UCD (Unified Content Descriptor)
Version 1.9.9

IVOA Working Draft

2003-10-15

Previous versions: 1.9.5 (2003-10-02)

Authors:

Sébastien **Derriere** <derriere@newb6.u-strasbg.fr>
Norman **Gray** <norman@astro.gla.ac.uk>
Bob **Mann** <rgm@roe.ac.uk>
Andrea Preite **Martinez** <andrea@rm.iasf.cnr.it>
Jonathan **McDowell** <jcm@cfa.harvard.edu>
François **Ochsenbein** <francois@vizir.u-strasbg.fr>
Pedro **Osuna** <Pedro.Osuna@esa.int>
Guy **Rixon** <gtr@ast.cam.ac.uk>
Roy **Williams** <roy@cacr.caltech.edu>

1. Abstract

This document describes the current understanding of the IVOA controlled vocabulary for describing astronomical data quantities, called Unified Content Descriptor (UCD).

2. Status of this document

This is an IVOA Proposed Recommendation 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/.

3. Acknowledgments

This document is based on the W3C documentation standards, but has been adapted for the IVOA.

1. Scope of UCD

1.1. A Controlled Vocabulary for Astronomy

The Unified Content Descriptor (UCD) is a formal vocabulary for astronomical data that is controlled by the International Virtual Observatory Alliance (IVOA). The vocabulary is *restricted* in order to avoid proliferation of terms and synonyms, and *controlled* in order to reduce ambiguity as far as possible. It is intended to be flexible, so that it is understandable to both humans and computers. UCD describe astronomical data quantities, and they are built by combining words from the vocabulary.

A UCD description of a quantity does not define the units or name of the quantity, but rather *'what sort of quantity is this?'*; for example `phys.temperature` is a semantic class description of temperature, without implying a particular unit.

It would be possible to describe astronomical data quantities in a natural language such as English or Hungarian or Uzbek; however, it would be very difficult to expect a machine to 'understand' in any sense. At the opposite extreme, there is an attempt within the IVOA to describe astronomical data in terms of a hierarchical data model, so that there is a place for everything, and everything is in its place. The UCD vocabulary falls between these extremes, and is (we hope) understandable to both human and computer.

1.2. Interoperability as a goal

The UCD committee has tried to resist the temptation to allow the UCD syntax to be overly expressive. Every measurement in science has the possibility of essentially infinite description – the people, the instruments, the error analysis, the reasons, the funders, and so on. We have tried to find a way of organizing atomic specifiers (*words*) so that it is easy to write simple software for machine use, but also possible to write better, more sophisticated software. This organization – in terms of properties and concepts – maps well to knowledge representation methods outside astronomy, and we hope to build more sophisticated "intelligent" systems in the future, a project that has come to be called "UCD3".

The major goal of UCD is to ensure interoperability between heterogeneous datasets. The use of a controlled vocabulary will hopefully allow an homogeneous, non-ambiguous description of concepts that will be shared between people and computers in the IVO.

We hope in the future to put more semantic expressiveness into the UCD framework, but always keeping a pragmatic eye on those who would create and use the software that is to "understand" UCD.

2. UCD Syntax

A UCD is a string which contains textual tokens that we shall call *words*, which are separated by semicolons. A word may be composed of several *atoms*, separated by period characters. The order of these atoms induces a hierarchy. Standard UCD, which are validated by the IVOA, can start with the `ivoa:` namespace, but this namespace is optional. The use of namespaces, indicated by the presence of a colon in the word is possible, but should be avoided as far as possible. They should be used only temporarily, for words that are not yet included into the UCD validated by the IVOA, and they should be replaced by the standard word as soon as it is created. Section 8 describes a procedure for incorporation of new UCDs into the IVOA-approved list.

The character set that may be used in a UCD is the upper and lower-case alphabet, digits, and hyphen. The colon, semicolon, and period are special characters as discussed above.

- The UCD syntax is case-insensitive – all uppercase characters should be converted to lowercase before parsing.
- There should be no whitespace within a UCD.

