.

Formally, one needs an ontology - a systematic mathematical description of how the concepts are both named and connected with each other - in order to process astronomical information by computer to any depth of complexity. On the other hand, there are many uses of the VO where it would be perfectly adequate to enable computers to handle astronomical tokens that intelligent humans have standardized and for which context-specific processing can be pre-defined.

One of the best examples for the need of a simple token-based vocabulary within the VO is *VOEvent*, the VO standard for handling astronomical events: if someone broadcasts ("publishes") the occurrence of an event, the implication is that someone else is going to want to respond to it, but no institution is interested in all possible events, so some standardized information about what the event "is about" is necessary and in a form which insures that the parties communicate effectively. If a "burst" is announced, is it a Gamma-Ray Burst due to the collapse of a star in a distant galaxy, a solar flare, or the brightening of an accretion disk around a stellar or AGN accretion disk? If a publisher doesn‘t use the label one would have expected, how is one to guess what other equivalent labels might have been used? Thus, rather than waiting for someone to perform the Herculean task of creating a useful VO ontology for astrophysics, most of us would be very happy simply to agree on how we label certain things, independent of what those things mean to individual researchers or computer processes.

There have been many attempts to create something *less* than a full astrophysical ontology - call them "vocabularies" or "taxonomies" - for astronomical purposes.

- The *Second Reference Dictionary of the Nomenclature of Celestial Objects* (Lortet, Borde & Ochsenbein 1994) [3] contains 500 pages (!) of astronomical nomenclature.
- For decades, professional journals have used a set of reasonably compatible keywords to help classify the content of whole articles. These keywords have been analyzed by Preite Martinez & Lesteven (2007), from which they derived a set of

common keywords constituting one of the potential bases for an official VO vocabulary. A similar but less formal attempt was made by Hessman (2005) for the *VOEvent* working group, resulting in a similar list.

- Astronomical databases generally use simple sets of keywords - sometimes hierarchically organized - to aid the users in the querying of the databases. Two examples from totally different contexts are the list of object types used in the *Simbad* database and the search keywords used in the educational *Hands-On Universe(TM)* image database portal.
- The *Astronomical Outreach Imagery* (AOI) working group has created a simple taxonomy for helping to classify images used for educational or public relations.
- Preite Martinez & Lesteven (2007) also attempted to derive a set of common concepts by analyzing the contents of abstracts in journal articles, the list of which should contain more up-to-date tokens/concepts than the old list of journal keywords.
- *Remote Telescope Markup Language* [4], a document definition for the transfer of observing requests that has been adopted by the *Heterogeneous Telescope Network* (HTN) Consortium [5] and is indirectly supported by the *VOEvent* protocol, currently contains several telescope and observation-related taxonomies of terms (e.g. for devices, filters, objects).

The first purpose of this document is to define a VO-wide standard format for such vocabularies. While the definition of the vocabulary format **does** specify how such vocabularies are to be encoded (in the form of an XML document with standard properties), it **does not** prescribe how they are stored, published, transmitted, used or processed.

The second purpose of this document is to describe the proposed IVOA "Standard Vocabulary" (hereafter "**SV**"), a special *VOcabulary* that provides the VO with a common set of standard tokens for astronomical objects, processes, events, observations, instruments, and concepts which are likely to be needed within all VO contexts. In order to make it possible to translate between different standard vocabularies, the format of IVOA *VOcabulary*'s includes not only the individual token strings, their definitions and aliases, but also their equivalences expressed in terms of composed tokens from other Vocabularies, e.g. UCD and SV.

Several examples of SV-compatible vocabularies that could be useful in contexts within and external to the VO are presented at the end of this document.

2 The Format of IVOA VOcabularies

An IVOA-conform vocabulary is formally defined by an XML document that has the form expressed symbolically in Fig. 1 and contains the following elements (the details are defined by the XML schema listed in http://ivoa.net/xml/VOcabulary/VOcabulary-v1.0.xsd):

- **<VOcabulary>**

  the top-level XML element containing references to the defining schemata, IVOA resources, potentially other vocabularies, and the required identifier (e.g. IVORN), name, and version-number attributes;

  - **<Description>**

a short description of the vocabulary (optional);

- **<Reference>**

  a link to an external *VOcabulary* (e.g. the UCD or SV *VOcabularies*) used to define one or more of the defining tokens, optionally including a textual description and/or a namespace prefix used in the document (the prefixes “ucd” and "sv" should be always be used for UCD and SV, respectively);

- **<Entry>**

  the basic unit of the vocabulary. The required attribute "token" contains the token string that constitutes the working part of the vocabulary. Each token can be described by one or more <Definition>'s.