2.1. Examples of Legal Syntax

The following examples have legal UCD syntax:

```
1  meta.id;src
2  arith.ratio;phot.flux;em.radio
3  ivoa:arith.ratio;ivoa:phot.flux;ivoa:em.radio
4  mynames:concentration;src.galaxy
5  MyNames:Concentration;src.Galaxy
```

In this list, 2 and 3 are equivalent becuase `ivoa:` is the default namespace. Entries 4 and 5 are identical because of the case insensitivity.

2.2. Backus-Naur Form

```
<alpha> ::=  a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z
            |A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|Q|R|S|T|U|V|W|X|Y|Z
<digit> ::=  0|1|2|3|4|5|6|7|8|9
<char>  ::= <alpha>|<digit>|-
<period> ::== .
<semicolon> ::== ;
<colon> ::== :
<word-component> ::= <alpha>|<digit>|<word-component><char>
<namespace-ref> ::= <word-component>
<word> ::= <word-component>|<word> <period> <word-component>
<nword> ::= <namespace-ref> <colon> <word> | <word>
<UCD> ::= <nword> | <UCD> <semicolon> <nword>
```

Note: A UCD is always case-insensitive.

3. Interpretation of UCD

3.1. UCD Semantics

In the semantic web community, knowledge is represented through triples of well-defined words, representing *a property of an instance of a concept and its value*. For example in the statement: "The velocity of the M31 galaxy is 120 km/sec", we have a property (velocity), a concept (galaxy), an instance of the concept (the M31 galaxy), and the value (120 km/sec).

Traditionally in science, it is the values that are most important, and the properties and concepts left implicit, defined by a natural language context. Properties are analogous to the members of data structures, and values as the instances. Thus the "declination" property inherits from the mathematical idea of angle, which inherits from "floating point number". Knowing the property (class) associated with a value (instance) tells us how to compute with the value.

The properties and concepts are metadata (descriptions of data), and the values as the data itself. Often in scientific data there are a large number of statements that all have similar form. For example a table of a thousand galaxies may have a column of velocities -- we can think of a thousand statements, each of the form "The velocity of this galaxy is ....", where the first part (property/concept) is in the column header, it is the same for each entry in the table; and the value part is contained in the table cell, being the property for an instance of the galaxy. This separation of metadata from data allows a computer to process large amounts of knowledge quickly by reasoning only once with metadata, then computing many times with data.

The difficulty of these knowledge systems is in creating a suitable set of concepts and properties. The semantic web community has built semantic nets, ontologies, etc through collections of these property/concept/value triples expressed in the RDF language (Resource Description Format). The data modelling community is involved in this sort of activity, codifying knowledge into a consistent and logical framework. However, a major problem is in the subjective nature of knowledge, that each person tends to have a point of view shaped by experience and current requirement. For example the term "equinox" seems to some to be related to the concept of time, but it is also related to a rather different concept, a coordinate frame.

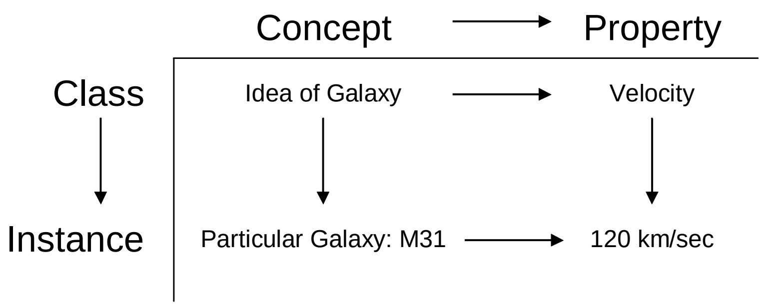

Figure 1: How concepts, properties, classes and instances are related

The UCD system is an attempt to describe simply the most commonly used quantities that astronomers want to exchange. It gives ***standard names to properties of instances of concepts*** (read this sentence twice). In Figure 1, we have:

- The ***concepts*** are the top-level elements in an ontology (top-left). They are analogous to the classes of object-oriented programming. A concept can have subclasses that inherit its properties.
- A concept has a list of ***properties*** (or slots, or parameters) – top right. These may also be classes; in the example, velocity inherits from measurement, and has a subclass relative velocity. A property is like a member of a data structure, it is like a particular view or projection of an object. It is something that can be measured from something abstract.
- The concept has ***instances***, for example M31 is an instance of the concept of Galaxy. The instance is still an abstract object to the computer – there is no M31 in the computer, only the instances of its properties.
- Finally we come to the ***real data!*** An instance of a property has a value, for example the 120 km/sec which is the velocity of M31.

There is a fundamental distinction between properties and concepts as far as the computer is concerned: an instance of a property can be manipulated by the methods of the class (eg, take the cosine of a declination property), but both concept (eg, a telescope, a galaxy) and instance of concept (Keck telescope, M31) are opaque to the computer.

As another example, we can see *'astronomical object'* as a high-level concept, *'star'* is a sub-class of *'astronomical object'*. This class *'star'* has properties such as *'effective temperature'*, *'radius'*, *'spectral type'*, ... (many of them). Vega is an instance of the class *'star'*, and the different properties can be instantiated (filled with data).

An ontology together with a set of individual instances of concepts constitutes a knowledge base.

3.2. The UCD Knowledge Base

The UCD1 set has been built at CDS Strasbourg as a collection of metadata (properties and concepts) that can be used to describe the nature of the large astronomical data holdings at CDS. It has an immeasurable advantage over *ab initio* knowledge systems, in that it claims to be built with minimal subjective interference. The UCD system is built from classifying tens of thousands of existing astronomical metadata, drawn from four thousand published tables. Each column heading is written through as symbol and a description, and an astronomical curator at CDS has tried to find the property and concept that is behind the description.

While many column headings fall naturally into the property/concept semantics, many do not. In a large number of cases, the concept description carries little information and need not be used. For example a table of properties of stars might list the RA, Dec, color, etc etc of each star, and each of these properties could be listed without the explicit statement that they refer to a star. In other words, we abbreviate "*Declination of star*" to simply "*Declination*". The concept description is most relevant when different concept types are involved; for example if there is both "*Declination of star*" and "*Declination of plate-center*" in the same context.

3.3. Building UCD

The UCD system is a description of a knowledge base. It is not an attempt to achieve a purely formal description of astronomy, but rather an attempt to provide a flexible wat of describing commonly used properties, concepts and, in some cases, instances (see section 3.1 for definitions of these terms).

Defining the list of words composing the controlled vocabulary is quite subjective, and the vocabulary will certainly evolve (see section 8). The controlled vocabulary must reflect “things” usually described in natural language, but it must avoid ambiguities and homonyms or shortcuts that are commonly made in descriptions.

Our guideline in the definition of the vocabulary was to study the quantities used in practice (in data tables, in FITS keywords, in the metadata of archives), and try to identify common properties and common concepts.

For example, the temperature is a property that can be measured for a telescope, the atmosphere, or for a star. We therefore define the property `phys.temperature`, and concepts `instr.telescope`, `obs.air`, `src.star`.

A problem often faced is to decide whether we should define some specific words for instances or not. For example, consider the magnitude of a star, the magnitude of a galaxy, and the magnitude of a source irrespective of its classification. The basic property here of course is *magnitude*, which has the UCD word `phot.mag`. The question now is to decide how the concepts of *star*, *galaxy*, and *source* should be described. There are two possible views:

- There is a high level src.class property, that has values `src.class`=’star’, `src.class`=’galaxy’, `src.class`=’unknown’; or
- Create specific words: `src.star` and `src.galaxy` as subclasses of `src`, that would be used in a UCD to indicate that the `phot.mag` property refers to one of them specifically.

The first approach uses additional pairs of "*Property=value*", while the second requires the definition in the vocabulary of word that correspond to the words commonly used by astronomers (a new word in each instance). In general, we recommend the use of the second form where there is already an existing word in common use. However, if the class of `src` were not in the vocabulary (eg Seyfert galaxy), we would recommend the first approach.

This is the case in the description of the electromagnetic spectrum. For a magnitude measured in a given filter, we can imagine two schemes:

- there is the concept of "*filter*", that has many possible instances, for example *"Johnson V*" and *"K"*;
- there are specific words for these instances: `em.opt.V.Jhn` and `em.IR.K`.

We recommend adoption of the second approach because a magnitude always needs to be associated with the filter from which it is measuresd, and because very often the complete formal description of the properties of the filter are not available, and a dedicated words allows a fast comparison.