  - **<Definition>**

    one possible meaning of the token in this context, containing the following description, alias, and equivalence elements;

    - **<Description>**

      an optional short description of the <Definition>, including any optional suggested rules associated with the token (e.g. constraints on the use of sub-classifications);

    - **<Alias>**

      one of the optional free-format aliases for the token which are to be considered equivalent with the token but may have the specialized meaning associated with this <Definition>;

    - **<Equivalence>**

      one or more equivalences of the token’s <Definition>, expressed as semi-colon-separated concatenations of tokens from external Vocabularies, referenced by the prefixes listed in the <Reference> elements (see above), e.g. "ucd:phys.absorption (optional).

While there is no formal restriction on the format of the tokens (other than being XML strings), the IVOA suggests that publishers of *VOcabularies* stick to the UCD-like syntax used by the Standard *VOcabulary* as described in the next section.

If no namespace prefix is given in a <Reference>, then the external tokens found in the document without prefixes can be assumed to be from any referenced *VOCabularies* without an assigned prefix. In order to avoid ambiguities, *VOCabularies* with multiple <Reference>’s should be careful to use no more than one without a namespace prefix.

The <Description> and <Alias> elements can have the usual “lang” attribute to indicate which language is used or appropriate; the standard ISO 3166-1 country codes are to be used, e.g. “en” is English, “fr” is French.

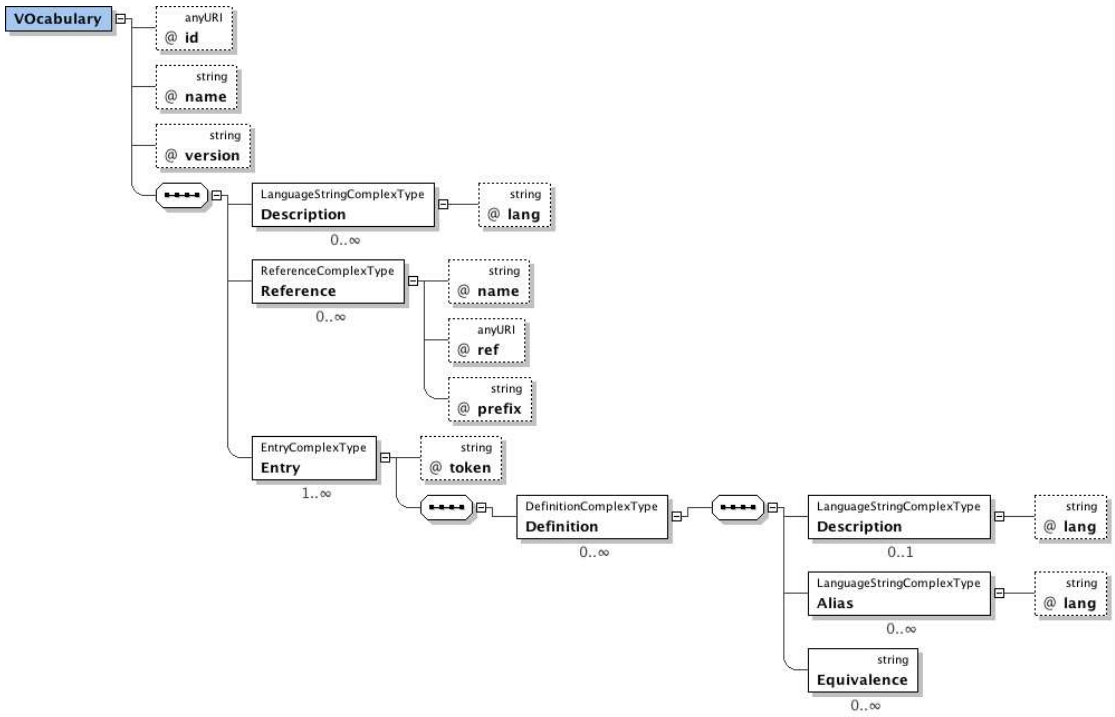

Figure 1. Structure of the IVOA VOcabulary schema.