We adopt the following conventions in building a UCD:

The primary word of the UCD refers to a property, e.g.

- `meta.id`
- `meta.note`
- `phot.flux`
- `phys.mass`

Secondary words can indicate either:

- A concept that the property refers to:
  - * `meta.id;src`
  - * `meta.id;inst.telescope`
- Another property that the primary word refers to:
  - * `stat.error;pos.eq.ra`
- Or information related to the primary word:
  - * `phot.flux;em.opt.V.Johnson`

Some of the words we define can only be secondary UCDs. But properties can be secondary UCDs in some cases (`stat.error;pos.eq.ra`). We do not describe the exact relation between primary and secondary words in UCD2. In the future (UCD3), the links will be described by facets of a ontology or some similar knowledge description that will allow inference.

The UCD syntax defined above yields a collection of words. The first word of the UCD is called the primary word of the UCD, and it is interpreted as the property that is being expressed, and the other words may define one or more concepts to which that property refers.

Something that is a concept (eg `instr.telescope`) may not be used as a property. But something that is a property (eg `pos.eq.dec`) may be used as a secondary word, for example the "error of the declination" has primary word `stat.error` and secondary word `pos.eq.dec`.

3.4. Examples

The UCD team has used these guidelines to make a "best effort" to reduce many complex statements to this framework -- some examples are described below.

- **Literature citation, NGC identifier, URL**. The property here does not connect to a concept, so the UCD has only the primary word (its property), but no concept word. Each of them is part of the same tree of generalized pointers:
  `meta.refer.bibcode`
  `meta.id`
  `meta.link.url`

- **Error of Right Ascension of galaxy**. We identify the central property as "error", and the concept as "right ascension", with a subsidiary word about "galaxy". Therefore the UCD might be
  `stat.error;pos.eq.ra;src.galaxy`

- **Latitude of the Telescope (on Earth)**.
  `pos.planetographic.lat;instr.telescope;planet.Earth`

- **Photometric flux through a Johnson K filter**. The property is `phot.flux`, and the related concept is that the bandpass is in the infrared. Therefore we use a UCD like
  `phot.flux;em.IR.K`

- **The ratio of flux in one band to another**. The primary property is “ratio”, and the concept is photometric flux:
  `arith.ratio;phot.flux`
  This is in many ways similar to a color index – which is a difference of a magnitude in one filter (eg B) from a standard magnitude (in V). However, in this case, there is a comon word used by astronomers (color). The result is therefore:
  `phot.colorIndex.Johnson.B-V`

- **Theoretical prediction of an absolute magnitude**. We already have a UCD to cover the concept of absolute magnitude, it is `phot.mag.abs`. A specifier can be added such as `meta.model` to indicate that this is not measured, but rather modeled.

4. Use cases

4.1. Discovery and UCD: Matching Function

The idea of the UCD vocabulary is that it is used to label parameters, table columns, etc in the astronomical literature; hopefully this will allow a scientist to search for a given data concept (*"find a table with photometric magnitudes in the infrared"*), and it will allow validation by the computer of proposed operations (*"are you sure you want to add a magnitude to a semi-major axis?"*). In this section, we define our concept of computer understanding in terms of the existence of a "matching function" ***Mu(u1;u2)*** that takes two UCD descriptions *u1* and *u2* and evaluates a degree of match from zero (no similarity) to one (exact match). The functionality above can then be coded in terms of good or bad matches *("find a table with a UCD u so that Mu(u; “*`phot.mag;em.IR`*”) is greater than 0.8").*

The compromise in defining the UCD standard is between expressiveness and the work of building matching functions. A very simple UCD vocabulary cannot express sophisticated concepts, but the matching function would be just a string match -- exact match gives mu=1, else mu=0. A complex UCD vocabulary can express complexity, the most complex being natural language, but this makes the matching function a great challenge. We expect to see building of matching functions (mu) for the use of UCD as a discovery tool. Reasonable behavior for such functions would include axioms such as:

- Self-match: Mu(u, u) = 1

- Commutativity: Mu(u1, u2) = Mu(u2, u1)
- Transitivity: If Mu(u1, u2) > 0 and Mu(u2, u3) > 0 then Mu(u1, u3) > 0

A very simple mu could be built by substring match on the primary words of the UCDs. This would tell us that the error of a quantity is semantically related to the quantity itself – and of course it is. It would tell us that two UCDs in the same branch of the tree are semantically related – and presumably they are. Another type of matching could work with the textual descriptions of the UCD, looking for Google-like phrase matching between the descriptions of u1 and u2.

More complex matching functions would take account of the secondary words in the UCD in a way that has not yet been well defined. Work is needed in this area.

4.2. Database Access and UCD: Translation Layer

UCD will be used in practice for ***exchanging*** information using a controlled vocabulary. They are used in the VOTable standard to attach a standard description to table column names, for example. The data providers do not need to change the internal descriptions of their existing databases. Nor is it required that people building from scratch a new VO-compliant service use UCD in the core of their system.

What is needed for interoperation with other systems is a "***translation layer***" that is able to associate UCD to the parameters that are used internally, so that the output of the service contains a standard description that can be interpreted by other VO services.

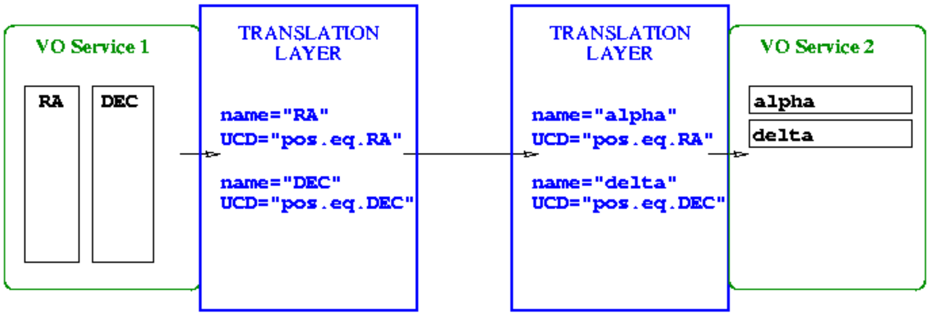

Figure 2: Services use UCD to exchange information. A translation layer is used to interpret the internal description in terms of UCD.

In Figure 2, a first VO service describes internally the right ascension and declination with names RA and DEC. For sending data to another service expecting right ascension and declination as an input, it uses a translation layer to attach UCD to its parameters. The second service also has a translation layer that can interpret UCD into its own parameters.

The mapping done by the translation layer can be done using XML files. For the second service above, we can specify that quantities corresponding to UCD `pos.eq.ra` and `pos.eq.dec` are to be found in the database table Obs-Table, which has column names **`alpha`** and **`delta`**:

```
<ucdToDb>
      <ucd name="pos.eq.RA" table="Obs-Table" col="alpha" />
      <ucd name="pos.eq.DEC" table="Obs-Table" col="delta" />
      <ucd name= ... />
</ucdToDb>
```

5. Software and Services

What is the nature of the software and services that will work with UCD?

5.1. Services at CDS

Several web services have been implemented at CDS Strasbourg to aid in the exploitation of UCD. Those below are available at http://cdsweb.u-strasbg.fr/UCD. The following list covers some of these:

5.1.1. Resolver

The resolver service – given a UCD, the associations of the previous section will enable us to get a textual description of what it means. New namespaces can provide a resolver by the mechanism of section 4.

5.1.2. Listing and Browsing

These services allow a dynamic view of the tree of UCD, either as a single text file, or as a Javascript-enabled tree-browser.

5.1.3. Search Engine

This service allows the input of natural language, and it searches for matches in the text description of the UCDs . A further extension connects to metadata about Vizier tables that use those UCDs. This tool can be used to find an appropriate UCD for labeling data. A batch-oriented version accepts a file of keywords, data types, and other information and tries to find suitable UCDs.

6. The Proposed UCD Tree

The list enclosed here proposes the basic elements only, and does not detail each node.