To illustrate the form of an IVOA *VOcabulary* document, here is a fake XML document defining a VO-compatible cheese vocabulary (the specialized contents have been highlighted in red):

```
<?xml version="1.0"?>
<VOcabulary name="cheese" version="42.0"
        xsi:noNamespaceSchemaLocation="http://www.ivoa.net/xml/VOcabulary/VOcabulary_v0.92.xsd">
        <Description>A silly cheese vocabulary.</Description>
        <Description lang=”de”>Ein lustiger Kaese-Vokabular</Description>
        <Reference name=”food”
                ref=”http://www.ivoa.net/xml/VOcabulary/VOFood.xml” />
        <Entry token="cheese">
                <Definition id="cheese1">
                        <Description>Moldy milk.</Description>
                        <Equivalence>food.cheese</Equivalence>
                        <Alias lang=”fr”>fromage</Alias>
                        <Alias lang=”de”>Kaese</Alias>
                        <Alias lang=”sp”>queso</Alias>
                </Definition>
                <Definition id="cheese2">
                        <Description>Something you say to get people to show their teeth.
                        </Description>
                </Definition>
        </Entry>
        <Entry token="blue cheese">
                <Definition>
                        <Description>A cheese containing a Penicillium culture</Description>
                        <Equivalence>food.cheese;color.blue;appearance.inner.moldy</Equivalence>
                        <Alias>Roquefort</Alias>
                        <Alias>Stilton</Alias>
                        <Alias>Bavaria Blue</Alias>
                </Definition>
        </Entry>
        <Entry token="feta">
                <Definition>
                        <Description>A cheese made from goat's or sheep's milk and aged in brine.
                        </Description>
                        <Alias lang=”tr”>Beyaz Peynir</Alias>
```

```
                    <Equivalence>food.cheese.goat;color.white;taste.salty</Equivalence>
                    <Equivalence>food.cheese.sheep;taste.salty</Equivalence>
                </Definition>
        </Entry>
</VOcabulary>
```

Note that the first <Entry> contains two very different definitions for "cheese", that the second one uses multiple aliases for one definition, and that the last one lists two different equivalences. The tokens in the <Equivalence>'s do not have any namespace prefixes, but this is no problem: there is only one <Reference> given, so all tokens can be assumed to be from the referenced external *VOcabulary*.

There are no other constraints on the form of a IVOA-conform *VOcabulary*. The default description text does not have to be in English if the context requires the description to be in some other language and the token or any alias does not have to be as simple or in the format required for the SV (see below): the assumption is that the individual contexts know what they are doing and will try to make things as simple and useful as is "appropriate".

3 The IVOA Standard Vocabulary

The purpose of the IVOA *Standard Vocabulary* is two-fold:

- by providing a large common base of vocabulary tokens that are likely to be needed in all VO contexts, the SV makes it much simpler to define specialized *VOcabularies* which then only need to express specialized or highly detailed concepts which are impractical or impossible to be maintained in an official IVOA vocabulary maintained by a centralized IVOA Board of Editors consisting of non-specialists;
- when specialized VO contexts express their own vocabulary tokens in terms of their UCD+SV equivalents, it is possible to translate between *VOcabularies*, perhaps even automatically (indeed, the connections and differences between vocabularies may enable the semi-automated construction of ontological structures).

The SV is defined in terms of *VOcabulary* tokens following the typical UCD syntax of period-separated words. While it would be possible - and formally much cleaner because less ontological - to define the SV in terms of words in their simplest form without any formal hierarchies, the names we use for concepts often imply ontological relationships whether we like it or not and the UCD-like syntax is simpler to define, administer, and process.

3.1 Rules for token formation

The process of defining the tokens which make up the SV must be, by definition, an on-going one: as the needs of the VO change, it will be necessary to update, extend, and trim the SV. Early versions of the prototype-SV suffered from the problem of preserving simplicity, consistency, and ease of use while not creating a heavy ontological burden. These experiences resulted in the definition of a set of general rules that have been used to define the SV and should be used to guide the form and choice of future additions to the SV as well as other *VOcabularies*:

a. A token should be preferably written in it's **full, singular, and unambiguous form** (e.g. "star", not "stars" or "st"). SV tokens should be in English.

b. A token can be written following a slight variation on the syntactical rules already used in the IVOA standard UCD, as **a period-separated list of words** :

```
root-token[.sub-token[.sub-sub-token[...]]]
```

(the square brackets indicate optional content) that contain only the ASCII alphabetic (a-z, A-Z) and numeric (0-9) characters and the character "-" (hyphen). The hierarchy suggested by the use of period-separated words is intended to make SV easier to define and use, but there is no formal ontological constraint implied, since even a hierarchically constructed token remains a simple token.

c. If a hierarchy is used, the higher-level token should also be defined (e.g. only define "star.cluster" if "star" has already been defined).

d. A token can be **prefixed by a namespace label** followed by a semi-colon:

```
namespace-prefix:unprefixed-token
```

The prefix must be defined in a <Reference> element in the *VOcabulary*. The standard prefixes "sv:" for the Standard Vocabulary or "ucd:" for the UCDs expressed in their *VOcabulary* form should be used. Note that this looks and should be used like the namespace feature of XML but the namespace prefixes are determined only via <Reference>.

e. The **number of hierarchical levels should be kept to a minimum**. The presence of some level of an implied hierarchy is good in that it makes the tokens simpler to organize, define and process. However, too many levels implies a high degree of ontological organization which may not be helpful later on.

f. **Tokens can be concatenated** to express more complicated meanings following the standard UCD rules, i.e. as a semi-colon-separated list of tokens

```
first-token[;second-token[;...]]
```

but semi-colon-concatenation of UCDs or tokens should NOT occur in the *definition* of the token (i.e. the "token" attribute of <Entry>). For example, "blueCheese" and "cheese.blue" are acceptable tokens but "cheese;color.blue" is not. This restriction is necessary because concatenated tokens must be parseable into the smallest semantic units. For example, does "color.blue;cheese;food.Italian" mean "color.blue;cheese" + "food.Italian" (Italian food having the color of blue cheese") or "color.blue" + "cheese;food.Italian" (Itailan blue cheese)?

g. Generally accepted **abbreviations can be used when the fully written-out original is too long**, although the choice of what is "common" and what is "quite long" is very difficult: the border is somewhere between "PSPC" instead of "position-sensitiveProportionalCounter" and "supernova" instead of "SN". This problem is mitigated by the use of (presumedly shorter) aliases, so the longer forms are preferred.

h. **Capital letters and hyphens are used to conform to standard practice** (e.g. "diffuse.nebula.HII" not "diffuse.nebula.hii" and "process.mass-loss" not "process.massloss").

i. In following standard software practices, **all embedded spaces are dropped and bridged with capital letters** (e.g. "star.brownDwarf", not "star.browndwarf" or "star.brown dwarf").

j. **The names of individual objects** are representable as sub-tokens of the special root-token “named”: e.g. "named.MilkyWay", "named.Earth", "named.deltaCep", “named.MyBackyardTelescope". Although constructs like “galaxy.spiral.MilkyWay” or even “galaxy.spiral.named.MilkyWay” would have been adequate as mere tokens for those named objects with clear token heritages, we needed 1) a generic solution to deal with any potentially named object or concept or a set of concepts (the problem with the first Milky Way token) and 2) a very simple parsing strategy (using special sub-tokens for purely syntactic purposes makes parsing more difficult). The use of a particular root-token for all named things enforces the consistent use of named objects and object names, and removes the need for a formal method for attaching separate name strings to tokens. The ontological information about what the name means is easily placed in the <Equivalence> element by prepending the token “named”: e.g. the SV equivalence description of “named.MilkyWay” is “named;galaxy.spiral”. i.e. the Milky Way is a named spiral galaxy. This solution assumes that each VOcabulary context will not want to use the same name for different objects and permits other contexts to use the same name for something different if needed: e.g. “named.MilkyWay” might mean “named;food.candy-bar” in a IVOA food context, but the use of namespace prefixes will insure that this doesn’t cause any problems (e.g. “food:named.MilkyWay” is not “sv:named.MilkyWay”)

k. **Whenever an object's or a person's name is used as an archetype** of an object class (e.g. "delta Cep" in the sense of "δ Cepheid stars" or "Seyfert" = "AGN of a type first described/discovered by Prof. Seyfert"), **the suffixes "-class" or "-type" are to be appended to the subordinate-token** in order to distinguish the use of the name of the original object or person from a classification derived from that name (which classification just as easily could have been something like "Ia" or "Class B"): e.g. "star.variable.deltaCep-class" and "galaxy.AGN.Seyfert-class". This syntax insures that the use of object classifications is independent of the form of the classification and insures that tokens like "planetary.planet.Jupiter-class" (not actually in the SV!) clearly mean "Jupiter-like planet" and not the planet Jupiter itself.