`arith`	quantities related to arithmetic and mathematics, including count, difference, ratio
`em`	the electromagnetic spectrum
`meta`	quantities related to metadata, such as identifiers, flags, notes, URL, and
`instr`	quantities related to an instrument; typical sub-levels are telescope, observatory, etc.
`obs`	observation methods such as detector, filter, plate, spectrograph, exposure time, etc.
`phot`	All photometric measurements, organized according to the wavelength; includes polarization.
`phys`	Generic physical quantities, such as length, velocity, mass, and including atomic & molecular concepts and properties, temperature, pressure, gravity, etc...
`pos`	Position in the sky, reference frames; including equatorial, galactic etc coordinates; geocentric, heliocentric etc; and precession and nutation. Also includes position on the surface of the Earth.
`spec`	Quantities related to spectroscopic measurements
`src`	properties of the observed source of radiation: source classifications and morphology, extension in the sky, variability,
`stat`	statistical quantities and quantities related to model fitting, including concepts such as error, maximum, residuals.

`time` Quantities related to time.

The new tree was simplified compared to the current UCD tree; important modifications include:

- Atomic and molecular data are moved to a branch of `phys`
- Fitting and model data are moved to a branch of `stat`
- The `src` branch is introduced for photon sources such as stars, galaxies etc.
- Observatory information is moved to the `instr` branch.

More details about some of the most important branches are shown in the annex below.

7. Discussion

There are a few points we have faced when trying to describe the existing columns with the new UCD scheme. We are listing these points here, together with a few more questions that, we think, should be answered before submitting a draft to the discussion forum.

7.1. Freedom in Hierarchy

The new UCD scheme does not keep the concept of `node' and `leaves': UCDs at any level can be used to describe some parameter, whether sub-levels are existing or not. This rule implies that we do not use a qualifier like `misc` or `gen`: if a quantity is not accurately defined, we just use the `parent' UCD. An example comes with the division of the electromagnetic spectrum: the standard UCD words can label parts of the spectrum, for example `em.IR.3-4um` and `em.IR.4-8um`. To label a region from 3 to 5 um, the recommended UCD is the generic `em.IR`.

7.2. Standard Usage

The elements of the tree make use of a standard vocabulary, in the sense that a single word is used to designate a physical concept or quantity. For instance, if `electron` is used to designate any electron-related quantity, we write e.g. `phys.temp.electron` to designate the electronic temperature and not an abbreviation like `phys.temp.el`; conversely, `electron` keeps the same meaning among all UCDs. We should try to maintain a list of these meanings -- we are for instance using `temp` for temperature, `phys` for physical, and so on.

7.3. Aliases

The `standard' UCD list makes use of a restricted vocabulary, but the vocabulary could be extended by synonyms (aliases). For instance `arith.ratio` and `arith.divide` could be considered as synonyms. Another example: `em.IR.100-200um` could be a synonym to `em.radio.1500-3000GHz`. We expect to make extensions like these to the tree in due time.

7.4. UCD as Semantic Web

In the next version of UCD, we will use it as part of a larger effort to build a semantic grid of astronomical data. This will be a large new project tentatively called UCD3. The idea is to build a semantic net that connects parameters, UCDs, names of table attributes (in multiple tables), identifiers of datasets in the VO registry, abstract grouping concepts, and so on. We hope to use the language of the semantic web -- RDF -- to express relationships, and topic maps or ontology to build, expose, and reason from this knowledge.

8. UCD Steering Committee

8.1. Creation of a Board for New UCD Words

We believe that the inclusion of new UCD words must be a flexible process, yet controlled. The best way to accomplish these two needs would be to create a proper scientific board that would study new UCD requirements and, within a given period of time, give an answer as to whether a new UCD must or must not be included in the UCD standards.

The use of “mission-specific” namespaces has been addressed in many occasions, and we believe that namespaces should be avoided as much as possible. There has been an exercise in revising the VOX words for the SIAP protocol and trying to assign existing UCDs to them, or proposing new UCD words for the non-existing ones.

The responsibility of the board would consist of studying the cases where a UCD word is proposed and to figure out whether the proposed word should be accepted or rejected, and in case of rejection recommending the closest existing word that should be used.

In case a new word is accepted into the main tree, an internal procedure should be established so that the new UCD becomes live after a proper internal new release in a short period of time.

It should be agreed whether this board would study the proposed cases in an “on demand” basis or would collect requests and study them on a periodic basis.