l. In order to make a *VOcabulary* flexible and to minimized the amount of taxonometric detail in the definition of the standard tokens, subordinate tokens corresponding to particular but **unspecified sub-classifications are indicated in the defining document with the pound-character ("#").** This indicates that any string found in a token in place of the "#" is a detailed sub-classification of undoubtedly useful, but not centrally pre-defined, purpose. An example of this use is "star.spType.#" to enable "star.spType.K5III" as well as "star.spType.dM". The latter shows the benefit of not specifying all possible detail (an impractical if not impossible task in principle). However, the freedom of an unspecified sub-classification puts a considerable burden on the users to conform to standard use, so sub-classifications with syntactically ill-constrained usages and/or with a small number of entries are best placed either directly in the SV (e.g. "star.supernova.TypeIa") or the suggested possibilities should at least be listed directly in the official description to help avoid confusion. The use of the “#” placeholder also means that parsers must be somewhat forgiving if they cannot recognize the substituted entry.

m. **Each entry** of the SV **can be given one or more comma-separated standardized aliases** to help identify standard abbreviations that should frequently occur and can use the classification placeholder "#": e.g. "SN#" for "star.supernova.Type#".

n. When the use of token hierarchies suggests a taxonometric structure but the thing to be described could be placed in different token hierarchies, then the choice should

be determined by fundamental questions like "What is the astrophysically more fundamental meaning?" or "What is the most common usage?". For example, a dwarf nova is physically a close binary star of the Cataclysmic Variable class: should this concept be given the name "star.binary.CV.dwarfNova" or "star.variable.dwarfNova"? Fortunately, this distinction should not be a great problem *by definition*: **the choice of the token should be made intelligently but that choice does not in any way restrict the use of the token**, so the token – in principle – could have any form (e.g. “star.variable.StrawberryJam”, as long as we agree this really means “dwarf nova”).

3.2 Top-level SV categories

The top-level categories or “root-tokens” (*atoms* in UCD jargon) - i.e. those consisting of a single word - define the highest level of informal taxonometric organization within the Standard Vocabulary. The main purpose for this hierarchy is not to sneak in an ontological model but to help the identification, organization, administration, and processing of the tokens.

- **cosmology** (having to do with the large-scale properties of the universe)
- **device** (having to do with astronomically relevant instruments and machines)
- **galaxy** (having to do with galaxies)
- **method** (having to do with astronomical methods, calculations, and calibrations)
- **diffuse** (having to do with diffuse media, e.g. ISM)
- **location** (adjectives expressing cosmic location)
- **math** (having to do with mathematical concepts)
- **misc** (a random collection of standard definitions of potentially wide interest which may relieve the need to create a separate external vocabulary)
- **morphology** (having to do with concepts which are primarily geometric rather than physical)
- **named** (object or concepts with commonly accepted or identifiable names)
- **optics** (having to do with optical surfaces or concepts)
- **physics** (having to do with fundamental physical concepts or processes)
- **planetary** (having to do with non-stellar objects within a planetary system around a stellar object)
- **process** (having to do with astrophysically relevant phenomena, processes, and features)
- **sky** (having to do with definitions or phenomena relevant to an astronomical observer)
- **star** (having to do with stellar objects)

- **stat** (having to do with statistical measures or concepts; see section 3.3 below)
- **source** (having to do with observable astronomical objects)
- **time** (having to do with time and temporal behavior)

The names and number of the root tokens are arbitrary – e.g. “optics” could be considered a part of “physics”. Thus, they have been selected for purely administrative reasons: the lengths of tokens must increase as the number of root tokens decreases, and a finite number of root-tokens makes the SV easier to manage.

The root-token "process" is a “grab-bag” of concepts that are either so complex that they are not simply expressable in terms of a few concepts (e.g. “process.accretion”) or are not immediately physical or mathematical in a fundamental sense but nevertheless represent potentially interesting or important ideas and so aren’t random enough to be stuffed into the root-token “misc”. Examples of process tokens include such concrete things as “process.mountain” (a concept needed in planetology) but also less concrete but important things like “process.rotation” (the generic concept of rotation).

In addition to the normal tokens, there are a few special SV tokens that can be used within a very primitive token grammar, as described in detail within the next section:

- **AND** (logical AND between adjacent tokens)
- **hasElements** (the following token is a subset, member, or part of the preceeding token)
- **isElementOf** (the preceeding token is a subset, member, or part of the following token)
- **NOT** (logical negation of the following token);
- **OR** (logical OR between adjacent tokens);

and the bracket tokens

- **[** (beginning of a token group, used to separate tokens into hierarchical token entities)
- **]** (end of a token group)

3.3 SV token grammar

The concatenation of *VOcabulary* tokens is fully un-constrained beyond the nominal constraints of XML strings and the UCD-like semi-colon separator. With a mere list of tokens, however, only a limited class of labels and relationships can be expressed. The SV therefore supports a primitive grammar to permit the creation of more complex tokens needed for real-world applications operating on complex data and metadata relationships.

The following are a simple set of grammatical rules, guidelines, and typical use cases that should guide VO users of the SV in the definition, parsing, and interpretation of complex tokens.

1. Following common UCD usage, the **main token should come first**, e.g.

   `diffuse.nebula.planetary;star.whiteDwarf`

   implicitly means that the object is primarily a planetary nebula and only secondarily contains a star. However, this distinction obviously is often an arbitrary matter of choice and taste, so some caution is in order when interpreting composite tokens.

2. **Multiple content at the same level of hierarchy** can be expressed using the grammar token “AND”:

   `star.whiteDwarf;AND;diffuse.nebula.planetary`

3. The grammar-token “OR” can be used to indicate **two equally plausible but fundamentally different identifications**, e.g.

   `star;OR;galaxy`

   means that it is not clear whether the object is a star or an unresolved galaxy. The same token without “OR” should be interpreted to mean “an object consisting of a star and an adjacent galaxy”.

4. The token brackets “[“ and “]” are used like classical parentheses: all of the tokens between matching brackets can be considered to represent a super-token or hierarchical token entity, enabling uses like

   `device.telescope;AND;[;device.camera;OR;device.spectrograph;]`

   which means either “telescope+camera” or “telescope+spectrograph”. Note that the bracket tokens are still tokens, i.e. that they need to be separated from adjacent tokens with semi-colons.

5. To indicate that an object is a **member of a multiple/composite object or part of a complex object**, the grammar-token “isElementOf” should be used, e.g.

   `galaxy;isElementOf;galaxy.cluster`

   means “a galaxy that is a member of a galaxy cluster”.

6. Use the grammar-token “NOT” to **negate the state of a token**, e.g.

   `galaxy;NOT;[;isElementOf;galaxy.cluster;]`

   means “a galaxy that is not a member of a galaxy cluster”. Note the use of bracket tokens to insure the correct interpretation (grammar tokens apply only to adjacent tokens or token groups). Another example of the usefulness of negation is the following:

   `[;source;time.variation.burst;em.gamma;];NOT;[;source;em.optical;]`

   which means “a gamma-ray burst source which does not have an optical counterpart”.

7. To indicate that an object is a **potential member of a class** of objects, include the token "stat.possible", e.g.

   ```
   diffuse.nebula.planetary;stat.possible
   ```

   means that the object may or may not be a planetary nebula, and

   ```
   galaxy;[;isElementOf;galaxy.cluster;stat.possible;]
   ```

   means that the object is a galaxy and *only possibly* a member of a galaxy cluster. Note that the “isElementOf” applies only to “galaxy.cluster” and not to “stat.possible”, i.e. only to the immediately following token or token group.

8. To indicate the **absence of information** about an object, include the token "stat.unknown", e.g.

   ```
   diffuse.nebula;stat.unknown
   ```

   means "Nebula or cloud of an unknown nature".

9. An **otherwise unidentifiable "part of" something else** can be labeled using "isElementOf" without a leading (primary) token, e.g.

   ```
   isElementOf;galaxy
   ```

   means that the otherwise unspecified thing can at least be said to be "part of a galaxy". One could have expressed even more information by inserting a leading (and hence primary) token like

   ```
   morphology.spiral;isElementOf;galaxy
   ```

   which then means “a spiral (arm) within a galaxy”. Either form is much more interpretable than that without a grammar-token:

   ```
   morphology.spiral;galaxy
   ```

   which could either mean “a spiral in a galaxy” or “a spiral structure made up of one or more galaxies”.

10. The “hasElements” token is the opposite of “isElementOf” and enables **the listing of contents** (the name “contains” probably better expresses the meaning but the former was chosen so that the tokens resemble each other), e.g.

    ```
    galaxy;hasElements;[;star;AND;diffuse.nebula;]
    ```

    Note that this label does not and is not intended to express quantities: the label above does not say how many stars and nebulae are in the galaxy no does it say that there isn’t anything else which may be contained in a galaxy.