A suggestion on the formation of this scientific committee would be that it might contain people from CDS (as they have the experience and the resources) but it should be offered to all relevant parties. It would also be very important to have a member from the data providers community, as the scientists view on some issues might not include other important views from data providers.

8.2. A procedure to request new UCD words

A procedural document should be created to make it easy to a user to ask for a new UCD and to understand the implications of doing so. This document would address:

- the contact point to ask for new UCD
- the life-cycle of the process of asking for a new UCD
- when and how a new UCD becomes live
- what to do if a UCD is rejected

This type of actions could (and should) be supported by tools like an automatic form that is filled in and sent to the scientific board, giving an answer back to the user acknowledging the request, and giving a time estimate for an answer. All these issues will be suggested in a separate point.

Lessons should be learnt from other projects where similar boards exist. There should be a thorough investigation (maybe from the board mentioned above) of how other projects have worked in this direction (like the Planetary Data System (PDS), the FITS consortium, the W3C) and try to get the right things from them while avoiding the wrong ones.

8.3. Creation of a Technical Board

There should be tools available for the user to check for the existence of UCDs, etc. Some of these tools exist already in CDS, and they are good candidates to become the sort of “official” tools for the UCD standards. However, we feel it is necessary to have a proper technical board that could, eventually, decide on what tools are really necessary to make the UCD work feasible and as easy as possible for the user. This board would be mainly in charge of writing proper requirements for the tools. The management of resources, etc., should be handled by the concepts wanting to work for the VO project, but the definitions of requirements, etc., should be centralized on this board.

8.4. Contact point for UCD issues

We feel the necessity to create a contact point to which all UCD related matters can be addressed. This contact point could be a web address devoted explicitly to that in the context of the VO, a properly organized web place, where all the tools would be available, as well as all documents and procedures for creation of new UCD words, etc., with practical examples and the like.

9. The Tree of Primary UCD Words

The table below contains just a few elements of the revised UCD tree; a fully qualified tree will be prepared as the result of discussions and exchanges. Notice that the first word of a UCD must be a property, but subsequent words can be either concepts or properties.

meta (metadata quantities)		
property	`meta.id`	An identifier or name – the concept may be named in a secondary UCD.
property	`meta.note`	A note or comment in natural language.
property	`meta.code`	A code or flag in some local system.
property	`meta.link.URL`	A URL.
property	`meta.link.IVO`	An IVOA identifier for a service or dataset.
property	`meta.refer.bibcode`	A bibcode pointing to a journal article
phys (physical quantities)		
concept	`phys.at`	A physical atom.
property	`phys.at.trans.rate`	Transition rate.
property	`phys.temp`	Temperatures (effective, electronic, etc...)
pos (positional data)		
property	`pos.ang`	Angular Distance and related quantities
property	`pos.pixcode.HTM`	Hierarchical Triangular mesh position
property	`pos.eq.ra`	Right Ascension in Equatorial frame (angular)
concept	`pos.gal`	Concept of galactic coordinates.
stat (statistical or fitting quantities)		
property	`stat.error`	A measure of the error of a parameter.
property	`stat.max`	A maximum or upper limit.

src (quantities that are a property of a source)

concept	src	An astronomical source in the sky

phot (properties measuring the EM radiation received from a source)

property	phot.flux	Radiation flux (energy per unit time per area).
property	phot.mag	Magnitude of a celestial source.
property	phot.count	A photon count measurement (per unit time per
property	phot.colorIndex	A difference in magnitudes, each measured in a

10. Organisation of the Wavelength Spectrum

There has been much debate in the UCD forum over division of the electromagnetic spectrum, since this is where the qualitative and the quantitative meet. If we put every word about spectrum coverage into the UCD, there would be hundreds of terms, therefore we have chosen to keep to a rational division (below) plus a very few special words.

The wavelength spectrum is first divided in the 7 classical domains radio / IR / Optical / UV / EUV / X-ray / gamma. Further divisions are made to define the large bands classically used in optical / IR / UV, and in radio frequencies we keep bands spaced by a factor 2. In Figure 3, a special word is there for `Halpha` as a subclass of `opt.R`. If a desired band does not fit in the rational list, it is recommended to use the smallest enclosing band.