11. The bracket tokens can be used to express complicated relations between tokens. For example, the following expresses the observed eclipse of a G3V star by an orbiting brown dwarf:

    ```
    time.variation.eclipse;[;star.spType.G3V;location.line-of-
    sight.back;];[;star.brownDwarf;location.line-of-sight.front;]
    ```

At first, this extreme variation on the UCD syntax model may look awful, but the structure is actually very simple: the string consists of a concatenation of distinct tokens (all separated by semi-colons, i.e. trivial to parse into equal units); the bracket tokens clearly separate the string into three token-groups; the location tokens can be clearly associated with different object tokens via the bracket tokens; and an explicit ordering of the three main token-groups is not required in order to be able to understand what the string represents.

Since *VOcabularies* are intended to be use in their raw form only by computers, the ability to parse the tokens is primary and the syntactical beauty of the expression is secondary. In fact – as always – the difficulty in parsing a complex expression lies in the interpretation of the results, not in the formal separation into metadata units. In the last example, if one was only interested in the use of "star.brownDwarf" one could simply ignore all of the rest or some application may only be interested in the occurrence of "time.variation.eclipse" with "star.spType.#". Only certain contexts may need to consider the fact that a "time.variation.eclipse" is only possible if there is something eclipsing and something eclipsed, i.e. two other entities necessary to understand the full meaning of the label. This is ontological information that should not be expressed by VOcabulary tokens alone. Thus, the level of relevant detail ultimately depends upon the application itself and the responsibility of the VOcabulary and SV standards is only to enable useful yet parseable expressions.

In summary, the simple rules for parsing SV token grammar are:

- the token string consists of a list of tokens separated in the UCD style by semi-colons;
- the bracket tokens separate the list of tokens into hierarchically organized token groups, each of which can be handled like a single token;
- the properties of a deeper token group hierarchy level don't apply to a higher one (e.g. the presence of an adjective token like "stat.possible" deep within a hierarchy doesn't mean the identification of objects at higher levels is also uncertain);
- the grammar tokens express a very small number of simple relationships between or states of tokens or token groups;
- the grammar tokens operate at least on the following token or token group, and usually define a relationship with the previous token or token group (the only exception to the latter is "NOT", which only operates on the following token or token group).

The SV grammatical tokens and the rules for their use are not part of the *VOcabulary* standard and their use implies the acceptance of these rules.

Some grammar tokens express quite explicit ontological relationships – statements like "objectX is part of objectY but not a member of objectZ" are possible – even though we have gone to great lengths to argue that the whole purpose of IVOA *VOcabularies* and the SV is primarily **not** to express ontological information. The purpose of the SV token grammar is just to enable the expression of a few very simple relationships needed to produce useful labels likely to be encountered in real-life VO contexts. The eclipsing binary and GRB examples above are very good ones: the token strings aren't attempts at expressing what the objects really are – the job of an ontology – but just examples of complex labels conveying a maximum amount of useful label-information with a minimum number of atomic tokens (we don't want to have to define the token "eclipseOfG3VStarByBrownDwarf") and a

minimal amount of ontological baggage. Thus, users of the SV are strongly encouraged not to over-do it by creating overly complex token strings which few of us will be willing to interpret.

3.4 A List of SV tokens

The proposed IVOA Standard Vocabulary is contained in an XML document in *VOcabulary* format in the IVOA Semantics WG home page . The tokens were chosen based on an initial cut of the previous suggestions and sources (see Introduction and references therein), sometimes modified by the above *General Rules*.

For convenience, in the same WG page we provide an alphabetic index of SV tokens for the proposed vocabulary.

In order to add, modify or suppress SV tokens, the same procedure adopted to maintain the list of UCD words will be used. The procedure is described in the document *Maintenance of the list of UCD words, v1.2*, IVOA Recommendation 28 May 2006.

.

4 Example VOcabularies

The Example Vocabularies described below can all be found in the IVOA Semantics WG home page in the form of XML files in *VOcabulary* format.

4.1 UCD-words in VOcabulary format

The XML file containing the UCD list of words (Version 1.23) in *VOcabulary* format can be found in the WG page under the name of `UCD_VOcabulary.xml` .