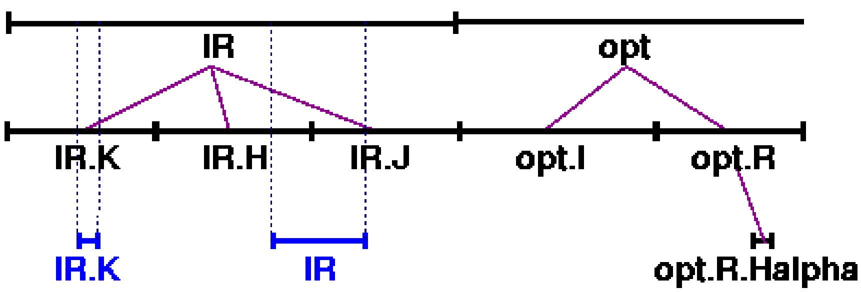

Figure 3: Hierarchical organization of the electromagnetic spectrum. The standard bands are represented in black. The suggested description of the non-standard blue ranges is shown in blue: in each case, we use the smallest enclosing standard band.

The overall list is as follows:

UCD designation	Lambda	Freq	Energy	Notes
		Radio Regime		
em.radio.20-100MHz	>3m	<100MHz		
em.radio.100-200MHz	1.5-3m	100-200MHz		
em.radio.200-400MHz	75-150cm	200-400MHz		
em.radio.400-750MHz	40-75cm	400-750MHz		
em.radio.750-1500MHz	20-40cm	750-1500MHz		
em.radio.1500-3000MHz	10-20cm	1.5-3GHz		

em.radio.3-6GHz	5-10cm	3-6GHz		
em.radio.6-12GHz	2.5-5cm	6-12GHz		
em.radio.12-25GHz	1.2-2.5cm	12-25GHz		
em.radio.25-50GHz	6-12mm	25-50GHz		
em.radio.50-100GHz	3-6mm	50-100GHz		
em.radio.100-200GHz	1.5-3mm	100-200GHz		
em.radio.200-400GHz	750-1500µm	200-400GHz		
em.radio.400-750GHz	400-750µm	400-750GHz		
em.radio.750-1500GHz	200-400µm	750-1500GHz		COBE 240µm
em.radio.1500-3000GHz	100-200µm	1500-3000GHz		COBE 140µm
Infra-Red Regime				
em.IR.60-100um	60-100µm	3-5THz		IRAS 100um
em.IR.30-60um	30-60µm	5-10THz		IRAS 60µm
em.IR.15-30um	15-30µm	10-20THz		IRAS 25µm
em.IR.8-15um	8-15µm	20-37.5THz		N band; IRAS 12µm
em.IR.4-8um	4-8µm	37.5-75THz		M band;
em.IR.3-4um	3-4µm	100-150THz		L, L', L"
em.IR.K	2-3µm	75-100THz		K band
em.IR.H	1.5-2.0µm	200-300THz		H band;
em.IR.J	1.0-1.5µm	150-200THz		J band;
Optical Regime				
em.opt.I	750-1000nm	300-400THz	1.2-1.6eV	I band;
em.opt.R	600-750nm	400-500THz	1.6-2.0eV	R band;
em.opt.V	500-600nm	500-600THz	2.0-2.4eV	V band;
em.opt.B	400-500nm	600-750THz	2.4-3.0eV	B band;
em.opt.U	300-400nm	750-1000THz	3.0-4.0eV	U band;
Ultra-Violet Regime				
em.UV.200-300nm	200-300nm	1000-1500THz	4-6eV	UV1 band
em.UV.100-200nm	100-200nm	1500-3000THz	6-12eV	UV2 band;
Extreme Ultra-Violet Regime				
em.EUV.50-100nm	50-100nm	3-6PHz	12-24eV	Lv{Limit}=91.2nm
em.EUV.10-50nm	10-50nm	6-30PHz	24-120eV	
X-ray Regime				
em.X-ray.soft	6-100Å	30-500PHz	0.12-2keV	
em.X-ray.hard	0.1-6Å	0.5-30EHz	2-12keV	
Gamma Regime				
em.gamma.soft	0.25-10pm	30-1200EHz	12-500keV	
em.gamma.hard	<250fm	> 1200EHz	>500keV	e+/e-