4.2 Proposed VOcabulary for VOEvent

VOEvent defines the content and meaning of a standard information packet for representing, transmitting, publishing and archiving the discovery of a transient celestial event, with the implication that timely follow-up is being requested. The *VOEvent* syntax provides several possibilities for describing the astronomical content and context of an event but the current version (1.1) doesn't specify any standard for that information. The documents are supposed to be as compact as possible so that they can be transported and processed within a very short time. This means that it is undesirable for the descriptions of the events to contain too much un-preprocessed metadata: if the event is, for example, a Gamma-Ray Burst, then the consumers of the events don't want to have to parse the different possible permutations of "time.variation.burst;em.gamma;..." and just want to look for the acronym "GRB". By providing for the possibility of aliases, this common usage is not only documentable but can be translated to a different context via the SV equivalent.

The beginning of a *VOcabulary* for *VOEvent* is contained in the WG page in file `VOEvent_VOcabulary.xml`. Note that some definitions could have been left out or made perfectly equivalent to the IVOA/SV if the assumption that the IVOA/SV is also used: this may not always be the case.

4.3 Proposed VOcabulary for the AOIM Taxonomy.

The *Astronomical Outreach Imagery Metadata* (AOIM) working group has come up with a simple image taxonomy hierarchy to enable the classification of astronomical images used for outreach or educational purposes. Their work has helped us to identify concepts of interest within the greater astronomical community in a context removed from the typical journal-keyword list or application proposal. Thus, the AOIM working group taxonomy provides a good test of the usefulness of the Standard Vocabulary, since the latter doesn't replace the former but does enable us to create automatic connections between both via the translations implicit in the <Definition> element of the vocabulary.

The point of the proposed AOIM *VOcabulary*, contained in file `AOIM_VOcabulary.xml`, is not just to show that equivalents can be made between the vocabulary chosen by the AOIM working group and the proposed Standard Vocabulary (since the latter was extended to be able to cover the former) but to show that it ultimately shouldn't matter what taxonomy the AOIM ultimately chooses for their own purposes - it frankly shouldn't be the IVOA's business to determine what the solutions to the AOIM working groups problems are - since a translation between the taxonomy and the SV is possible, so that any conversions between the resources of the IVOA community at large and the products purveyed by the AOIM community are easily made. For example, if the data provided by some VO data publisher should be made available for outreach purposes by a AOIM publisher, any internal information used by the data publisher to describe the data can be translated into the corresponding outreach taxonomy token independent of whether either publisher uses the SV as it's primary internal metadata medium.

4.4 Example VOcabulary for the Hand-On Universe(TM) Image Database.

The *Hands-On Universe* (*TM*) project has maintained a public database of images for use by the general public since 199?. The images are very heterogeneous, since they are gathered from a variety of professional, semi-professional, amateur, and school observatories, so a simple taxonomy is used to facilitate the browsing by the users of the database. Thus, the HOU database is a good and simple example of how the Standard Vocabulary could be used outside of the VO.

The proposed HOU *VOcabulary*, in the XML file `HOU_VOcabulary.xml`, was very simple to construct: the HOU image data portal page lists the internal codes (in the HTML source) and the descriptions given to the users, so only the SV correspondances had to be looked up.

4.5 *Example VOcabulary for the* ApJ, A&A, *and* MNRAS *Journal Keywords.*

A list of astronomical concepts, processes and object types is provided by the editors of astronomical journals (namely: The Astrophysical Journal, Astronomy and Astrophysics, M.N.R.A.S.) to help authors of astronomical papers class their works. These **astronomical keywords** have been analyzed by Preite Martinez & Lesteven (2007), from which they derived a set of keywords common to the three journals ApJ, A&A and MNRAS, constituting one of the potential bases for an official VO vocabulary. This common list of astronomical keywords was translated into a *VOcabulary* in XML format (file `AAkeys_Vocabulary.xml`), using SV and UCD correspondances.

Appendix A: Changes from previous versions

1. List of changes from version 0.10:

   - added root-token “named”

References

[1] R. Hanisch, *Resource Metadata for the Virtual Observatory* , http://www.ivoa.net/Documents/latest/RM.html

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

[3]. M.-C. Lortet, S. Borde, F. Ochsenbein, 1994, *Second Reference Dictionary of the Nomenclature of Celestial Objects*, Astron. Ap. Suppl. 107, 193

[4]. *Remote Telescope Markup Language*, Version 3.1, http://monet.uni-sw.gwdg.de/twiki/bin/view/RTML/WebHome

[5]. *Heterogeneous Telescope Network* (HTN), http://www.telescope-networks.org/wiki/index.php/Main_Page